```
 OPTIONS NODECK,LIST,XREF,NOREL,OBJ(P)

THE LIST OF OPTIONS USED DURING THIS ASSEMBLY IS--   NODECK,LIST,XREF,NOREL,OBJ
```

```
                              EXTERNAL SYMBOL LIST
SYMBOL    TYPE                                    VER 15, MOD 00  04/07/22  PAGE   1
#KSYMB    MODULE
```

```
ERR LOC  OBJECT CODE     ADDR STMT SOURCE STATEMENT          VER 15, MOD 00  04/07/22  PAGE   2

    0000                        1 #KSYMB START 0
                                2        PRINT ON,NODATA
                                3 *      @SYS  EXP-N
                              214+       PRINT ON
                              215 *      @FXD  EXP-N
                              620+       PRINT ON
                              621 *      @SPF  EXP-N
                             1084+       PRINT ON
                             1085 *      @HDW  EXP-N
                             1270+       PRINT ON
                             1271 *      @CAN  EXP-N
                             1374+       PRINT ON
                             1375 *      @ERM  EXP-N
                             1997+       PRINT ON
                             1998 *      @B@E  EXP-N
                             2898+       PRINT ON
```

$B@EQU – S/3 BASIC COMPILER FIXED ADDRESS EQUATES

```
                    2901 ************************************************************************
                    2902 * 5703-XM1      COPYRIGHT IBM CORP. 1970                           *
                    2903 *              REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083  *
                    2904 *                                                                  *
                    2905 ************************************************************************
                    2906 *STATUS                                                            *
                    2907 *    VERSION 1 MODIFICATION 0                                       *
                    2908 *                                                                  *
                    2909 *FUNCTION                                                          *
                    2910 *    KSYMBL PERFORMS THE FUNCTION OF THE 'SYMBOLS' KEYWORD.  THIS   *
                    2911 *    COMMAND CAUSES A LISTING OF ALL VARIABLE NAMES USED IN THE BASIC *
                    2912 *    PROGRAM IN THE WORKFILE.                                       *
                    2913 *                                                                  *
                    2914 *ENTRY POINTS                                                      *
                    2915 *    THE ENTRY POINT TO KSYMBL IS #KSYMB, THE FIRST BYTE FOLLOWING THE *
                    2916 *    PROGRAM HEADER.                                                *
                    2917 *                                                                  *
                    2918 *INPUT                                                             *
                    2919 *    INPUT TO KSYMBL IS THE INPUT LINE BUFFER AND THE PROGRAM IN THE *
                    2920 *    WORKAREA.                                                      *
                    2921 *                                                                  *
                    2922 *OUTPUT                                                            *
                    2923 *    OUTPUT FROM KSYMBL IS THE LISTING OF A SYMBOL TABLE.           *
                    2924 *                                                                  *
                    2925 *EXTERNAL REFERENCES                                               *
                    2926 *    * $XRSAV - REGISTER 2 (@XR) SAVE AREA                          *
                    2927 *    * SCANIT - ENTRY TO DELIMITER SCAN MODULE                      *
                    2928 *    * SCKOUT - ENTRY TO MODULE SCKOUT TO SYNTAX-CHECK OUTPUT SPEC   *
                    2929 *    * $CAERR - ERROR CODE SAVE AREA                                *
                    2930 *    * SCKDEV - ENTRY TO MODULE SCKOUT TO CHECK OUTPUT DEVICE READY  *
                    2931 *    * GRABIT - ENTRY TO MODULE TO RETRIEVE FILE LINES              *
                    2932 *    * GRWHAT - BRABIT INDICATOR, SET TO RETURN TEXT (GRAEFR)        *
                    2933 *    * GRTEXT - AREA WHERE FILE TEXT IS PLACED BY GRABIT            *
                    2934 *    * SVARAB - ENTRY TO MODULE TO FIND VARIABLES IN FILE LINES      *
                    2935 *    * $CARPL - ENTRY TO LOAD #GUFUD, THE FILE UPDATE PROGRAM        *
                    2936 *    * 4CAERK - ENTRY TO LOAD #ERRPG, THE ERROR PROGRAM             *
                    2937 *    * DLPRNT - ENTRY TO MODULE TO PRINT A LINE                     *
                    2938 *    * SCKERR - ERROR EXIT FROM SCKOUT                              *
                    2939 *                                                                  *
                    2940 *EXITS,NORMAL                                                      *
                    2941 *    NORMAL EXIT FROM KSYMBL IS TO $CARPL TO LOAD #UFUD.            *
                    2942 *                                                                  *
                    2943 *EXITS,ERROR                                                       *
                    2944 *    ERROR EXIT FROM KSYMBL IS TO $CAERK TO LOAD #ERRPG.            *
                    2945 *                                                                  *
                    2946 *TABLES/WORKAREAS                                                  *
                    2947 *    * KSYALP - TABLE OF THE 29 ALPHABETIC CHARACTER                *
                    2948 *    * KSYLVC - BEGINNING OF SINGLE LETTER VARIABLE SYMBOL TABLE     *
                    2949 *    * KSYLDC - BEGINNING OF LETTER-DIGIT VARIABLE SYMBOL TABLE      *
                    2950 *    * KSYNAC - BEGINNING OF ARITHMETIC ARRAY SYMBOL TABLE          *
                    2951 *    * KSYCVC - BEGINNING CHARACTER VARIABLE SYMBOL TABLE           *
                    2952 *    * KSYCAC - BEGINNING CHARACTER ARRAY SYMBOL TABLE              *
                    2953 *    * KSYBFR - 64-BYTE OUTPUT BUFFER                               *
                    2954 *    * GRLINE - SAVE AREA FOR LINE NUMBER                           *
                    2955 *    * GRTYPE - SAVE AREA FOR TYPE CODE                             *
                    2956 *    * GRTEXT - SAVE AREA FOR FILE LINE TEXT                        *
```

```
                              2957 *    * DLIBUF - BUFFER FOR DLPRNT                              *
                              2958 *                                                             *
                              2959 *ATTRIBUTES                                                   *
                              2960 *    RELOCATABLE                                              *
                              2961 *                                                             *
                              2962 *CHARACTER CODE DEPENDENCY                                    *
                              2963 *    NONE                                                     *
                              2964 *                                                             *
                              2965 *NOTES                                                        *
                              2966 *    ERROR PROCEDURES                                         *
                              2967 *      IF KSYMBL DETECTS A SYNTAX ERROR FOR INVALID DELIMITER, TOO   *
                              2968 *      MANY PARAMETERS, OR IF SCKOUT FINDS AN ERROR, THE ERROR CODE  *
                              2969 *      IS PLACED $CAERR AND EXIT IS TO $CAERK.  NON-SYNTAX ERROR ARE *
                              2970 *      FOR THE OUTPUT DEVICE NOT BEING READY OR FOR NO VARIABLES     *
                              2971 *      BEING FOUND IN THE WORK FILE. EXIT HERE IS ALSO TO SCAERK WITH *
                              2972 *      THE ERROR CODE IN $CAERR                                *
                              2973 *                                                             *
                              2974 *    REGISTER USAGE                                           *
                              2975 *      * REGISTER 1 (@BR) IS USED AS A POINTER WITHIN THE CHARACTER  *
                              2976 *       TABLE.                                                *
                              2977 *      * REGISTER 2 (@XR) IS USED INITIALLY TO POINT WITHIN THE      *
                              2978 *       BUFFER AND LATER TO THE FILE TEXT (GRTEXT)            *
                              2979 *                                                             *
                              2980 *    SAVED/RESTORED AREAS                                     *
                              2981 *      NONE                                                   *
                              2982 *                                                             *
                              2983 *    MODIFICATION CONSIDERATIONS                              *
                              2984 *      NONE                                                   *
                              2985 *                                                             *
                              2986 *    REQUIRED MODULES                                         *
                              2987 *      * @SYSEQ - COMMON SYSTEM EQUATES                       *
                              2988 *      * @FXDEQ - FIXED CORE LOCATIONS IN NUCLEUS EQUATES     *
                              2989 *      * @HDWEQ - HARDWARE EQUATES                            *
                              2990 *      * @CANEQ - CORE LOCATIONS (FIXED) OUTSIDE NUCLEUS EQUATES   *
                              2991 *      * @ERMEQ - ERROR MESSAGE EQUATES (SELECTED ERROR CODES)     *
                              2992 *      * $B@EOQ - BASIC COMPILER PARAMETER AND SYSTEM EQUATES      *
                              2993 *      * SCANIT - DELIMITER SCAN MODULE                       *
                              2994 *      * SCKOUT - CHECK O/P DEVICE MODULE                     *
                              2995 *      * SVARAB - MODULE TO FIND VARIABLES IN A FILE LINE     *
                              2996 *      * GRABIT - MODULE TO RETRIEVE FILE LINES               *
                              2997 *      * DL4ICS - 4-TRACK LOGICAL DISK IOCR                   *
                              2998 *                                                             *
                              2999 *    OTHER                                                    *
                              3000 *      NONE                                                   *
                              3001 **************************************************************************
```

```
ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT           VER 15, MOD 00  04/07/22  PAGE   6

                        05FF 3003 KSYMBL EQU   *                        SYMBOLS KEYWORD MODULE
                             3004 *        HDR   #KSYMB,1                GENERATE PROGRAM HEADER
                             3005 ************************************************************************
                             3006 * PROGRAM HEADER FOR DISK LOAD                                        *
                             3007 ************************************************************************
                             3008 *#$KSYM EQU   X'0600'                  DISK ADDR OF #KSYMB
                             3009 *#$$KSY EQU   X'0C00'                  CORE LOAD ADDRESS OF #KSYMB
                             3010 *#$@KSY EQU   015                      SECTOR CNT OF #KSYMB
      0C00                   3011        ORG   #$$KSY                    CORE LOAD ADDRESS
                        0C00 3012 $$$$$$ EQU   *                        FIRST LOCATION IN PROGRAM
      0C00 7BD2E2E8D4C2  0C05 3013        DC    CL6'#KSYMB'              PROGRAM NAME
      0C06 27            0C06 3014        DC    IL1'039'                 PROGRAM NUMBER OF #KSYMB
                        0C07 3015 #KSYM  EQU   *                        ENTRY POINT TO PROGRAM
                             3016 *** END OF EXPANSION ***

      0C07 3C 00 1B95        3018        MVI   KSYCAC+KSYTB0,KSYCLR      CLEAR SYMBOL TABLE
      0C0B 0C FF 1B94 1B95   3019        MVC   KSYCAC+KSYTB0-1(KSYSCT),KSYCAC+KSYTB0
      0C11 0C 94 1A94 1A95   3020        MVC   KSYLVC+KSYREM-1(KSYREM),KSYLVC+KSYREM
                             3021 *
                             3022 *            SYNTAX-CHECK INPUT LINE AND SET O/P DEVICE
                             3023 *
      0C17 35 02 03C7        3024        L     $XRSAV,@XR               POINT XR TO BYTE AFTER KEYWORD
      0C1B BD 60 00          3025        CLI   KSY000(,@XR),KSYDSH      XR REF A DASH ?
      0C1E C0 81 0E49        3026        BE    KSY900                   YES, 'INV DELIM' ERROR
                             3027 *
      0C22 C0 87 0E99        3028        B     SCANIT                   BYPASS BLANKS
      0C26 BD 1E 00          3029        CLI   KSY000(,@XR),@EOS        IS XR POINTING TO EOS ?
      0C29 F2 81 13          3030        JE    KSY100                   IS EOS, SKIP SYNTAX CHECKS
                             3031 *
      0C2C C0 87 0EDA        3032        B     SCKOUT                   CHECK OUTPUT SPEC
      0C30 3C 11 03CD        3033        MVI   $CAERR,@@E131            SET 'INV PARAM' ERR CODE
      0C34 C0 04 0E5B        3034        BNH   KSY999                   IF ERR IN SCKOUT, CALL ERR PROG
                             3035 *
      0C38 BD 1E 00          3036        CLI   KSY000(,@XR),@EOS        XR REF AN EOS ?
      0C3B C0 01 0E57        3037        BNE   KSY950                   NO, 'TOO MANY PARAMS' ERROR
```

```
                             3039 *
                             3040 *               SYNTAX OK -- BEGIN PROCESSING
                             3041 *
     0C3F C0 87 0F7E         3042 KSY100 B     SCKDEV                    CHECK OUTPUT DEVICE
     0C43 C0 87 14F2         3043        B     GRABIT                    PRIME GRABIT BUFFERS
                             3044 *
     0C47 3C 3F 0E96         3045        MVI   KSYPPL+KSYCNT,KSY063      INITLZ PRINT COUNT
     0C4B 3C 01 1677         3046        MVI   GRWHAT,GRAEFR             SET RETURN TEXT CODE FOR GRABIT
     0C4F C0 87 14F2         3047 KSY150 B     GRABIT                    RETRIEVE ONE FILE STATEMENT
                             3048 *
     0C53 3D 1C 1B98         3049        CLI   GRTEXT,@EOF               IS THIS THE EOF LINE ?
     0C57 C0 81 0D01         3050        BE    KSY800                    YES, GET OUT OF LOOP
                             3051 *
     0C5B C2 02 1B98         3052 KSY200 LA    GRTEXT,@XR                POINT XR TO FIRST BYTE OF TEXT
                             3053 *
     0C5F C2 01 0E67         3054 KSY250 LA    KSYALP,@BR                POINT BR TO FIRST BYTE OF
                             3055 *                                     * ALPHABET TABLE
     0C63 C0 87 0FDF         3056        B     SVARAB                    CALL VARIABLE SCAN ROUTINE
                             3057 *
     0C67 BD 1E 00           3058        CLI   KSY000(,@XR),@EOS         WAS AN EOS FOUND ?
     0C6A C0 81 0C4F         3059        BE    KSY150                    YES, GO TO END OF LOOP
                             3060 *
     0C6E 9D 00 00 00        3061 KSY300 CLC   KSY000(,@XR),KSY000(KSYLN1,@BR) DOES XR PT TO THIS CHAR
                             3062 *                                     * IN ALPHA CHAR TABLE ?
     0C72 F2 81 07           3063        JE    KSY350                    YES, BR CONTAINS CHAR DISP
                             3064 *
     0C75 D2 01 01           3065        LA    KSY001(,@BR),@BR          NO,TRY NEXT ENTRY IN ALPHA TBL
     0C78 C0 87 0C6E         3066        B     KSY300                    BRANCH TO TEST IT
                             3067 *
     0C7C 36 01 0E89         3068 KSY350 A     KSYFBR,@BR                CALCULATE LTR DISP IN BR
     0C80 38 01 1400         3069        TBN   SVAVTC,SVALVC             IS IT A LETTER VARIABLE REF ?
     0C84 F2 90 07           3070        JF    KSY400                    NO, TEST LETTER DIGIT VAR REF
                             3071 *
     0C87 36 01 1943         3072        A     KSYASL,@BR                POINT BR TO ADDR IN SYMBOL TBL
     0C8B F2 87 58           3073        J     KSY700                    JUMP TO SET SW INDR AT THE ADDR
                             3074 *
     0C8E 38 10 1400         3075 KSY400 TBN   SVAVTC,SVALDC             IS IT A LETTER-DIGIT VAR REF ?
     0C92 F2 90 31           3076        JF    KSY550                    NO, TEST FOR ARITH ARRAY
                             3077 *
     0C95 34 02 0CC2         3078        ST    KSY525+@OP1,@XR           SAVE XR
     0C99 E2 02 01           3079        LA    KSY001(,@XR),@XR          INCR XR PAST ALPHA CHAR
     0C9C C0 87 0E99         3080        B     SCANIT                    PT XR TO DIGIT
     0CA0 0C 01 0E85 1945    3081        MVC   KSYDIG(@CADDR),KSYALD     SET KSYDIG=ADDR OF LETTER-DIGIT
                             3082 *                                     * PORTION OF SYMBOL TABLE
     0CA6 BD F0 00           3083 KSY450 CLI   KSY000(,@XR),KSYXFO       HAS ADDR BEEN CALCULATED ?
     0CA9 F2 81 0F           3084        JE    KSY500                    YES, GET OUT OF LOOP
                             3085 *
     0CAC 0E 01 0E85 0E62    3086        ALC   KSYDIG(@CADDR),KSYNXT     INCR TO NEXT TBL
     0CB2 8F 00 00 0E90      3087        SLC   KSY000(,@XR),KSYONE(KSYLN1) DECR VALUE XR IS POINTING TO
     0CB7 C0 87 0CA6         3088        B     KSY450                    CONTINUE LOOP

     0CBB 36 01 0E85         3090 KSY500 A     KSYDIG,@BR                PUT CORRECT DISP IN BR
     0CBF C2 02 0000         3091 KSY525 LA    *-*,@XR                   RESTORE XR
     0CC3 F2 87 20           3092        J     KSY700                    JUMP TO'SET INDR
                             3093 *
     0CC6 38 08 1400         3094 KSY550 TBN   SVAVTC,SVANAC             IS IT AN ARITHMETIC ARRAY REF
```

```
     0CCA F2 90 07              3095       JF    KSY600                    NO, TEST FOR CHAR VAR REF
                               3096 *
     0CCD 36 01 1947           3097       A     KSYAAA,@BR                POINT BR TO ADDR IN SYMBOL TBL
     0CD1 F2 87 12             3098       J     KSY700                    JUMP TO SET INDR
                               3099 *
     0CD4 38 04 1400           3100 KSY600 TBN  SVAVTC,SVACVC             IS IT A CHARACTER VAR REF ?
     0CD8 F2 90 07             3101       JF    KSY650                    NO, PROCESS CHAR ARRAY REF
                               3102 *
     0CDB 36 01 1949           3103       A     KSYACV,@BR                POINT BR TO ADDR IN SYMBOL TBL
     0CDF F2 87 04             3104       J     KSY700                    JUMP TO SET INDR
     0CE2 36 01 194B           3105 KSY650 A    KSYACA,@BR                POINT BR TO ADDR IN SYMBOL TBL
                               3106 *
     0CE6 7A 01 00             3107 KSY700 SBN  KSY000(,@BR),KSYREF       SET INDR -- VAR WAS REFERENCED
                               3108 *
     0CE9 0C 00 0E8B 1401      3109       MVC   KSYTMP(KSYLN1),SVALNG     EXPAND LENGTH TO TWO BYTES
     0CEF 36 02 0E8B           3110       A     KSYTMP,@XR                INCR XR TO NEXT NON-BLANK
                               3111 *
     0CF3 38 80 1B96           3112       TBN   GRTYPE,KSYDIS             IS LINE DISABLED ?
     0CF7 F2 90 03             3113       JF    KSY750                    NO, SKIP SETTING INDR
                               3114 *
     0CFA 7A 80 00             3115       SBN   KSY000(,@BR),KSYDIS       YES, SET INDR IN SYMBOL TABLE
                               3116 *
     0CFD C0 87 0C5F           3117 KSY750 B    KSY250                    REPEAT LOOP TO SET INDICATOR
                               3118 *                                    * IN SYMBOL TABLE
                               3119 *
                               3120 *           PRINT LTR AND LTR-DGT VARIABLES
                               3121 *
     0D01 3C 40 1BD6           3122 KSY800 MVI  KSYBFR+KSY062,KSYBLA      CLEAR O/P BFR TO ALL BLANKS
     0D05 0C 3D 1BD5 1BD6      3123       MVC   KSYBFR+KSY062-1(KSY062),KSYBFR+KSY062
                               3124 *
     0D0B C2 01 1B98           3125       LA    KSYBFR,@BR                LOAD BR WITH ADDR OF O/P BFR
     0D0F 0C 01 0E87 1943      3126 KSY805 MVC  KSYTAB(@CADDR),KSYASL     INITLZ TBL ADDR TO LTR VAR TBL
     0D15 3C EF 0E8E           3127       MVI   KSYDGT,KSYXEF             SET DIGIT BYTE TO 'EF'
                               3128 *
     0D19 35 02 0E87           3129 KSY810 L    KSYTAB,@XR                POINT XR TO TBL ADM
     0D1D 36 02 0E8D           3130       A     KSYDSP,@XR                ADD CHAR DISP TO TEL ADDR
                               3131 *
     0D21 B8 01 00             3132       TBN   KSY000(,@XR),KSYREF       WAS VAR AT THIS ADDR REFERENCED
     0D24 F2 90 1E             3133       JF    KSY840                    NO, CHECK NEXT VARIABLE
                               3134 *
     0D27 4C 00 01 0E8F        3135       MVC   KSY001(,@BR),KSYCHR(KSYLN1)  MOVE LTR CODE TO O/P BFR
                               3136 *
     0D2C B8 80 00             3137       TBN   KSY000(,@XR),KSYDIS       WAS THIS VAR IN A DISABLED LN ?
     0D2F F2 90 03             3138       JF    KSY820                    NO, JUMP TO CHECK FOR LTR-DGT
                               3139 *
     0D32 7C 5C 00             3140       MVI   KSY000(,@BR),KSYAST       ELSE, MOVE '*' TO O/P BFR
                               3141 *
     0D35 3D EF 0E8E           3142 KSY820 CLI  KSYDGT,KSYXEF             IS THIS A LETTER VARIABLE
     0D39 F2 81 05             3143       JE    KSY830                    YES, BRANCH TO PRINT RTN
                               3144 *
     0D3C 4C 00 02 0E8E        3145       MVC   KSY002(,@BR),KSYDGT(KSYLN1)  ELSE, MOVE DIGIT TO O/P BFR
     0D41 C0 87 0E1F           3146 KSY830 B    KSYPRN                    BRANCH TO PRINT ROUTINE
     0D45 0E 01 0E87 0E62      3147 KSY840 ALC  KSYTAB(@CADDR),KSYNXT     INCR PTR TO NEXT TABLE ADDR
     0D4B 0E 00 0E8E 0E90      3148       ALC   KSYDGT(KSYLN1),KSYONE     INCR DGT BY '1'
     0D51 3D F9 0E8E           3149       CLI   KSYDGT,KSYXF9             IS PTR PAST LAST LTR/DIG TABLE ?
     0D55 C0 04 0D19           3150       BNH   KSY810                    NO, TEST FOR THIS VAR REFERENCED
```

```
                              3151 *
     0D59 0E 00 0E8D 0E90     3152        ALC   KSYDSP(KSYLN1),KSYONE    INCR DISP WITHIN TABLES BY ONE
                              3153 *
     0D5F C2 02 0E67          3154        LA    KSYALP,@XR               PUT ADDR OF ALPHA TBL IN XR
     0D63 36 02 0E8D          3155        A     KSYDSP,@XR               ADDR DISP TO GET CORRECT REF
     0D67 2C 00 0E8F 00       3156        MVC   KSYCHR(KSYLN1),KSY000(,@XR)  MOVE EBCDIC CODE TO KSYCHR
                              3157 *
     0D6C 3D 1C 0E8D          3158        CLI   KSYDSP,KSYTB0            HAVE ALL ARITH VAR BEEN TESTED ?
     0D70 C0 04 0D0F          3159        BNH   KSY805                   NO, CONTINUE TESTING
```

```
                                   3161 *
                                   3162 *              PRINT ARITH ARRAYS, CHAR VAR, AND CHAR ARRAYS
                                   3163 *
    0D74 35 02 1947                3164        L     KSYAAA,@XR              PT XR TO ARITH ARRAYS TBL
    0D78 3C 00 0E8D                3165 KSY845 MVI   KSYDSP,KSY000           INITLZ DISP TO ZERO
    0D7C B8 01 00                  3166 KSY847 TBN   KSY000(,@XR),KSYREF     WAS VAR AT ADDR IN XR REF ?
    0D7F F2 90 26                  3167        JF    KSY865                  NO, CHECK NEXT VAR
                                   3168 *
    0D82 B8 80 00                  3169        TBN   KSY000(,@XR),KSYDIS     IS VAR IN A DISABLED LINE ?
    0D85 F2 90 03                  3170        JF    KSY855                  NO, JUMP AROUND MOVING      IN
                                   3171 *
    0D88 7C 5C 00                  3172 KSY850 MVI   KSY000(,@BR),KSYAST     ELSE, MOVE IN '*' TO IND THIS
                                   3173 *
    0D8B 34 02 0DA3                3174 KSY855 ST    KSY860+@OP1,@XR         SAVE XR
                                   3175 *
    0D8F C2 02 0E67                3176        LA    KSYALP,@XR              PT XR TO ALPHA TABLE
    0D93 36 02 0E8D                3177        A     KSYDSP,@XR              ADD CHAR DISP TO XR
    0D97 6C 00 01 00               3178        MVC   KSY001(KSYLN1,@BR),KSY000(,@XR)  MOVE CHAR TO O/P BFR
                                   3179 *
    0D9B 4C 02 04 0E94             3180 KSY858 MVC   KSY004+*-*(,@BR),KSYRAY+*-*(KSYLRY+*-*)  MOVE '$' , '(*)',
                                   3181 *                                   * OR '$(*)' TO O/P
    0DA0 C2 02 0000                3182 KSY860 LA    *-*,@XR                 RESTORE XR
    0DA4 C0 87 0E1F                3183        B     KSYPRN                  BRANCH TO PRINT ROUTINE
                                   3184 *
    0DA8 E2 02 01                  3185 KSY865 LA    KSY001(,@XR),@XR        INCR TO PT TO NEXT ENTRY
    0DAB 0E 00 0E8D 0E90           3186        ALC   KSYDSP(KSYLN1),KSYONE   INCR DISP WITHIN TABLES BY ONE
                                   3187 *
    0DB1 3D 1C 0E8D                3188        CLI   KSYDSP,KSYTB0           IS THIS TABLE FINISHED ?
    0DB5 C0 04 0D7C                3189        BNH   KSY847                  NO, CONTINUE
                                   3190 *
    0DB9 F2 80 16                  3191 KSY870 JC    KSY875,@NOP             NOP IF CHAR VAR TBL NEXT, UCB
                                   3192 *                                   * IF CHAR ARRAY NEXT
    0DBC 3C 02 0D99                3193        MVI   KSY858-@D1,KSY002       SET UP DISPLACEMENTS FOR
    0DC0 3C 00 0D9C                3194        MVI   KSY858+@Q,KSY000        * SIMPLE CHAR VAR IN O/P BFR
    0DC4 0F 01 0D9F 0E66           3195        SLC   KSY858+@DOP2(@CADDR),KSYTHR
                                   3196 *
    0DCA 3C 87 0DBA                3197        MVI   KSY870+@Q,@UCB          SET SW TO PROCESS CHAR ARRAYS
    0DCE C0 87 0D78                3198        B     KSY845                  BRANCH TO TEST CHAR VARIABLES
                                   3199 *
    0DD2 F2 80 16                  3200 KSY875 JC    KSY880,@NOP             NOP IF CHAR ARRAY TBL NEXT. UCB
                                   3201 *                                   * IF ALL VARIABLES WERE TESTED
    0DD5 3C 05 0D9D                3202        MVI   KSY858+@D1,KSY005       SET UP DISPLACEMENTS FOR
    0DD9 3C 03 0D9C                3203        MVI   KSY858+@Q,KSYLRY        * SIMPLE CHAR VAR IN O/P BFR
    0DDD 0E 01 0D9F 0E66           3204        ALC   KSY858+@DOP2(@CADDR),KSYTHR
                                   3205 *
    0DE3 3C 87 0DD3                3206        MVI   KSY875+@Q,@UCB          SET SW TO GET OUT OF LOOP
    0DE7 C0 87 0D78                3207        B     KSY845                  BRANCH TO CONTINUE TESTING
                                   3208 *
    0DEB 3D 40 1B99                3209 KSY880 CLI   KSYBFR+KSY001,KSYBLA    DOES O/P BFR CONTAIN ANYTHING ?
    0DEF F2 81 20                  3210        JE    KSY885                  NO, JUMP TO CALL PRINT RTN LAST
                                   3211 *
    0DF2 3C 01 0E2F                3212        MVI   KSY887+@Q,@BNE          SET INDR TO PRINT UNFILLED LINE
                                   3213 *
    0DF6 34 01 0E8B                3214        ST    KSYTMP,@BR              SAVE PTR TO O/P BFR
    0DFA C2 02 1B98                3215        LA    KSYBFR,@XR              GET START ADDR OF O/P BFR
    0DFE 34 02 0E8D                3216        ST    KSYDSP,@XR              SAVE START ADDR OF O/P BFR
```

```
     0E02 0F 01 0E8B 0E8D        3217        SLC   KSYTMP(@CADDR),KSYDSP     GET LENGTH OF O/P BFR
     0E08 0C 00 0E96 0E8B        3218        MVC   KSYPPL+KSYCNT(KSYLN1),KSYTMP  STORE LENGTH IN PPL
     0E0E C0 87 0E1F             3219        B     KSYPRN                   PRINT THIS LINE
                                 3220 *
     0E12 F2 87 3B               3221 KSY885 JC    KSY925,@UCB              UCB UNLESS THERE WAS A VAR REF
     0E15 C0 87 174C             3222        B     DLPRNT                   WAIT FOR LAST LINE
     0E19 057F            0E1A   3223        DC    AL(@CADDR)($WAITF)       * OF PRINTER O/P
                                 3224 *
     0E1B C0 87 04A1             3225        B     $CARPL                   EXIT
                                 3226 *
     0E1F 34 08 0E48             3227 KSYPRN ST    KSY899+@OP1,@ARR         SAVE RETURN ADDRESS
     0E23 3C 80 0E13             3228        MVI   KSY885+@Q,@NOP           SET SW TO IND A VAR WAS REF
     0E27 D2 01 07               3229        LA    KSY007(,@BR),@BR         PT BR TO NEXT O/P ENTRY ADDR
     0E2A 3D 40 1BD1             3230        CLI   KSYBFR+KSY057,KSYBLA     IS OUTPUT LINE FILLED ?
     0E2E F2 81 14               3231 KSY887 JC    KSY899,@BE               NO, RETURN TO PT WHERE CALLED
                                 3232 *
     0E31 C0 87 174C             3233        B     DLPRNT                   ELSE, PRINT OUT ONE LINE
     0E35 0E95            0E36   3234        DC    AL(@CADDR)(KSYPPL)       * USING KSYPPL AS PPL
     0E37 3C 40 1BD6             3235        MVI   KSYBFR+KSY062,KSYBLA     CLEAR O/P BFR TO
     0E3B 0C 3D 1BD5 1BD6        3236        MVC   KSYBFR+KSY062-1(KSY062),KSYBFR+KSY062  ALL BLANKS
     0E41 C2 01 1B98             3237        LA    KSYBFR,@BR               POINT BR TO FIRST OF O/P BFR
     0E45 C0 87 0000             3238 KSY899 B     *-*                      RETURN TO PT WHERE CALLED
                                 3239 *
     0E49 3C 18 03CD             3240 KSY900 MVI   $CAERR,@@E139            SET 'INV DELIM' ERR CODE
     0E4D F2 87 0B               3241        J     KSY999                   JUMP TO CALL ERR PROG
                                 3242 *
     0E50 3C 41 03CD             3243 KSY925 MVI   $CAERR,@@E255            SET ERR CODE FOR NO VAR IN PROG
     0E54 F2 87 04               3244        J     KSY999                   CALL ERROR PROGRAM
                                 3245 *
     0E57 3C 12 03CD             3246 KSY950 MVI   $CAERR,@@E133            SET 'TOO MANY PARAMS' ERR CODE
                                 3247 *
     0E5B C0 87 0469             3248 KSY999 B     $CAERK                   CALL ERROR PROGRAM
```

```
                              3250 *
                              3251 *              CONSTANTS
                              3252 *
     0E5F FFFF                0E60 3253 KSYMIN DC     XL2'FFFF'                  CONSTANT OF NEGATIVE ONE
     0E61 001D                0E62 3254 KSYNXT DC     XL2'001D'                  DISP BETWEEN TABLES
     0E63 0000                0E64 3255 KSYZER DC     XL2'0000'                  INITIAL LN NO. FOR GFINDN
     0E65 0003                0E66 3256 KSYTHR DC     XL(@CADDR)'0003'           CONSTANT TO PT ADDR TO ')' OF
                              3257 *                                            * 1(*)' DC
                              0E67 3258 KSYALP EQU    *                          START OF ALPHABET TABLE
     0E67 5B7B7CC1C2C3C4C5    0E83 3259        DC     CL29'$#@ABCDEFGHIJKLMNOPQRSTUVWXYZ'  *** TABLE ***
                              0000 3260 KSY000 EQU    0                          ZERO DISPLACEMENT
     0E84                     0E85 3261 KSYDIG DS     CL2                        USED FOR LETTER-DIGIT VAR REF
     0E86                     0E87 3262 KSYTAB DS     CL(@CADDR)                 HOLD AREA FOR TABLE ADDRESS
                              F199 3263 KSYDUM EQU    $$ZERO-KSYALP
     0E88 F199                0E89 3264 KSYFBR DC     AL(@CADDR)(KSYDUM)
                              0E8A 3265 KSYTMI EQU    *                          TWO-BYTE TEMPORARY SAVE AREA,
     0E8A                     0E8B 3266 KSYTMP DS     XL(@CADDR)                 * INITIALIZED TO ZERO.
     0E8A                     3267        ORG    KSYTMI                          * LEFT BYTE REMAINS ZERO
     0E8A 0000                0E8B 3268        DC     XL(@CADDR)'0'              * THROUGHOUT PROCESSING.
                              0E8C 3269 KSYDSI EQU    *                          SAVE AREA FOR DISPLACEMENT
     0E8C                     0E8D 3270 KSYDSP DS     XL(@CADDR)                 * WITHIN TABLE OF THE 29
     0E8C                     3271        ORG    KSYDSI                          * ALPHABETIC CHARACTERS.
     0E8C 0000                0E8D 3272        DC     XL2'0000'                  * INITIALIZED TO ZERO
     0E8E                     0E8E 3273 KSYDGT DS     XL1                        SAVE AREA FOR EBCDIC DIGIT
     0E8F                     0E8F 3274 KSYCHR DS     CL1                        SAVE AREA FOR ALPHA CHAR
     0E8F                     3275        ORG    KSYCHR
     0E8F 5B                  0E8F 3276        DC     CL1'$'                     * INITIALIZED TO '$'
     0E90 01                  0E90 3277 KSYONE DC     XL1'01'                    CONSTANT OF ONE
     0E91 5B4D5C5D            0E94 3278 KSYRAY DC     CL4'$(*)'                   STRING TO SHOW CHAR ARRAY
                              0001 3279 KSY001 EQU    1                          INCREMENT OF '1'
                              003F 3280 KSY063 EQU    63                         LENGTH OF PRINT LINE
                              0003 3281 KSYLRY EQU    3                          LENGTH OF *(A)'
                              005C 3282 KSYAST EQU    C'*'                       ASTERISK
                              00F9 3283 KSYXF9 EQU    X'F9'                      HIGHEST EVCDIC INTEGER
                              00EF 3284 KSYXEF EQU    X'EF'                      ONE BELOW LOWEST EBCDIC INTEGER
                              0040 3285 KSYBLA EQU    X'40'                      BLANK
                              0039 3286 KSY057 EQU    57                         DISP IN O/P BFR TO LAST FLD-1
                              003E 3287 KSY062 EQU    62                         INCR TO LAST ENTRY IN O/P IN

                              3289 *KSYPPL PPL    FUNC-@PRETR,CADDR-KSYBFR
                              0E95 3290 KSYPPL EQU    *                          PPL ADDRESS
     0E95 C0                  0E95 3291        DC     AL1(@PRETR)                FUNCTION REQUESTED
     0E96 00                  0E96 3292        DC     AL1(*-*)                   PRINT COUNT
     0E97 1B98                0E98 3293        DC     AL2(KSYBFR)                DATA ADDRESS
                              3294 *** END OF EXPANSION ***

                              0002 3296 KSY002 EQU    2                          DISP OF '2' IN O/P BFR FIELD
                              0003 3297 KSY003 EQU    3                          DISP OF '3' IN O/P BFR FIELD
                              0004 3298 KSY004 EQU    4                          DISP OF '4' IN O/P BFR FIELD
                              0005 3299 KSY005 EQU    5                          DISP OF '5' IN O/P BFR FIELD
                              0006 3300 KSY006 EQU    6                          DISP OF '6' IN O/P BFR FIELD
                              0007 3301 KSY007 EQU    7                          DISP OF '7' IN O/P BFR FIELD
                              00F0 3302 KSYXFO EQU    X'F0'                      LOWEST EBCDIC CODE FOR A NUMBER
                              0002 3303 KSYREG EQU    2                          REGISTER LENGTH OF 2 BYTES
                              0001 3304 KSYREF EQU    X'01'                      BIT INDR MASK FOR VAR REF
                              0080 3305 KSYDIS EQU    X'80'                      BIT INDR MASK FUR DISABLED LN
```

```
       #KSYMB - SYMBOL KEYWORD MODULE
```

```
                         0001 3306 KSYLN1 EQU   1                        LENGTH OF ONE CHARACTER
                         0002 3307 KSYLN2 EQU   2                        LENGTH OF 2-BYTE LINE NO.
                         0195 3308 KSYLNG EQU   29*14-1                  LENGTH OF SYMBOL TABLE - '1'
                         0060 3309 KSYDSH EQU   C'-'                     DASH CHARACTER
                         0000 3310 KSYCLR EQU   0                        TO CLEAR SYMBOL TABLE TO ZEROS
                         0100 3311 KSYSCT EQU   256                      LENGTH OF ONE SECTOR
                         0095 3312 KSYREM EQU   KSYLNG-KSYSCT            LENGTH OVER 1 SCTR OF SYM TBL
                         001C 3313 KSYTB0 EQU   29-1                     SINGLE TABLE LENGTH - '1'
                         0469 3314 SCKERR EQU   $CAERK                   ERROR EXIT FROM SCKOUT
                         0001 3315 KSYCNT EQU   1                        DISP IN PPL TO PRINT COUNT
                              3316 *
                              3317 *       $CANI
```

SCANIT - DELIMETER SCAN MODULE

```
                              3319+****************************************************************************
                              3320+*  5703-XM1    COPYRIGHT IBM CORP. 1970                                     *
                              3321+*                   REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083      *
                              3322+*                                                                           *
                              3323+****************************************************************************
                              3324+*STATUS                                                                     *
                              3325+*    VERSION 1 MODIFICATION 0                                               *
                              3326+*                                                                           *
                              3327+*FUNCTION                                                                   *
                              3328+*    THE FUNCTION OF SCANIT IS TO SCAN PAST VALID DELIMITERS AND            *
                              3329+*    RETURN A POINTER TO THE FIRST CHARACTER THAT'S NOT A DELIMITER.        *
                              3330+*                                                                           *
                              3331+*ENTRY POINTS                                                               *
                              3332+*    * THE ENTRY POINT IS SCANIT.                                           *
                              3333+*    * THE CALLING SEQUENCE IS AS FOLLOWS:                                  *
                              3334+*             B     SCANIT                                                  *
                              3335+*      WITH REGISTER 2 (@XR) POINTING TO THE FIRST CHARACTER TO BE          *
                              3336+*      EXAMINED.                                                            *
                              3337+*                                                                           *
                              3338+*INPUT                                                                      *
                              3339+*    NONE                                                                   *
                              3340+*                                                                           *
                              3341+*OUTPUT                                                                     *
                              3342+*    NONE                                                                   *
                              3343+*                                                                           *
                              3344+*EXTERNAL REFERENCES                                                        *
                              3345+*    $CAERR - ERROR CODE SAVE AREA                                          *
                              3346+*                                                                           *
                              3347+*EXITS, NORMAL                                                              *
                              3348+*    NORMAL EXIT FROM SCANIT IS TO THE BYTE FOLLOWING THE BRANCH TO         *
                              3349+*    SCANIT IN THE CALLING ROUTINE. THE PSR (REGISTER 4) WILL CONTAIN       *
                              3350+*    A ZERO IF NO DELIMITERS WERE FOUND OR A HIGH CONDITION IF ONE OR       *
                              3351+*    MORE DELIMITERS WERE SCANNED.                                          *
                              3352+*                                                                           *
                              3353+*EXITS, ERROR                                                               *
                              3354+*    ERROR EXIT FROM SCANIT IS TO THE BYTE FOLLOWING THE BRANCH TO          *
                              3355+*    SCANIT IN THE CALLING ROUTINE. THE PSR WILL CONTAIN A LOW              *
                              3356+*    CONDITION.                                                             *
                              3357+*                                                                           *
                              3358+*TABLES/WORKAREAS                                                           *
                              3359+*    * SCACNT - AREA CONTAINING NUMBERS OF DELIMITERS SCANNED               *
                              3360+*    * SCAMMA - LOC WHERE SCACOM MAY BE MOVED IF ONE COMMA IS ALSO          *
                              3361+*      TO BE CONSIDERED A DELIMITER. MOVING SCACOF BACK INTO SCAMMA         *
                              3362+*      INDICATES THAT ONLY BLANKS SHOULD BE CONSIDERED DELIMITERS.          *
                              3363+*                                                                           *
                              3364+*ATTRIBUTES                                                                 *
                              3365+*    RELOCATABLE AND RE-USABLE                                              *
                              3366+*                                                                           *
                              3367+*CHARACTER CODE DEPENDENCY                                                  *
                              3368+*    THE OPERATION OF THIS MODULE DOES NOT DEPEND UPON A PARTICULAR         *
                              3369+*    INTERNAL REPRESENTATION OF THE EXTERNAL CHARACTER SET.                 *
                              3370+*                                                                           *
                              3371+*NOTES                                                                      *
                              3372+*    ERROR PROCEDURES                                                       *
                              3373+*        THE ONLY ERROR CONDITION DETECTED BY SCANIT IS THE CASE WHERE      *
                              3374+*        A CARRIAGE-RETURN CODE FOLLOWS A COMMA.  UPON RETURN TO THE        *
```

```
                              3375+*        CALLING ROUTINE, @PSR WILL BE SET TO A LOW CONDITION, THE     *
                              3376+*        ERROR CODE IS SET IN $CAERR, AND MG WILU BE POINTING TO THE   *
                              3377+*        CARRIAGE-RETURN CHARACTER.                                    *
                              3378+*                                                                      *
                              3379+*    REGISTER USAGE                                                    *
                              3380+*       REGISTER 2 (@XR) IS USED AS A POINTER ACROSS THE AREA BEING    *
                              3381+*       SCANNED FOR DELIMETERS.                                        *
                              3382+*                                                                      *
                              3383+*    SAVED/RESTORED AREAS                                              *
                              3384+*       UPON ENTRY TO SCANIT, REGISTER 8 (@ARR) IS SAVED AND USED AS   *
                              3385+*       THE RETURN ADDRESS.                                            *
                              3386+*                                                                      *
                              3387+*    MODIFICATION CONSIDERATIONS                                       *
                              3388+*       NONE                                                           *
                              3389+*                                                                      *
                              3390+*    REQUIRED MODULES                                                  *
                              3391+*       * @SYSEQ - COMMON SYSTEM EQUATES                               *
                              3392+*       * @FXDEQ - FIXED NUCLEUS ADDRESSES EQUATES                     *
                              3393+*                                                                      *
                              3394+*    OTHER                                                             *
                              3395+*       SCANIT IS INITIALIZED TO BYPASS BLANKS ONLY. IF SCACOM IS      *
                              3396+*       MOVED TO SCAMMA, ONE COMMA WILL BE SCANNED ALONG WITH BLANKS.  *
                              3397+*       THE INSTRUCTION TO DO THIS IS AS FOLLOWS:                      *
                              3398+*             MVI   SCAMMA,SCACOM                                       *
                              3399+*                                                                      *
                              3400+*       TO DROP THE COMMA FROM ITS DELIMITER STATUS, SCACOF SHOULD BE  *
                              3401+*       MOVED TO SCAMMA, USING THE FOLLOWING INSTRUCTION:              *
                              3402+*             MVI   SCAMMA,SCACOF                                       *
                              3403+*                                                                      *
                              3404+*****************************************************************************


                              3406+*
                              3407+*                        EQUATES USED IN THIS SUBROUTINE
                              3408+*
                         0001 3409+SCAINC EQU   1                          TO INCREMENT POINTER
                         0001 3410+SCACOM EQU   @BNE                       SWITCH TO ALLOW SCANNING COMMA
                         0087 3411+SCACOF EQU   @UCB                       SWITCH TO SET OFF THE INDICATON
                              3412+*                                       * FOR SCANNING A COMMA
                         0E99 3413+SCANIT EQU   *                          ENTRY POINT TO THIS SUBROUTINE
  0E99 34 08 0ED5             3414+       ST    SCA500+@OP1,@ARR           SAVE RETURN ADDRESS
  0E9D 34 02 0ED7             3415+       ST    SCASVE,@XR                 SAVE POINTER VALUE
  0EA1 3C 04 03CD             3416+       MVI   $CAERR,@@E110              SET ERROR CODE
  0EA5 F2 87 03               3417+       J     SCA200                     GO TO PROCESS
  0EA8 E2 02 01               3418+SCA100 LA    SCAINC(,@XR),@XR           INCREMENT POINTER TO NEXT CHAR
  0EAB BD 40 00               3419+SCA200 CLI   0(,@XR),@BLANK             IS THIS CHAR BLANK ?
  0EAE C0 81 0EA8             3420+       BE    SCA100                     YES, FETCH NEXT ONE
  0EB2 BD 6B 00               3421+       CLI   0(,@XR),@COMMA             IS IT A COMMA ?
  0EB5 F2 87 10               3422+SCA250 JC    SCA400,@UCB                UCS TO RETURN -- OR NOP IF
                              3423+*                                       * SCAMMA IS ACTIVE AND CHAR
  0EB8 E2 02 01               3424+SCA300 LA    SCAINC(,@XR),@XR           INCREMENT POINTER TO NEXT CHAR
  0EBB BD 40 00               3425+       CLI   0(,@XR),@BLANK             IS THIS CHAR A BLANK ?
  0EBE C0 81 0EB8             3426+       BE    SCA300                     YES, FETCH NEXT ONE
  0EC2 BD 1F 00               3427+       CLI   0(,@XR),@EOS+1             IS THIS EOS ?
  0EC5 F2 82 0A               3428+       JL    SCA500                     IF NOT, SKIP ERROR ROUTINE
  0EC8 34 02 0ED9             3429+SCA400 ST    SCACNT,@XR                 SAVE NEW POINTER VALUE
```

```
     SCANIT - DELIMETER SCAN MODULE


ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT               VER 15, MOD 00  04/07/22  PAGE  16

    0ECC 0F 01 0ED9 0ED7        3430+      SLC    SCACNT(2),SCASVE            SET PSR TO EQUAL IF POINTER
                                3431+*                                       * NOT ADVANCED
    0ED2 C0 87 0000             3432+SCA500 B     *-*                        YES, RETURN
                           0EB6 3433+SCAMMA EQU   SCA250+@Q                  TO SET SCAN COMMA INDICATOR
                                3434+*
                                3435+*                  SAVE AREA
                                3436+*
                           0ED6 3437+SCASV1 EQU   *                          FIRST BYTE OF SCASVE
    0ED6                   0ED7 3438+SCASVE DS    CL2                        ORIGINAL POINTER VALUE SAVE
    0ED8                   0ED9 3439+SCACNT DS    CL2                        SAVE AREA FOR TOTAL CHAR SCAN
                                3440+***                     END OF SCANIT                           ***
                                3441 *
                                3442 *      $CKOU
```

```
                        3444+*************************************************************************
                        3445+* 5703-XM1      COPYRIGHT IBM CORP. 1970                              *
                        3446+*                  REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083 *
                        3447+*                                                                      *
                        3448+*************************************************************************
                        3449+*STATUS                                                                *
                        3450+*    VERSION 1 MODIFICATION 0                                          *
                        3451+*                                                                      *
                        3452+*FUNCTION                                                              *
                        3453+*    SCKOUT, ENTERED AT SCKOUT, WILL CHECK THE NEXT PARAMETER FOR THE  *
                        3454+*    'CRT' OR 'PRINTER' PARAMETER AND SET THE APPROPRIATE INDICATORS   *
                        3455+*    FOR DLPRNT.  SCKOUT, ENTERED AT SCKDEV, WILL TEST THE NUCLEUS     *
                        3456+*    INDICATORS FOR THE SPECIFIED OUTPUT DEVICE AND, IF NO ERRORS ARE  *
                        3457+*    FOUND, WILL RETURN TO THE USER WITH THE APPROPRIATE OUTPUT DEVICE *
                        3458+*    READY.                                                            *
                        3459+*                                                                      *
                        3460+*ENTRY POINTS                                                          *
                        3461+*    SCKOUT HAS THE FOLLOWING TWO ENTRY POINTS:                        *
                        3462+*       * SCKOUT - ENTRY TO CHECK THE NEXT PARAMETER FOR THE 'CRT' OR  *
                        3463+*          'PRINTER' SPECIFICATION                                     *
                        3464+*       * SCKDEV - ENTRY TO CHECK AND MAKE READY THE SPECIFIED OUTPUT  *
                        3465+*          DEVICE.                                                     *
                        3466+*                                                                      *
                        3467+*INPUT                                                                 *
                        3468+*    INPUT TO SCKOUT (ENTRY POINT SCKOUT) IS THE INPUT LINE BUFF WITH  *
                        3469+*    @XR POINTING TO THE FIRST CHARACTER TO BE TESTED.  THERE IS NO    *
                        3470+*    INPUT TO SCKOUT AT ENTRY POINT SCKDEV.                            *
                        3471+*                                                                      *
                        3472+*OUTPUT                                                                *
                        3473+*    THERE IS NO OUTPUT FROM SCKOUT.                                   *
                        3474+*                                                                      *
                        3475+*EXTERNAL REFERENCES                                                   *
                        3476+*    * SCANIT - ENTRY TO DELIMITER SCAN ROUTINE                        *
                        3477+*    * SCAMMA - SCANIT INDICATOR SET TO ALLOW A COMMA                  *
                        3478+*    * $CAERR - ERROR CODE SAVE AREA                                   *
                        3479+*    * $CAERK - EXIT TO LOAD #ERRPG, THE ERROR PROGRAM                 *
                        3480+*    * DLPTYP - DLPRNT INDICATOR FOR OUTPUT DEVICE                     *
                        3481+*    * $IOIND - NUCLEUS INDICATOR WHICH TELLS WHETHER OR NOT THE       *
                        3482+*      PRINTER IS DOWN ($MPDWN) AND WHETHER OR NOT THE CRT IS PRESENT  *
                        3483+*      ON THE SYSTEM ($CRTAV), AND CONTAINS THE COMMAND KEYS ONLY IND  *
                        3484+*    * $KEYCD - NUCLEUS INDICATOR TO GIVE INPUT MODE                   *
                        3485+*    * $CRTIN - NUCLEUS INDICATOR CONCERNING CRT                       *
                        3486+*    * $EXFTR - CORE EXPANSION FACTOR                                  *
                        3487+*    * $$PYCD - ENTRY TO CLEAR CRT AND LIGHT COMMAND INDICATORS        *
                        3488+*    * $$PRES - ENTRY TO ENABLE KEYBOARD TO DEPRESS                    *
                        3489+*                                                                      *
                        3490+*EXIT, NORMAL                                                          *
                        3491+*    NORMAL EXIT FROM SCKOUT (AT BOTH ENTRY POINTS) IS TO THE BYTE     *
                        3492+*    FOLLOWING THE BRANCH TO SCKOUT OR SCKDEV.  UPON EXIT FROM SCKOUT, *
                        3493+*    THE PSR WILL BE SET HIGH TO INDICATE A VALID PARAMETER AND ZERO   *
                        3494+*    TO INDICATE THAT NEITHER 'CRT' NOR 'PRINTER' WAS FOUND.  IF       *
                        3495+*    SCKDEV RETURNS TO THE BYTE FOLLOWING THE BRANCH, THIS INDICATES   *
                        3496+*    THAT NO ERRORS ARE ENCOUNTERED.                                   *
                        3497+*                                                                      *
                        3498+*EXIT, ERROR                                                           *
                        3499+*    ERROR EXIT FROM SCKOUT (ENTRY POINT SCKOUT) IS TO THE BYTE        *
```

SCKOUT - CHECK THE NEXT PARAMETER

```
3500+*   FOLLOWING THE BRANCH TO SCKOUT, WITH THE ERR CODE SET IN $CAERR, *
3501+*   THE PSR SET LOW, AND @XR POINTING TO THE FIRST INVALID CHARACTER. *
3502+*   ERROR EXIT FROM SCKOUT (ENTRY PT SCKDEV) IS TO THE USER-DEFINED   *
3503+*   LABEL, $CKERR, WITH THE ERROR CODE SET IN $CAERR AND @XR POINTS   *
3504+*   OUTSIDE THE INPUT LINE BUFFER (USER VALUE DESTROYED).             *
3505+*                                                                     *
3506+*TABLES/WORKAREAS                                                     *
3507+*   NONE                                                              *
3508+*                                                                     *
3509+*ATTRIBUTES                                                           *
3510+*   RELOCATABLE AND RE-ENTERABLE                                      *
3511+*                                                                     *
3512+*CHARACTER CODE DEPENDENCY                                            *
3513+*   NONE                                                              *
3514+*                                                                     *
3515+*NOTES                                                                *
3516+*   ERROR PROCEDURES                                                  *
3517+*      UPON DETECTING AN ERROR, SCKOUT SETS THE APPROPRIATE ERR CODE  *
3518+*      IN $CAERR AND RETURNS EITHER TO THE BYTE FOLLOWING THE BRANCH  *
3519+*      TO SCKOUT OR TO THE USER-DEFINED LABEL, $CKERR.                *
3520+*                                                                     *
3521+*   REGISTER USAGE                                                    *
3522+*      REGISTER 2 (@XR) IS USED TO SCAN ACROSS THE INPUT LINE BUFFER. *
3523+*      RESISTER 4 (@PSR) IS SET TO INDICATE THE CONDITION FOUND IN    *
3524+*      SCKOUT (ENTRY POINT SCKOUT).                                   *
3525+*                                                                     *
3526+*   SAVED/RESTORED AREAS                                             *
3527+*      NONE                                                           *
3528+*                                                                     *
3529+*   MODIFICATION CONSIDERATIONS                                       *
3530+*      NONE                                                           *
3531+*                                                                     *
3532+*   REQUIRED MODULES                                                  *
3533+*      * @SYSEQ - COMMON SYSTEM EQUATES                               *
3534+*      * @FXDEQ - FIXED CORE LOCATIONS INSIDE NUCLEUS                 *
3535+*      * @ERMEQ - ERROR MESSAGE EQUATES (SELECTED ERROR CODES)        *
3536+*      * @CANEQ - FIXED CORE LOCATIONS OUTSIDE NUCLEUS                *
3537+*      * $CANIT - DELIMITER SCAN ROUTINE                              *
3538+*      * DLPRNT - ROUTINE TO PRINT THE CURRENT LINE                   *
3539+*                                                                     *
3540+*   OTHER                                                             *
3541+*      NONE                                                           *
3542+***********************************************************************
```

```
                      0EDA 3544+SCKOUT EQU    *                          BEGINNING OF SCKOUT SUBROUTINE
     0EDA 34 08 0F6D       3545+        ST     SCK460+@OP1,@ARR           SAVE RETURN ADDRESS
     0EDE 34 02 0F61       3546+        ST     SCK440+@OP1,@XR            SAVE XR POINTER
     0EE2 3C 01 0EB6       3547+        MVI    SCAMMA,SCACOM              SET SCANIT INDR TO ALLOW COMMA
                           3548+*
                           3549+*                 TEST FOR 'CRT' OR 'PRINTER'
                           3550+*
     0EE6 8D 02 02 0F70    3551+        CLC    SCK001-1(SCK001,@XR),SCKCCR IS 'CRT' SPECIFIFD ?
     0EEB F2 81 0F         3552+        JE     SCK100                     YES, PROCESS CRT PARAMETER
                           3553+*
     0EEE 8D 06 06 0F77    3554+        CLC    SCK002-1(SCK002,@XR),SCKCMP IS 'PRINTER' SPECIFIED ?
     0EF3 F2 81 11         3555+        JE     SCK150                     YES, PROCESS 'PRINTER' PARAM
                           3556+*
                           3557+*                 NEITHER CRT NOR PRINTER SPECIFIED
                           3558+*
     0EF6 35 04 0F79       3559+        L      SCK003,@PSR                SET PSR TO BRANCH ZERO
     0EFA F2 87 69         3560+        J      SCK450                     BRANCH TO RETURN
                           3561+*
                           3562+*                 CALL SCANIT AND CHECK DELIMITER AFTER PARAM
                           3563+*
     0EFD 3C 87 0F1C       3564+SCK100 MVI    SCK300+@Q,@UCB             SET SW TO PROCESS 'CRT'
     0F01 E2 02 03         3565+        LA     SCK001(,@XR),@XR           INDR XR PAST 'CRT'
     0F04 F2 87 03         3566+        J      SCK200                     JUMP TO CALL SCANIT
                           3567+*
     0F07 E2 02 07         3568+SCK150 LA     SCK002(,@XR),@XR           INCR XR PAST 'PRINTER'
                           3569+*
     0F0A C0 87 0E99       3570+SCK200 B      SCANIT                     BYPASS BLANKS AND A COMMA
     0F0E C0 82 0469       3571+        BL     $CAERK                     CALL ERR PROG IF DANGLING COMMA
     0F12 F2 84 06         3572+        JH     SCK300                     IF CHARS SCANNED, SET DLPRNT SW
                           3573+*
     0F15 BD 1E 00         3574+        CLI    @ZERO(,@XR),@EOS           ELSE, IS PARAM FOLLOWED BY EOS ?
     0F18 F2 01 31         3575+        JNE    SCK410                     NO, SET 'INV PARAM' ERROR
                           3576+*
     0F1B F2 80 15         3577+SCK300 JC     SCK350,@NOP                NOP IF PRINTER -- UCB IF CRT
```

```
                              3579+*
                              3580+*                    PRINTER SPECIFIED
                              3581+*
     0F1E 3D 1B 1776          3582+      CLI   DLPTYP,DLPCRT          WAS CRT SPECIFIED BEFORE ?
     0F22 F2 81 2E            3583+      JE    SCK420                YES, SET 'CONFLICTING PARAM' ERR
                              3584+*
     0F25 3D 85 1776          3585+      CLI   DLPTYP,DLPMPR          WAS PRINTER SPECIFIED BEFORE ?
     0F29 F2 81 2E            3586+      JE    SCK430                YES, SET 'DUPLICATING PARAM' ERR
                              3587+*
     0F2C 3C 85 1776          3588+      MVI   DLPTYP,DLPMPR          SET SW FOR MATRIX PRINTER
     0F30 F2 87 12            3589+      J     SCK400                RETURN TO CALLING PGM
                              3590+*
                              3591+*                    CRT SPECIFIED
                              3592+*
     0F33 3D 1B 1776          3593+SCK350 CLI  DLPTYP,DLPCRT          WAS CRT SPECIFIED BEFORE ?
     0F37 F2 81 20            3594+      JE    SCK430                YES SET 'DUPLICATE PARAM' ERR
                              3595+*
     0F3A 3D 85 1776          3596+      CLI   DLPTYP,DLPMPR          WAS PRINTER SPECIFIED BEFORE ?
     0F3E F2 81 12            3597+      JE    SCK420                YES, SET 'CONFLICTING PARAM' ERR
                              3598+*
     0F41 3C 1B 1776          3599+      MVI   DLPTYP,DLPCRT          SET SW FOR CRT
     0F45 35 04 0F7B          3600+SCK400 L    SCK004,@PSR           SET SW FOR BRANCH HIGH
     0F49 F2 87 1A            3601+      J     SCK450                RETURN TO CALLING PROGRAM
                              3602+*
                              3603+*                    SET ERROR CODES
                              3604+*
     0F4C 3C 11 03CD          3605+SCK410 MVI  $CAERR,@@E131          SET 'INV PARAM' ERROR CODE
     0F50 F2 87 0B            3606+      J     SCK440                RETURN
                              3607+*
     0F53 3C 15 03CD          3608+SCK420 MVI  $CAERR,@@E136          SET 'CONFLICTING PARAM' ERR CODE
     0F57 F2 87 04            3609+      J     SCK440                RETURN
                              3610+*
     0F5A 3C 13 03CD          3611+SCK430 MVI  $CAERR,@@E134          SET 'DUPLICATE PARAM' ERR CODE
                              3612+*
     0F5E C2 02 0000          3613+SCK440 LA   *-*,@XR                RESTORE XR VALUE
     0F62 35 04 0F7D          3614+      L     SCK005,@PSR            SET PSR TO BL TO IND ERROR
                              3615+*
                              3616+*                    EXIT
                              3617+*
     0F66 3C 80 0F1C          3618+SCK450 MVI  SCK300+@Q,@NOP         SET CRT OR POINTER INDR OFF
     0F6A C0 87 0000          3619+SCK460 B    *-*                   RETURN TO CALLING PROGRAM
                              3620+*
                              3621+*                    EQUATES USED IN SCKOUT
                              3622+*
                         0003 3623+SCK001 EQU  3                     LENGTH OF 'CRT' PARAMETER
                         0007 3624+SCK002 EQU  7                     LENGTH OF 'PRINTER' PARAMETER
                              3625+*
                              3626+*                    CONSTANTS USED IN SCOUT
                              3627+*
     0F6E C3D9E3         0F70 3628+SCKCCR DC   CL(SCK001)'CRT'        CRT PARAMETER IMAGE
     0F71 D7D9C9D5E3C5D9 0F77 3629+SCKCMP DC   CL(SCK002)'PRINTER'    PRINTER PARAMETER IMAGE
     0F78 0081          0F79 3630+SCK003 DC    XL2'81'               PRINTER CODE FOR BRANCN ON ZERO
     0F7A 0084          0F7B 3631+SCK004 DC    XL2'84'               PSR CODE FOR BRANCH HIGH
     0F7C 0082          0F7D 3632+SCK005 DC    XL2'82'               PSR CODE FOR BRANCH LOW
                              3633+*
```

```
                          0F7E 3635+SCKDEV EQU    *                            PORTION OF SCKOUT TO READY CRT
     0F7E 34 08 0FDE            3636+        ST     SCK650+@OP1,@ARR            SAVE RETURN ADDRESS
     0F82 3C 01 03D3            3637+        MVI    $CRTIN,$CRTUP               SET CRT IN ROLL-UP MODE
                                3638+*
     0F86 3D 1B 1776            3639+        CLI    DLPTYP,DLPCRT               WAS CRT THE SPECIFIED PARM ?
     0F8A F2 81 15              3640+        JE     SCK475                      YES, CHECK FOR ITS EXISTENCE
                                3641+*
     0F8D 3D 85 1776            3642+        CLI    DLPTYP,DLPMPR               ELSE, WAS PRINTER SPECIFIED ?
     0F91 F2 01 47              3643+        JNE    SCK650                      NO, RETURN TO USER
                                3644+*
     0F94 38 01 03D2            3645+        TBN    $IOIND,$MPDWN               ELSE, IS PRINTER DOWN ?
     0F98 F2 90 40              3646+        JF     SCK650                      NO, RETURN TO USER
                                3647+*
     0F9B 3C 96 03CD            3648+        MVI    $CAERR,@@E549               SET ERR CODE FOR PRINTER DOWN
     0F9F F2 87 19              3649+        J      SCK550                      DESTROY YR AND EXIT
                                3650+*
     0FA2 38 02 03D2            3651+SCK475  TBN    $IOIND,$CRTAV               IS CRT ON THE SYSTEM ?
     0FA6 F2 90 0E              3652+        JF     SCK500                      NO, SET ERROR CODE
                                3653+*
     0FA9 38 01 03C3            3654+        TBN    $KEYCD,$CARDI               IS CRT SPECIFIED FROM CARDS ?
     0FAD F2 90 13              3655+        JF     SCK600                      IF NOT, SKIP ERROR ROUTINE
                                3656+*
     0FB0 3C 3A 03CD            3657+        MVI    $CAERR,@@E248               SET ERROR CODE - 'CRT SPECIFIED
                                3658+*                                         * WHEN I/O IS FROM CARD READER'
     0FB4 F2 87 04              3659+        J      SCK550                      SET PSR AND EAT
                                3660+*
     0FB7 3C 38 03CD            3661+SCK500  MVI    $CAERR,@@E241               SET ERR CODE-CRT NOT ON SYSTEM
                                3662+*
     0FBB C2 02 0F7E            3663+SCK550  LA     SCKDEV,@XR                  INCR XR TO AVOID SYNTAX ERROR
     0FBF C0 87 0469            3664+        B      SCKERR                      RETURN TO CALLING PROGRAM
                                3665+*
                                3666+*              READY CRT
                                3667+*
     0FC3 3A 08 03D2            3668+SCK600  SBN    $IOIND,$CMDKY               SET CMND KEYS ONLY INDR ON
                                3669+*                                         SCKCL LITE
     0FC7 0E 00 0FCF 043B       3670+SCKCL0  ALC    SCKCL1+@D1(1),$EXFTR        CALCULATE ENTRY ADDRESS
     0FCD C0 87 2200            3671+SCKCL1  B      $$PYCD                      CLEAR CRT / LIGHT CMND INDRS
     0FD1 0F 00 0FCF 043B       3672+        SLC    SCKCL1+@D1(1),$EXFTR        INITIALIZE ENTRY ADDRESS

     0FD7 C0 87 0890            3674+        B      $$PRES                      ENABLE KEYBOARD ENTRY TO DEPRES
                                3675+*
     0FDB C0 87 0000            3676+SCK650  B      *-*                         RETURN TO CALLING PROGRAM
                          0FDF 3677+SCKEND EQU    *                            END OF ROUTINE
                                3678+*** END OF EXPANSION ***
                                3679+***                          END OF SCKOUT                              ***
                                3680 *
```

```
                          3682 *********************************************************************
                          3683 * 5703-XM1       COPYRIGHT IBM CORP. 1970                            *
                          3684 *                REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083  *
                          3685 *                                                                    *
                          3686 *********************************************************************
                          3687 *STATUS -                                                            *
                          3688 *    VERSION 1 MODIFICATION 0                                         *
                          3689 *                                                                    *
                          3690 *FUNCTION -                                                           *
                          3691 *    * SVARAB SCANS THE BASIC LINE, BEGINNING AT A POINT PASSED BY    *
                          3692 *      THE CALLING ROUTINE                                            *
                          3693 *    * A POINTER IS SET TO THE FIRST VARIABLE OR ARRAY SYMBOL         *
                          3694 *      ENCOUNTERED                                                    *
                          3695 *    * SVAVTC IS SET TO A CODE THE WILL INDICATE THE VARIABLE TYPE    *
                          3696 *    * SVALNG IS SET TO THE VARIABLE LENGTH                           *
                          3697 *                                                                    *
                          3698 *ENTRY POINTS -                                                       *
                          3699 *    * SVARAB HAS ONLY 1 ENTRY POINT                                  *
                          3700 *    * THE CALLING SEQUENCE IS                                        *
                          3701 *        B      SVARAB                                                *
                          3702 *                                                                    *
                          3703 *INPUT -                                                              *
                          3704 *    * REGISTER @XR - CONTAINS THE CORE ADDRESS OF THE INITIAL        *
                          3705 *      CHARACTER TO BE EXAMINED                                       *
                          3706 *    * GRTYPE - CONTAINS THE STATEMENT TYPE CODE FOR THE BASIC        *
                          3707 *      STATEMENT LINE BEING PROCESSED                                 *
                          3708 *                                                                    *
                          3709 *OUTPUT -                                                             *
                          3710 *    * REGISTER @XR - CONTAINS THE CORE ADDRESS OF THE FIRST          *
                          3711 *      CHARACTER OF THE FIRST VARIABLE ENCOUNTERED                    *
                          3712 *    * IF NO VARIABLE EXISTS, @XR CONTAINS CORE ADDRESS OF THE        *
                          3713 *      CARRIAGE RETURN BYTE                                           *
                          3714 *    * SVAVTC - 1 BYTE, CONTAINS THE VARIABLE TYPE CODE OF THE        *
                          3715 *      VARIABLE                                                       *
                          3716 *    * SVALNG - 1 BYTE, CONTAINS THE LENGTH OF THE VARIABLE           *
                          3717 *                                                                    *
                          3718 *EXTERNAL REFERENCES -                                                *
                          3719 *    GRTYPE - BASIC STATEMENT TYPE CODE                                *
                          3720 *                                                                    *
                          3721 *EXITS, NORMAL -                                                      *
                          3722 *    NORMAL EXIT IS TO THE FIRST INSTRUCTION FOLLOWING THE CALLING    *
                          3723 *    SEQUENCE, THE BASE REGISTER IS RESTORED.  THE RETURN ADDRESS IS  *
                          3724 *    IN THE ADDRESS RECALL REGISTER (@ARR)                            *
                          3725 *                                                                    *
                          3726 *EXITS, ERROR -                                                       *
                          3727 *    N/A                                                             *
                          3728 *                                                                    *
                          3729 *TABLES/WORK AREAS -                                                  *
                          3730 *    * THE STATEMENT BRANCH TABLE - ONE ENTRIES FOR EACH BASIC        *
                          3731 *      STATEMENT TYPE.  EACH ENTRY IS 3 BYTES AND CONTAINS            *
                          3732 *      * LENGTH OF STATEMENT KEYWORD - 1 BYTE                          *
                          3733 *      * CORE ADDRESS OF STATEMENT PROCESSING ROUTINE - 2 BYTES       *
                          3734 *    * THE CONSTANTS AND WORK AREAS RESIDE AT THE END OF THE          *
                          3735 *      EVECUTABLE CODE AND ARE REFERENCED BY OR                       *
                          3736 *                                                                    *
                          3737 *ATTRIBUTES -                                                         *
```

```
                              3738 *   N/A                                                      *
                              3739 *                                                            *
                              3740 *CHARACTER CODE DEPENDENCY                                   *
                              3741 *    THE OPERATION OF THIS MODULE DEPENDS UPON THE FOLLOWING *
                              3742 *    PROPERTIES OF THE INTERNAL REPRESENTATION OF THE EXTERNAL *
                              3743 *    CHARACTER SET                                           *
                              3744 *       * MOST CODING HAS BEEN ARRANGED SO THAT REDEFINITION OF *
                              3745 *         CHARACTER CONSTANTS, BY REASSEMBLY, WILL RESULT IN A CORRECT *
                              3746 *         MODULE FOR THE NEW DEFINITION                      *
                              3747 *       * ALPHABETIC LETTERS A THROUGH Z ARE PRESUMED TO BE CODED IN *
                              3748 *         INCREASING COLLATING SEQUENCE, AND THE RANGE OF CHARACTER *
                              3749 *         CONSTANTS FOR THIS SERIES IS EXPECTED TO EXCLUDE ALL NUMERIC *
                              3750 *         CHARACTER CONSTANTS                                *
                              3751 *       * NUMERIC CHARACTERS 0 - 9 ARE PRESUMED TO BE CODED IN *
                              3752 *         INCREASING COLLATING SEQUENCE                       *
                              3753 *       * EXTENDED ALPHABETIC LETTERS ($, #, @) ARE PRESUMEMED TO BE *
                              3754 *         IN INCREASING COLLATING SEQUENCE, AND ARE ALL EXPECTED TO *
                              3755 *         COLLATE LOWER THAN LETTER (A).                       *
                              3756 *       * DECIMAL NUMBERS MUST BE CODED SO THAT THE LOW ORDER FOUR *
                              3757 *         BITS, WHEN CONSIDERED AS A BINARY INTEGER, IDENTIFY THE *
                              3758 *         VALUE OF THE DIGIT                                 *
                              3759 *    THE SPECIFIC INSTRUCTIONS (INSTRUCTION SEQUENCES) WHICH REQUIRE *
                              3760 *    MODIFICATION IF THESE PROPERTIES OF THE CHARACTER SET ARE CHANGED *
                              3761 *    MAY BE IDENTIFIED BY -                                  *
                              3762 *       * INSTRUCTION SEQUENCES AT LABELS SVA075 AND SVA080  *
                              3763 *       * INSTRUCTION SEQUENCES AT LABELS SVA460 AND SVA465  *
                              3764 *       * INSTRUCTION SEQUENCES AT LABEL SVA930             *
                              3765 *                                                            *
                              3766 *NOTES -                                                     *
                              3767 *    ERROR PROCEDURES                                        *
                              3768 *       N/A                                                  *
                              3769 *                                                            *
                              3770 *    REGISTER USAGE                                          *
                              3771 *       * REGISTER @XR IS BOTH AN INPUT AND OUTPUT PARAMETER *
                              3772 *       * REGISTER @BR IS SAVED ON ENTRY, USED DURING EXECUTION, *
                              3773 *         AND RESTORED ON EXIT                              *
                              3774 *                                                            *
                              3775 *    SAVED/RESTORED                                          *
                              3776 *       N/A                                                  *
                              3777 *                                                            *
                              3778 *    MODIFICATION CONSIDERATIONS                             *
                              3779 *       N/A                                                  *
                              3780 *                                                            *
                              3781 *    REQUIRED MODULES                                        *
                              3782 *       @SYSEQ - COMMON SYSTEM EQUATES                       *
                              3783 *       $B@EQU - COMPILER SYSTEM EQUATES                     *
                              3784 *                                                            *
                              3785 *    OTHER                                                   *
                              3786 *       N/A                                                  *
                              3787 ********************************************************************
```

```
                              3789 ******************************************************************************
                              3790 *                                                                            *
                              3791 ******************************************************************************
                              3792 *                                                                            *
                              3793 *                    SVARAB BASIC SYNTAX SCAN ROUTINE                        *
                              3794 *                                                                            *
                              3795 ******************************************************************************
                              3796 *                                                                            *
                              3797 ******************************************************************************

                              3799 *
                              3800 * SVARAB ENTRY, SET RETURN LINKAGE AND ADDRESSABILITY
                              3801 *
                         0FDF 3802 SVARAB EQU    *                          SVARAB ENTRY POINT
    0FDF 34 08 11B3           3803        ST     SVA320+@OP1,@ARR           SAVE RETURN ADDR
    0FE3 34 01 11AF           3804        ST     SVA315+@OP1,@BR            SAVE PT
                         13DE 3805        USING  SVA960,@BR                 SET BASE ADDR
    0FE7 C2 01 13DE           3806        LA     SVA960,@BR                 LOAD BASE
    0FEB 7C 78 24             3807        MVI    SVAMAG(,@BR),B@TDUM        SET MAGIC TYPE              1-4
    0FEE 7C 00 25             3808        MVI    SVAZRO(,@BR),@ZERO         CLEAR INDEX                1-4
                              3809 *
                              3810 * DETERMINE THE BRANCH TABLE INDEX
                              3811 *
    0FF1 0C 00 13FA 1B96      3812        MVC    SVASTC(1),GRTYPE           SAVE STATEMENT TYPE CODE
    0FF7 3B 80 13FA           3813 SVA020 SBF    SVASTC,SVADIS              SET DISABLE SW OFF
                              3814 *   CODE FOR STRING FUNCTION...                                     1-4
    0FFB 5F 00 24 1C          3815        SLC    SVAMAG(,@BR),SVASTC(,@BR)  CHECK TYPE CODE            1-4
    0FFF F2 02 0C             3816        JNL    SVA030                     NOT SPECIAL---SKIP         1-4
    1002 5F 00 25 24          3817        SLC    SVAZRO(,@BR),SVAMAG(,@BR)   COMPLEMENT TYPE           1-4
    1006 5E 00 25 25          3818        ALC    SVAZRO(,@BR),SVAZRO(,@BR)   DOUBLE IT                1-4
    100A 5E 00 1C 25          3819        ALC    SVASTC(1,@BR),SVAZRO(,@BR)  ACCUMULATE               1-4
    100E C2 01 1468           3820 SVA030 LA     SVABRT-3,@BR               DISTRIBUTOR TABLE          1-4
    1012 36 01 13FA           3821        A      SVASTC,@BR                 INDEX TABLE ENTRY
    1016 1C 00 13F8 00        3822        MVC    SVAKWL(1),SVAOTD(,@BR)     SAVE KEYWORD LENGTH
    101B 75 01 02             3823        L      SVA2TD(,@BR),@BR           LOAD BRANCH ADDR
    101E D0 87 00             3824        B      SVADP0(,@BR)               BRANCH TO PROC ROUTINE
```

```
                         3826 ************************************************************************
                         3827 ************************************************************************
                         3828 *                                                                      *
                         3829 * ROUTINE TO PROCESS STATEMENT LINES THAT CONTAIN NO VARIABLES         *
                         3830 *                                                                      *
                         3831 ************************************************************************
                         3832 ************************************************************************

    1021 BD 1E 00        3834 SVA050 CLI   SVADP0(,@XR),B@EOST        AT END OF STATEMENT ?
    1024 C0 81 11AC       3835        BE    SVA315                    YES, RETURN TO CALLING RTN
    1028 36 02 1405       3836        A     SVAIO1,@XR                INCR PT TO NEXT BYTE
    102C C0 87 1021       3837        B     SVA050                    CYCLE LOOP UNTIL EOS
```

```
                         3839 *********************************************************************
                         3840 *                                                                   *
                         3841 * ROUTINE TO PROCESS ARITHMETIC AND CHARACTER ASSIGMENT STATEMENTS   *
                         3842 *                                                                   *
                         3843 *********************************************************************
                         3844 *********************************************************************
                         3845 *
    1030 C2 01 13DE      3846 SVA070 LA    SVA960,@BR            SET BASE ADDR
    1034 7D 00 18        3847        CLI   SVABSW(,@BR),SVAOFF   IS BRANCH SW OFF ?
    1037 F2 01 25        3848        JNE   SVA085               NO, SCAN FOR VARIABLE
    103A BD C1 00        3849 SVA075 CLI   SVADP0(,@XR),@CHARA   IF BYTE IS IN STANDARD
    103D F2 82 06        3850        JL    SVA080               * ALPHABET, EXIT LOOP
    1040 BD E9 00        3851        CLI   SVADP0(,@XR),@CHARZ   * AND GO SCAN FOR VARIABLES
    1043 F2 04 19        3852        JNH   SVA085               * IN LINE
    1046 BD 7B 00        3853 SVA080 CLI   SVADP0(,@XR),@NUMBR   TEST FOR SPECIAL ALPHABETIC
    1049 F2 81 13        3854        JE    SVA085               * CHARACTERS, IF EQUAL TO
    104C BD 7C 00        3855        CLI   SVADP0(,@XR),@ASIGN   * ANY, GO SCAN FOR ANY
    104F F2 81 0D        3856        JE    SVA085               * VARIABLES IN THE LINE
    1052 BD 5B 00        3857        CLI   SVADP0(,@XR),@DOLAR   * $ INCLUDED FOR WTC
    1055 F2 81 07        3858        JE    SVA085               * CONSIDERATIONS
    1058 E2 02 01        3859        LA    1(,@XR),@XR          INCR PT
    105B C0 87 103A      3860        B     SVA075               LOOP UNTIL ALPHA CHAR IS FOUND
    105F C0 87 11BB      3861 SVA085 B     SVA395               SCAN FOR VARIABLE TYPE
    1063 7C 01 18        3862        MVI   SVABSW(,@BR),SVAONN  SET BR SW ON
    1066 C0 87 11AC      3863        B     SVA315               RETURN
```

```
                              3865 ************************************************************************
                              3866 *                                                                      *
                              3867 * ROUTINE TO PROCESS THE COMPUTED GOTO STATEMENT                       *
                              3868 *                                                                      *
                              3869 ************************************************************************
                              3870 ************************************************************************
                              3871 *
     106A C2 01 13DE          3872 SVA090 LA    SVA960,@BR               SET BASE ADDR
                              3873 *
                              3874 * TEST BRANCH SWITCH
                              3875 *
     106E 7D 00 18            3876 SVA095 CLI   SVABSW(,@BR),SVAOFF       IS BR SW OFF ?
     1071 F2 01 12            3877        JNE   SVA110                   NO, SKIP INCR PAST KEYWORD
                              3878 *
                              3879 * INCREMENT LINE POINTER PAST KEYWORD 'ON'
                              3880 *
     1074 7C 01 18            3881 SVA100 MVI   SVABSW(,@BR),SVAONN       BR SW ON
     1077 C0 87 132B          3882        B     SVA700                   INCR PAST KEYWORD
     107B C0 87 1342          3883        B     SVA900                   INCR TO 1ST ALPHA BYTE
     107F 7C 02 1A            3884        MVI   SVAKWL(,@BR),B@LKON       SET KEYWORD LNG
     1082 C0 87 132B          3885        B     SVA700                   INCR PAST KEYWORD
                              3886 *
                              3887 * DETERMINE VARIABLE TYPE AND RETURN
                              3888 *
     1086 C0 87 11BB          3889 SVA110 B     SVA395                   DETERMINE VAR TYPE
     108A C0 87 11AC          3890        B     SVA315                   RETURN
```

```
                           3892 *********************************************************************
                           3893 *                                                                   *
                           3894 *   ROUTINE TO PROCESS THE MAT ASSIGNMENT STATEMENTS                *
                           3895 *                                                                   *
                           3896 *********************************************************************
                           3897 *********************************************************************
                           3898 *
     108E C2 01 13DE       3899 SVA120 LA    SVA960,@BR              SET BASE ADDR
                           3900 *
                           3901 *  TEST BRANCH SWITCH
                           3902 *
     1092 7D 00 18         3903 SVA125 CLI   SVABSW(,@BR),SVAOFF     IS BRANCH SW OFF ?
     1095 F2 01 23         3904        JNE   SVA150                  YES, CHECK FOR VAR
                           3905 *
                           3906 *   INCREMENT LINE POINTER PAST KEYWORD
                           3907 *
     1098 C0 87 132B       3908 SVA130 B     SVA700                  INCR PAST KEYWORD
     109C 7C 01 18         3909        MVI   SVABSW(,@BR),SVAONN     SET BR SW ON
     109F 7C 00 1E         3910        MVI   SVAPCT(,@BR),SVAOFF     SET PAREN SW
                           3911 *
                           3912 *   INCREMENT TO THE ARRAY VARIABLE AND SET TYPE CODE
                           3913 *
     10A2 C0 87 1342       3914 SVA140 B     SVA900                  TO 1ST ALPHA BYTE
     10A6 34 02 10B6       3915        ST    SVA148+@OP1,@XR         SAVE VAR ADDR
     10AA 7C 01 23         3916        MVI   SVALNG(,@BR),SVAVL1     SET VAR LNG TO 1
     10AD D0 87 00         3917 SVA144 B     SVA960(,@BR)            DETM VAR LNG
     10B0 7C 08 22         3918        MVI   SVAVTC(,@BR),SVANAC     SET TYPE CODE TO ARITH ARRAY
     10B3 C2 02 0000       3919 SVA148 LA    *-*,@XR                 RESTORE VAR ADDR
     10B7 C0 87 11AC       3920        B     SVA315                  RETURN
                           3921 *
                           3922 *   STACK NEXT THREE NON-BLANK BYTES
                           3923 *
     10BB BD 7E 00         3924 SVA150 CLI   SVADP0(,@XR),B@EQUL     AT EQ SIGN ?
     10BE F2 81 06         3925        JE    SVA151                  YES, INCR PAST IT
     10C1 BD 40 00         3926        CLI   SVADP0(,@XR),B@BLNK     AT BLANK ?
     10C4 F2 01 03         3927        JNE   SVA152                  NO, SAVE PRESENT PT
     10C7 D0 87 00         3928 SVA151 B     SVA960(,@BR)            YES, SKIP TO NON-BLANK BYTE
     10CA 34 02 1127       3929 SVA152 ST    SVA165+@OP1,@XR         SAVE VAR ADDR
     10CE 6C 00 1F 00      3930        MVC   SVALS1(,@BR),SVADP0(1,@XR)  STACK CHAR
     10D2 D0 87 00         3931        B     SVA960(,@BR)            TO NEXT NON BLANK BYTE
     10D5 6C 00 20 00      3932        MVC   SVALS2(,@BR),SVADP0(1,@XR)  STACK CHAR
     10D9 D0 87 00         3933        B     SVA960(,@BR)            TO NEXT NON BLANK BYTE
     10DC 6C 00 21 00      3934        MVC   SVALS3(,@BR),SVADP0(1,@XR)  STACK CHAR
                           3935 *
                           3936 *   TEST FOR FUNCTIONS INV AND TRN
                           3937 *
     10E0 5D 02 21 80      3938 SVA154 CLC   SVALSA(B@LIFN,@BR),SVAINV(,@BR)  FUNC 'INV'
     10E4 F2 81 07         3939        JE    SVA156                  YES, INCR TO VAR
     10E7 5D 02 21 83      3940 SVA155 CLC   SVALSA(B@LIFN,@BR),SVATRN(,@BR)  FUNC 'TRN'
     10EB F2 01 1B         3941        JNE   SVA160                  NO, SCAN FOR VAR
                           3942 *
                           3943 *   INCREMENT LINE PT TO ARGUMENT VAR AND SET TYPE CODE
                           3944 *
     10EE 76 02 27         3945 SVA156 A     SVAIO1(,@BR),@XR        INCR PAST CHAR
     10F1 C0 87 1342       3946        B     SVA900                  TO 1ST ALPHA BYTE
     10F5 34 02 1105       3947        ST    SVA158+@OP1,@XR         SAVE VAR ADDR
```

```
    10F9 7C 01 23            3948         MVI    SVALNG(,@BR),SVAVL1      SET VAR LNG TO 1
    10FC D0 87 00            3949         B      SVA960(,@BR)            TO NEXT NON BLANK BYTE
    10FF 7C 08 22            3950         MVI    SVAVTC(,@BR),SVANAC     SET TYPE CODE TO ARITH ARRAY
    1102 C2 02 0000          3951 SVA158 LA     *-*,@XR                 RESTORE VAR ADDR
    1106 F2 87 A3            3952         J      SVA315                  RETURN
                            3953 *
                            3954 * TEST FOR FUNCTIONS CON, IDN, AND ZER
                            3955 *
    1109 5D 02 21 86         3956 SVA160 CLC    SVALSA(B@LIFN,@BR),SVACON(,@BR)  FUNC 'CON' ?
    110D F2 81 0E            3957         JE     SVA163                  YES, PROC FUNC
    1110 5D 02 21 89         3958         CLC    SVALSA(B@LIFN,@BR),SVAIDN(,@BR)  FUNC 'IDN' ?
    1114 F2 81 07            3959         JE     SVA163                  YES, PROC FUNC
    1117 5D 02 21 8C         3960         CLC    SVALSA(B@LIFN,@BR),SVAZER(,@BR)  FUNC 'ZER' ?
    111B F2 01 06            3961         JNE    SVA165                  NO, SCAN FOR VAR
    111E E2 02 01            3962 SVA163 LA     1(,@XR),@XR             INCR PT
    1121 F2 87 04            3963         J      SVA168                  SCAN FOR VARS
                            3964 *
                            3965 * BRANCH TO SCAN RTN TO FIND A VARIABLE AND RETURN
                            3966 *
    1124 C2 02 0000          3967 SVA165 LA     *-*,@XR                 RESTORE VAR ADDR
    1128 C0 87 118A          3968 SVA168 B      SVA270                  DETM VAR TYPE
```

```
                              3970 ***************************************************************************
                              3971 *                                                                         *
                              3972 * ROUTINE TO PROCESS GET AND PUT STATEMENTS                               *
                              3973 *                                                                         *
                              3974 ***************************************************************************
                              3975 ***************************************************************************
                              3976 *
     112C C2 01 13DE          3977 SVA170 LA     SVA960,@BR              SET BASE ADDR
                              3978 *
                              3979 * TEST BRANCH SWITCH
                              3980 *
     1130 7D 00 18            3981 SVA175 CLI    SVABSW(,@BR),SVAOFF      IS BR SW OFF ?
     1133 F2 01 07            3982        JNE    SVA190                  NO, CHECK FOR VAR
     1136 7C 01 18            3983        MVI    SVABSW(,@BR),SVAONN      SET BR SW ON
     1139 C0 87 132B          3984        B      SVA700                  SKIP KEYWORD
                              3985 *
                              3986 * SCAN FOR VARIABLE AND RETURN
                              3987 *
     113D C0 87 11BB          3988 SVA190 B      SVA395                  SCAN FOR VAR
     1141 F2 87 68            3989        J      SVA315                  RETURN
```

```
                              3991 ********************************************************************
                              3992 *                                                                  *
                              3993 * ROUTINE TO PROCESS THE REMAINING MATRIX STATEMENTS               *
                              3994 *                                                                  *
                              3995 ********************************************************************
                              3996 ********************************************************************
                              3997 *
      1144 C2 01 13DE         3998 SVA192 LA    SVA960,@BR              SET BASE ADDR
                              3999 *
                              4000 * TEST BRANCH SWITCH
                              4001 *
      1148 7D 00 18           4002 SVA194 CLI   SVABSW(,@BR),SVAOFF      IS BR SW OFF ?
      114B F2 01 3C           4003        JNE   SVA270                   NO, SCAN FOR VARS
                              4004 *
                              4005 * INCREMENT PAST THE KEYWORDS
                              4006 *
      114E 7C 01 18           4007 SVA196 MVI   SVABSW(,@BR),SVAONN      SET BR SW ON
      1151 7C 00 1E           4008        MVI   SVAPCT(,@BR),SVAOFF      SET PAREN COUNT TO 0
      1154 C0 87 132B         4009        B     SVA700                   INCR PAST KEYWORDS
      1158 F2 87 2F           4010        J     SVA270                   SCAN FOR VARS
```

```
                             4012 *********************************************************************
                             4013 *                                                                   *
                             4014 * ROUTINE TO PROCESS REMAINING NON-MATRIX STATEMENTS                *
                             4015 *                                                                   *
                             4016 *********************************************************************
                             4017 *********************************************************************
                             4018 *
    115B C2 01 13DE          4019 SVA200 LA     SVA960,@BR                SET BASE ADDR
                             4020 *
                             4021 * TEST BRANCH SWITCH
                             4022 *
    115F 7D 00 18            4023 SVA206 CLI    SVABSW(,@BR),SVAOFF       IS BR SW OFF ?
    1162 F2 01 07            4024        JNE    SVA220                    NO, SCAN FOR VAR
                             4025 *
                             4026 * INCREMENT PAST KEYWORD
                             4027 *
    1165 C0 87 132B          4028 SVA210 B      SVA700                    INCR PAST KEYWORD
    1169 7C 01 18            4029        MVI    SVABSW(,@BR),SVAONN       SET BR SW ON
                             4030 *
                             4031 * SCAN FOR VARIABLE AND RETURN
                             4032 *
    116C C0 87 11BB          4033 SVA220 B      SVA395                    SCAN FOR VAR
    1170 F2 87 39            4034        J      SVA315                    RETURN
```

```
                           4036 ****************************************************************************
                           4037 *                                                                          *
                           4038 * ROUTINE TO PROCESS MATRIX GET AND PUT STATEMENTS                         *
                           4039 *                                                                          *
                           4040 ****************************************************************************
                           4041 ****************************************************************************
                           4042 *
    1173 C2 01 13DE        4043 SVA250 LA     SVA960,@BR                SET BASE ADDR
                           4044 *
                           4045 * TEST BRANCH SWITCH
                           4046 *
    1177 7D 00 18          4047 SVA255 CLI    SVABSW(,@BR),SVAOFF       IS BR SW OFF ?
    117A F2 01 0D          4048        JNE    SVA270                    NO, SCAN FOR VAR
                           4049 *
                           4050 * INCREMENT PAST THE KEYWORDS AND FILE NAME
                           4051 *
    117D 7C 01 18          4052 SVA260 MVI    SVABSW(,@BR),SVAONN       SET BR SW ON
    1180 7C 00 1E          4053        MVI    SVAPCT(,@BR),SVAOFF       SET PAREN CT TO 0
    1183 C0 87 132B        4054        B      SVA700                    SKIP KEYWORD
    1187 F2 87 0A          4055        J      SVA305                    PROCESS AS VARIABLE       1-4
                           4056 *
                           4057 * DETERMINE IF VARIABLE IS IN A ARITHMETIC EXPRESION
                           4058 *
    118A C0 87 1342        4059 SVA270 B      SVA900                    TO FIRST ALPHA BYTE
    118E 7D 00 1E          4060 SVA300 CLI    SVAPCT(,@BR),@ZERO        IS PAREN COUNT ZERO ?
    1191 F2 81 07          4061        JE     SVA310                    YES, SET TYPE CODE
    1194 C0 87 11BE        4062 SVA305 B      SVA400                    NO, SCAN FOR VAR TYPE
    1198 F2 87 11          4063        J      SVA315                    RETURN
                           4064 *
                           4065 * SET VARIABLE TYPE CODE
                           4066 *
    119B 34 02 11A8        4067 SVA310 ST     SVA312+@OP1,@XR           SAVE VAR ADDR
    119F 7C 01 23          4068        MVI    SVALNG(,@BR),SVAVL1       SET LNG TO 1
    11A2 D0 87 00          4069        B      SVA960(,@BR)              COUNT BLANKS
    11A5 C2 02 0000        4070 SVA312 LA     *-*,@XR                   RESTORE VAR ADDR
    11A9 7C 08 22          4071        MVI    SVAVTC(,@BR),SVANAC       TYPE CODE TO ARITH ARRAY
    11AC C2 01 0000        4072 SVA315 LA     *-*,@BR                   RESTORE PT
    11B0 C0 87 0000        4073 SVA320 B      *-*                       RETURN TO CALLING RTN
```

```
                          4075 ****************************************************************************
                          4076 *                                                                          *
                          4077 * END OF STATEMENT ROUTINE                                                 *
                          4078 *                                                                          *
                          4079 ****************************************************************************
                          4080 ****************************************************************************
                          4081 *
                          4082 * TURN BRANCH SWITCH OFF AND RETURN TO CALLING ROUTINE
                          4083 *
    11B4 7C 00 18         4084 SVA330 MVI    SVABSW(,@BR),SVAOFF      SET BR SW OFF
    11B7 C0 87 11AC       4085        B      SVA315                  RETURN
```

```
                            4087 ********************************************************************
                            4088 *                                                                  *
                            4089 * VARIABLE SCAN ROUTINE                                            *
                            4090 *                                                                  *
                            4091 ********************************************************************
                            4092 ********************************************************************
                            4093 *
     11BB 7C 0F 1E          4094 SVA395 MVI    SVAPCT(,@BR),SVAL15       SET PAREN CT TO A MAX
                            4095 *
                            4096 * ENTER AND SAVE RETURN ADDRESS
                            4097 *
     11BE 34 08 132A        4098 SVA400 ST     SVA670+@OP1,@ARR          SAVE RETURN ADDR
                            4099 *
                            4100 * INCREMENT TO THE NEXT ALPHA BYTE ENCOUNTERED IN THE LINE AND STACK IT
                            4101 *
     11C2 C0 87 1342        4102 SVA410 B      SVA900                    TO ALPHA BYTE
     11C6 6C 00 1F 00       4103        MVC    SVALS1(1,@BR),SVADP0(,@XR)  STACK BYTE
     11CA 34 02 1326        4104        ST     SVA660+@OP1,@XR           SAVE VAR ADDR
                            4105 *
                            4106 * SET VARIABLE LENGTH INITIALLY TO ONE
                            4107 *
     11CE 7C 01 23          4108 SVA415 MVI    SVALNG(,@BR),SVAVL1       VAR LNG EQ 1
                            4109 *
                            4110 * INCREMENT THE LINE POINTER TO 1ST NON-BLANK BYTE AND STACK BYTE
                            4111 *
     11D1 D0 87 00          4112 SVA420 B      SVA960(,@BR)              TO 1ST NON-BLANK BYTE
     11D4 6C 00 20 00       4113        MVC    SVALS2(1,@BR),SVADP0(,@XR)  STACK BYTE
                            4114 *
                            4115 * TEST FOR A LETTER-DIGIT VARIABLE REFERENCE
                            4116 *
     11D8 BD F0 00          4117 SVA430 CLI    SVADP0(,@XR),B@DEC0       BYTE A DIGIT ?
     11DB F2 02 36          4118        JNL    SVA480                    YES, SET TAE CODE
                            4119 *
                            4120 * TEST FOR AN ARITHMETIC ARRAY REFERENCE
                            4121 *
     11DE BD 4D 00          4122 SVA440 CLI    SVADP0(,@XR),B@LPAR       BYTE A LEFT PAREN ?
     11E1 F2 81 3D          4123        JE     SVA490                    YES, SET TYPE CODE
                            4124 *
                            4125 * TEST CHARACTER VARIABLE OR ARRAY REFERENCE
                            4126 *
     11E4 BD 5B 00          4127 SVA450 CLI    SVADP0(,@XR),B@CVAR       BYTE A '$' ?
     11E7 F2 81 3D          4128        JE     SVA500                    YES, TEST FOR ARRAY REF
                            4129 *
                            4130 * TEST FOR A KEYWORD OR FUNCTION REFERENCE
                            4131 *
     11EA BD C1 00          4132 SVA460 CLI    SVADP0(,@XR),@CHARA       IF BYTE IS IN STANDARD ALPHABET
     11ED F2 82 06          4133        JL     SVA465                    * TEST FOR KEYWORD OR
     11F0 BD E9 00          4134        CLI    SVADP0(,@XR),@CHARZ       * FUNCTION
     11F3 F2 04 4B          4135        JNH    SVA530                    * REFERENCE
     11F6 BD 7B 00          4136 SVA465 CLI    SVADP0(,@XR),@NUMBR       TEST FOR SPECIAL ALPHABETIC
     11F9 F2 81 45          4137        JE     SVA530                    * CHARACTERS.  IF EOUAL TO
     11FC BD 7C 00          4138        CLI    SVADP0(,@XR),@ASIGN       * ANY, TEST FOR FUNCTION
     11FF F2 81 3F          4139        JE     SVA530                    * REFERENCE OR KEYWORD
     1202 BD 5B 00          4140        CLI    SVADP0(,@XR),@DOLAR       * $ INCLUDED FOR WTC
     1205 F2 81 39          4141        JE     SVA530                    * CONSIDERATIONS
```

```
                                4143 ***************************************************************************
                                4144 * ASSUME A LETTER VARIABLE REFERENCE AND SET TYPE CODE PARAMETER      *
                                4145 ***************************************************************************
                                4146 *
     1208 7D 00 1E              4147 SVA470 CLI    SVAPCT(,@BR),@ZERO          IS PAREN CT 0 ?
     120B F2 81 13              4148        JE     SVA490                      YES, IN A MAT STATEMENT
     120E 7C 01 22              4149        MVI    SVAVTC(,@BR),SVALVC         SET VAR TYPE CODE
     1211 F2 87 2A              4150        J      SVA525                      RETURN

                                4152 ***************************************************************************
                                4153 * SET TYPE CODE AND VARIABLE LENGTH PARAMETERS FOR LETTER-DIGIT VAR    *
                                4154 ***************************************************************************
                                4155 *
     1214 5E 00 23 27           4156 SVA480 ALC    SVALNG(,@BR),SVAIO1(1,@BR)  INCR LNG COUNT
     1218 D0 87 00              4157        B      SVA960(,@BR)                COUNT BLANKS TO WIN-BLANK BYTE
     121B 7C 10 22              4158        MVI    SVAVTC(,@BR),SVALDC         SET VAR TYPE WEI
     121E F2 87 1D              4159        J      SVA525                      RETURN

                                4161 ***************************************************************************
                                4162 * SET TYPE CODE FOR AN ARITHMETIC ARRAY VARIABLE                      *
                                4163 ***************************************************************************
                                4164 *
     1221 7C 08 22              4165 SVA490 MVI    SVAVTC(,@BR),SVANAC         SET VAR TYPE CODE
     1224 F2 87 17              4166        J      SVA525                      RETURN
                                4167 *
                                4168 ***************************************************************************
                                4169 *                                                                       *
                                4170 * CHARACTER REFERENCE PROCESSING ROUTINE                                *
                                4171 *                                                                       *
                                4172 ***************************************************************************
                                4173 *
                                4174 * INCREMENT LINE POINTER TO NEXT NON-BLANK BYTE
                                4175 *
     1227 5E 00 23 27           4176 SVA500 ALC    SVALNG(,@BR),SVAIO1(1,@BR)  INCR INC COUNT
     122B D0 87 00              4177        B      SVA960(,@BR)                TO NEXT NON-BLANK BYTE
                                4178 *
                                4179 * TEST FOR CHARACTER ARRAY REFERENCE
                                4180 *
     122E BD 4D 00              4181 SVA505 CLI    SVADP0(,@XR),B@LPAR         IS BYTE A LEFT PAREN ?
     1231 C0 81 123B            4182        BE     SVA520                      YES, SET CHAR ARRAY TYPE CODE

                                4184 ***************************************************************************
                                4185 *                                                                       *
                                4186 * SET TYPE CODE PARAMETER FOR CHARACTER VARIABLE                        *
                                4187 *                                                                       *
                                4188 ***************************************************************************
     1235 7C 04 22              4189 SVA510 MVI    SVAVTC(,@BR),SVACVC         SET VAR TYPE CODE
     1238 F2 87 03              4190        J      SVA525                      RETURN
```

```
                              4192 *******************************************************************************
                              4193 * SET TYPE CODE PARAMETER FOR A CHARACTER ARRAY VARIABLE              *
                              4194 *******************************************************************************
     123B 7C 02 22            4195 SVA520 MVI    SVAVTC(,@BR),SVACAC        SET VAR TYPE CODE
     123E F2 87 E2            4196 SVA525 J      SVA660                     RETURN

                              4198 *******************************************************************************
                              4199 *                                                                   *
                              4200 * KEYWORD OR FUNCTION REFERENCE DISCRIMINATION ROUTINE               *
                              4201 *                                                                   *
                              4202 *******************************************************************************
                              4203 *
                              4204 * TEST FOR PRESENCE OF AN EMBEDDED KEYWORD - IT IS ASSUMED THAT NO
                              4205 * INTRINSIC FUNCTION NAME BEGINS WITH A KEYWORD IDENTIFIER
                              4206 *
     1241 5D 01 2A 20         4207 SVA530 CLC    SVAKTO(SVAKLN,@BR),SVALS2(,@BR)  KEYWORD 'TO' ?
     1245 F2 81 B7            4208        JE     SVA635                            YES, PROCESS KEYWORD
     1248 5D 01 2C 20         4209        CLC    SVAKST(SVAKLN,@BR),SVALS2(,@BR)  KEYWORD 'STEP' ?
     124C F2 01 07            4210        JNE    SVA535                            NO, GO CHECK 'THEN'
     124F 3C 01 13F7          4211        MVI    SVASSS,@B1                        SET IND FOR POSSIBLE 'STEP'
     1253 F2 87 91            4212        J      SVA630                            GO PROCESS KEYWORD.  MAYBE
     1256 5D 01 2E 20         4213 SVA535 CLC    SVAKTH(SVAKLN,@BR),SVALS2(,@BR)  KEYWORD 'THEN' ?
     125A F2 81 8A            4214        JE     SVA630                            YES, PROCESS KEYWORD
     125D 5D 01 30 20         4215        CLC    SVAKGO(SVAKLN,@BR),SVALS2(,@BR)  KEYWORD 'GOTO' ?
     1261 F2 81 83            4216        JE     SVA630                            YES, PROCESS KEYWORD
                              4217 *
                              4218 * STACK NEXT NON-BLANK BYTE
                              4219 *
     1264 D0 87 00            4220 SVA540 B      SVA960(,@BR)              INCR LINE PT TO NON-BLANK BYTE
     1267 6C 00 21 00         4221 SVA545 MVC    SVALS3(,@BR),SVADP0(1,@XR)  STACK BYTE
                              4222 *
                              4223 * TEST FOR USER DEFINED FUNCTION - IT IS ASSUMED THAT NO
                              4224 * INTRINSIC FUNCTION NAME BEGINS WITH A USER FUNCTION IDENTIFIER
                              4225 *
     126B 5D 01 38 20         4226 SVA550 CLC    SVAFNC(B@LUFN,@BR),SVALS2(,@BR)  USER FUNCTION
     126F F2 81 94            4227        JE     SVA640                            YES, PROCESS USER FUNC.
     1272 5D 02 3B 21         4228        CLC    SVASTR(,@BR),SVALS3(,@BR) STR FUNCTION ?              1-4
     1276 F2 81 8D            4229        JE     SVA640                   TREAT AS USER FUNCTION       1-4
                              4230 *
                              4231 * TEST FOR PRESENCE OF AN INTRINSIC FUNCTION NAME - IT IS ASSUMED THAT
                              4232 * NO INTRINSIC FUNCTION NAME CONTAINS A KEYWORD IDENTIFIER
                              4233 *
     1279 34 02 129E          4234 SVA560 ST     SVA580+@OP1,@XR           SAVE PRESENT LINE PT CADOR
     127D 34 02 131E          4235        ST     SVA650+@OP1,@XR           SAVE PRESENT LINE PT CADDR
     1281 3C 41 128A          4236        MVI    SVA570+@D1,SVAFTD         DISP TO LAST SYM ENTRY
     1285 D2 02 3C            4237 SVA565 LA     SVAIFT(,@BR),@XR          CADDR INTRINSIC FUNC TBL
     1288 E2 02 00            4238 SVA570 LA     *-*(,@XR),@XR             ACCESS SYM ENTRY
     128B 6D 02 21 00         4239        CLC    SVALS3(B@LIFN,@BR),SVADP0(,@XR)  STACKED LETTERS A FUNC
     128F F2 81 89            4240        JE     SVA650                    YES, PROCESS THE FUNC
                              4241 *
                              4242 * DECREMENT TABLE DISP TO NEXT FUNCTION SYMBOL ENTRY
                              4243 *
     1292 1F 00 128A 28       4244        SLC    SVA570+@D1,SVAFIL(1,@BR)  DECR TO NEXT TILL ENTRY
     1297 C0 84 1285          4245        BH     SVA565                    LOOP UNTIL DISP = 0
                              4246 *
                              4247 * TEST FOR THE RANDOM NUMBER FUNCTION SYMBOL
```

```
                               4248 *
    129B C2 02 0000           4249 SVA580 LA    *-*,@XR                    RESTORE LINE PT
    129F 5D 02 21 36          4250        CLC   SVALS3(B@LIFN,@BR),SVARND(,@BR)  THE RANDOM FUNC SYM
    12A3 F2 01 0D             4251        JNE   SVA600                     NO, TEST FOR DET FUNC
                               4252 *
                               4253 * PROCESS RANDOM NUMBER FUNCTION
                               4254 *
    12A6 D0 87 00             4255 SVA590 B     SVA960(,@BR)               TO NEXT NON-BLANK BYTE
    12A9 BD 4D 00             4256        CLI   SVADP0(,@XR),B@LPAR        A LEFT PAREN ?
    12AC F2 81 5A             4257        JE    SVA645                     YES, SCAN FOR A VAR
    12AF C0 87 11C2           4258        B     SVA410                     CONTINUE SCAN
                               4259 *
                               4260 * TEST FOR THE DETERMINANT FUNCTION SYMBOL
                               4261 *
    12B3 5D 02 21 33          4262 SVA600 CLC   SVALS3(B@LIFN,@BR),SVADET(,@BR)  THE DET FUNC SYM
    12B7 F2 01 17             4263        JNE   SVA620                     NO, PROCESS VAR
                               4264 *
                               4265 * PROCESS THE DETERMINANT FUNCTION
                               4266 *
    12BA 76 02 27             4267 SVA610 A     SVAIO1(,@BR),@XR           INCR TO NEXT BYTE
    12BD C0 87 1342           4268        B     SVA900                     TO NEXT ALPHA BYTE
    12C1 34 02 1326           4269        ST    SVA660+@OP1,@XR            SAVE VAR ADDR
    12C5 7C 01 23             4270        MVI   SVALNG(,@BR),SVAVL1        RESET LNG CT TO 1
    12C8 D0 87 00             4271        B     SVA960(,@BR)               DETERMINE LNG OF VAR
    12CB 7C 08 22             4272        MVI   SVAVTC(,@BR),SVANAC        SET VAR TYPE CODE
    12CE F2 87 52             4273        J     SVA660                     RETURN
                               4274 *
                               4275 * ASSUME THAT WE HAVE A SIMPLE LETTER VARIABLE FOLLOWED WITH AN
                               4276 * EMBEDDED STATEMENT KEYWORD
                               4277 *
    12D1 0C 01 12DD 1326      4278 SVA620 MVC   SVA625+@OP1,SVA660+@OP1(@CADDR)  RESTORE PT TO VAR BYTE
    12D7 7C 01 22             4279        MVI   SVAVTC(,@BR),SVALVC        SET VAR TYPE CODE
    12DA C2 02 0000           4280 SVA625 LA    *-*,@XR                    RESTORE VAR ADDR TO PT
    12DE 7C 01 23             4281        MVI   SVALNG(,@BR),SVAVL1        RESET LNG CT TO 1
    12E1 D0 87 00             4282        B     SVA960(,@BR)               DETERMINE LNG OF VAR
    12E4 F2 87 3C             4283        J     SVA660                     RETURN

                               4285 *********************************************************************
                               4286 *                                                                   *
                               4287 * PROCESS EMBEDDED KEYWORD                                           *
                               4288 *                                                                   *
                               4289 *********************************************************************
                               4290 *
    12E7 D0 87 00             4291 SVA630 B     SVA960(,@BR)               INCR TO 3RD LETTER
    12EA 3D 01 13F7           4292        CLI   SVASSS,@B1                 IS 'STEP' POSSIBLE KEYWORD ?
    12EE F2 01 0B             4293        JNE   SVA632                     NO, GO INCR TO 4TH LETTER
    12F1 3C 00 13F7           4294        MVI   SVASSS,@ZERO               SET 'STEP' INDICATOR OFF
    12F5 BD C5 00             4295        CLI   0(,@XR),B@EXPC             DOES CHAR 'E' FOLLOW 'ST' ?
    12F8 C0 01 1267           4296        BNE   SVA545                     IF NOT.  RETURN TO SIMPLE VAR
    12FC D0 87 00             4297 SVA632 B     SVA960(,@BR)               INCR TO 4TH LETTER
    12FF D0 87 00             4298 SVA635 B     SVA960(,@BR)               TO 1ST NON-BLANK BYTE AFTER
                               4299 *                                     * KEYWORD
    1302 C0 87 11C2           4300        B     SVA410                     CONTINUE SCAN
```

```
                              4302 ********************************************************************
                              4303 *                                                                  *
                              4304 * PROCESS USER DEFINED FUNCTION REFERENCE                          *
                              4305 *                                                                  *
                              4306 ********************************************************************
                              4307 *
     1306 D0 87 00           4308 SVA640 B     SVA960(,@BR)              TO LEFT PAREN
     1309 5E 00 1E 27        4309 SVA645 ALC   SVAPCT(1,@BR),SVAIO1(,@BR)  INCR PAREN CT
     130D D0 87 00           4310        B     SVA960(,@BR)              INCR TO NEXT NON-BLNK BYTE
     1310 BD 4D 00           4311        CLI   0(,@XR),B@LPAR            AT LEFT PAREN
     1313 C0 01 11C2         4312        BNE   SVA410                    NO, SCAN FOR VARS
     1317 C0 87 1309         4313        B     SVA645                    YES, INCR PAREN COUNT

                              4315 ********************************************************************
                              4316 *                                                                  *
                              4317 * PROCESS INTRINSIC FUNCTION REFERENCE                             *
                              4318 *                                                                  *
                              4319 ********************************************************************
                              4320 *
     131B C2 02 0000         4321 SVA650 LA    *-*,@XR                   RESTORE LINE PT
     131F C0 87 1306         4322        B     SVA640                    SCAN TO VAR IN PARENS

                              4324 ********************************************************************
                              4325 *                                                                  *
                              4326 * VARIABLE SCAN EXIT ROUTINE                                       *
                              4327 *                                                                  *
                              4328 ********************************************************************
                              4329 *
     1323 C2 02 0000         4330 SVA660 LA    *-*,@XR
     1327 C0 87 0000         4331 SVA670 B     *-*
```

```
                         4333 ******************************************************************************
                         4334 *                                                                            *
                         4335 * ROUTINE TO INCREMENT PAST THE STATEMENT KEYWORD                            *
                         4336 *                                                                            *
                         4337 ******************************************************************************
                         4338 ******************************************************************************
                         4339 *
    132B 34 08 1341      4340 SVA700 ST    SVA750+@OP1,@ARR          SAVE RETURN ADDR
                         4341 *
                         4342 * INCREMENT TO NEXT ALPHA BYTE
                         4343 *
    132F C0 87 1342      4344 SVA710 B     SVA900                    GO TO NEXT ALPHA BYTE
    1333 E2 02 01        4345        LA    1(,@XR),@XR               INCR PT
                         4346 *
                         4347 * DECREMENT KEYWORD LETTER COUNT AND TEST FOR ZERO
                         4348 *
    1336 5F 00 1A 27     4349 SVA720 SLC   SVAKWL(,@BR),SVAIO1(1,@BR)  DECR LETTER CT
    133A C0 84 132F      4350        BH    SVA710                    LOOP UNTIL CT = 0
                         4351 *
                         4352 * RETURN
                         4353 *
    133E C0 87 0000      4354 SVA750 B     *-*                       RETURN
```

```
                              4356 ***********************************************************************
                              4357 *                                                                     *
                              4358 * SCAN STATEMENT LINE FOR FIRST ALPHABETIC BYTE                       *
                              4359 *                                                                     *
                              4360 ***********************************************************************
                              4361 ***********************************************************************
                              4362 *
     1342 34 08 13D2          4363 SVA900 ST    SVA940+@OP1,@ARR          SAVE RETURN ADDR
     1346 7C 40 1D            4364        MVI   SVADSW(,@BR),B@BLNK       INIT DIGIT SW
                              4365 *
                              4366 * TEST FOR INTERNAL CONSTANTS
                              4367 *
     1349 BD 50 00            4368 SVA902 CLI   0(,@XR),B@ICON           AN INTERNAL CON ?
     134C F2 01 18            4369        JNE   SVA910                   NO, TEST FOR LITERAL
                              4370 *
                              4371 * INCREMENT PAST THE INTERNAL CONSTANT
                              4372 *
     134F D0 87 00            4373 SVA904 B     SVA960(,@BR)             TO NEXT LETTER
     1352 BD C5 00            4374        CLI   0(,@XR),B@CIEX           &E
     1355 F2 81 0C            4375        JE    SVA908                   YES, INCR PAST IT
     1358 BD D7 00            4376        CLI   0(,@XR),B@CIPI           &PI
     135B F2 81 03            4377        JE    SVA906                   YES, INCR PAST IT
     135E D0 87 00            4378        B     SVA960(,@BR)             ASSUME @SQR2
     1361 D0 87 00            4379 SVA906 B     SVA960(,@BR)             INCR TO NEXT LETTER
     1364 D0 87 00            4380 SVA908 B     SVA960(,@BR)             INCR TO NEXT LETTER
                              4381 *
                              4382 * TEST FOR LITERAL
                              4383 *
     1367 BD 7D 00            4384 SVA910 CLI   SVADP0(,@XR),B@SQUO      IS BYTE A QUOTE ?
     136A F2 01 11            4385        JNE   SVA920                   NO, CHECK EOS
     136D 76 02 27            4386 SVA915 A     SVAIO1(,@BR),@XR         INCR TO NEXT BYTE
     1370 BD 7D 00            4387        CLI   SVADP0(,@XR),B@SQUO      IS BYTE A QUOTE
     1373 C0 01 136D          4388        BNE   SVA915                   NO, GET NEXT BYTE
     1377 76 02 27            4389        A     SVAIO1(,@BR),@XR         INCR TO NEXT BYTE
     137A C0 87 1349          4390        B     SVA902                   CHECK FOR QUOTE
                              4391 *
                              4392 * TEST FOR EOS
                              4393 *
     137E BD 1E 00            4394 SVA920 CLI   SVADP0(,@XR),@EOS        AT EOS ?
     1381 C0 81 11B4          4395        BE    SVA330                   YES, RETURN
                              4396 *
                              4397 * TEST AND PROCESS PARENS
                              4398 *
     1385 BD 4D 00            4399 SVA925 CLI   0(,@XR),B@LPAR           A LEFT PAREN ?
     1388 F2 01 07            4400        JNE   SVA928                   NO, TEST FOR RT PAREN
     138B 5E 00 1E 27         4401        ALC   SVAPCT(1,@BR),SVAIO1(,@BR)  INCR PAREN COUNT
     138F F2 87 41            4402        J     SVA950                   INCR TO NEXT BYTE
     1392 BD 5D 00            4403 SVA928 CLI   0(,@XR),B@RPAR           A RIGHT PAREN ?
     1395 F2 01 07            4404        JNE   SVA930                   NO, TEST FOR A LETTER
     1398 5F 00 1E 27         4405        SLC   SVAPCT(1,@BR),SVAIO1(,@BR)  DECR PAREN QUIT
     139C F2 87 34            4406        J     SVA950                   TO NEXT BYTE
                              4407 *
                              4408 * TEST FOR ALPHABETIC BYTE
                              4409 *
     139F BD 5B 00            4410 SVA930 CLI   SVADP0(,@XR),@DOLAR      IS BYTE A $ ?
     13A2 F2 81 2A            4411        JE    SVA940                   YES, RETURN
```

```
    13A5 BD 7B 00          4412          CLI     SVADP0(,@XR),@NUMBR      IS BYIE A # ?
    13A8 F2 81 24          4413          JE      SVA940                  YES, RETURN
    13AB BD 7C 00          4414          CLI     SVADP0(,@XR),@ASIGN     IS BYTE A @ ?
    13AE F2 81 1E          4415          JE      SVA940                  YES, RETURN
    13B1 BD C1 00          4416          CLI     SVADP0(,@XR),@CHARA     LT LETTER A ?
    13B4 F2 81 1C          4417          JE      SVA950                  YES, INCR TO NEXT BYTE
    13B7 BD E9 00          4418          CLI     SVADP0(,@XR),@CHARZ     GT LETTER Z ?
    13BA F2 84 16          4419          JH      SVA950                  YES, INCR TO NEXT BYTE
    13BD BD C5 00          4420          CLI     SVADP0(,@XR),B@EXPC     AN EXP ?
    13C0 F2 01 0C          4421          JNE     SVA940                  NO, RETURN
    13C3 7D F0 1D          4422          CLI     SVADSW(,@BR),B@DEC0     PREVIOUS CHAR A DIGIT ?
    13C6 F2 01 0A          4423          JNE     SVA950                  YES, INCR PAST EXP
    13C9 7D 4B 1D          4424          CLI     SVADSW(,@BR),B@DPNT     A DEC POINT ?
    13CC F2 81 04          4425          JE      SVA950                  YES, SKIP EXP
    13CF C0 87 0000        4426 SVA940 B        *-*                      RETURN
                           4427 *
                           4428 * INCREMENT LINE POINTER AND RECYCLE LOOP
                           4429 *
    13D3 6C 00 1D 00       4430 SVA950 MVC      SVADSW(,@BR),SVADP0(,@XR) SAVE PRESENT CHAR
    13D7 D0 87 00          4431          B       SVA960(,@BR)            INCR PT
    13DA C0 87 1349        4432          B       SVA902                  TEST FOR LITERAL
```

```
                           4434 ********************************************************************
                           4435 *                                                                  *
                           4436 * SCAN PAST BLANKS AND KEEP A COUNT                                *
                           4437 *                                                                  *
                           4438 ********************************************************************
                           4439 ********************************************************************
                           4440 *
                           4441 * SAVE RETURN ADDRESS AND ZERO BLANK COUNTER
                           4442 *
    13DE 74 08 0C          4443 SVA960 ST    SVA970+@OP1(,@BR),@ARR    SAVE RETURN ADDR
    13E1 76 02 27          4444        A     SVAIO1(,@BR),@XR          INCR TO NEXT BYTE
                           4445 *
                           4446 * TEST FOR BLANK
                           4447 *
    13E4 BD 40 00          4448 SVA966 CLI   SVADP0(,@XR),@BLANK       AT A BLANK
    13E7 C0 01 0000        4449 SVA970 BNE   *-*                       NO, RETURN
                           4450 *
                           4451 * INCREMENT LINE POINTER AND BLANK COUNT
                           4452 *
    13EB 76 02 27          4453 SVA975 A     SVAIO1(,@BR),@XR          INCR LINE PT
    13EE 5E 00 23 27       4454        ALC   SVALNG(,@BR),SVAIO1(1,@BR)  INCR BLANK CT
    13F2 C0 87 13E4        4455        B     SVA966                    CHECK FOR A BLANK
```

```
      #KSYMB - SYMBOL KEYWORD MODULE
```

```
                           4457 *********************************************************************
                           4458 *                                                                   *
                           4459 * SVARAB WORK AREAS, CONSTANTS, AND EQUATES                          *
                           4460 *                                                                   *
                           4461 *********************************************************************
                           4462 *********************************************************************

                           4464 *
                           4465 * SVRAB EQUATES REFERENCING CONSTANTS
                           4466 *
                0000 4467 SVADP0 EQU   0                         PT DISP OF 0
                0000 4468 SVAOTD EQU   0                         TBL DISP OF 0
                0000 4469 SVAOFF EQU   0                         TEST FOR BR SW OFF
                0001 4470 SVAONN EQU   1                         TO TURN BR SW ON
                0001 4471 SVA1TD EQU   1                         TBL DISP TO BR ADDR
                0001 4472 SVAVL1 EQU   1                         VAR LNG OF 1
                0002 4473 SVA2TD EQU   2                         TBL DISP TO KEYWORD LNG
                0002 4474 SVAKLN EQU   2                         LNG OF EMBEDDED KEYWORD INDR
                0002 4475 SVAVL2 EQU   2                         VAR LNG OF 2
                0003 4476 SVAVL3 EQU   3                         VAR LNG OF 3
                000F 4477 SVAL15 EQU   15                        PAREN COUNT SET SO IT WILL
                           4478 *                                * NOT BECOME 0 IN A SCALAR INST
                0041 4479 SVAFTD EQU   65                        DISP TO LAST BYTE FUNC TABLE
                0080 4480 SVADIS EQU   X'80'                     MASK TO TEST FOR DISABLE INST
                           4481 *
                           4482 * VARIABLE REFERENCE TYPE CODES
                           4483 *
                0001 4484 SVALVC EQU   X'01'                     CODE FOR A LETTER VAR REF
                0010 4485 SVALDC EQU   X'10'                     CODE FOR A LETTER DIGIT VAR REF
                0004 4486 SVACVC EQU   X'04'                     CODE FOR A CHAR VAR REF
                0008 4487 SVANAC EQU   X'08'                     CODE FOR A ARITN ARRAY REF
                0002 4488 SVACAC EQU   X'02'                     CODE FOR A CHAR ARRAY REF
                           4489 *
                           4490 * SVARAB WORK AREAS
                           4491 *
    13F6                 13F6 4492 SVABSW DS    CL1                       BRANCH SWITCH USED TO DETERMINE
    13F6                      4493        ORG   *-1                       A IF THE KEYWORD NEEDS TO BE
    13F6 00              13F6 4494        DC    XL1'00'                   * BY-PASSED. INITIALLY ZERO
    13F7                 13F7 4495 SVASSS DS    XL1                       INDICATOR FUN 'STEP' FUNCTION
    13F7                      4496        ORG   SVASSS                    * '01' -> POSSIBLE 'STEP'. 'ST'
    13F7 00              13F7 4497        DC    XL1'0'                    * HAS BEEN FOUND. '00' -> OFF
    13F8                 13F8 4498 SVAKWL DS    CL1                       KEYWORD LENGTH SAVE AREA.
    13F8                      4499        ORG   SVAKWL                    * INITIALLY SET TO ZERO. 1ST
    13F8 00              13F8 4500        DC    XL1'00'                   * BYTE ALWAYS ZERO
    13F9                 13FA 4501 SVASTC DS    CL2                       STATEMENT TYPE CODE SAVE AREA.
    13F9                      4502        ORG   *-2                       * INITIALLY SET TO ZERO. 1ST
    13F9 0000            13FA 4503        DC    XL2'00'                   * BYTE ALWAYS ZERO
    13FB                 13FB 4504 SVADSW DS    CL1                       DIGIT SW
    13FC                 13FC 4505 SVAPCT DS    CL1                       PARER COUNTER
    13FD                 13FF 4506 SVALSA DS    CL3                       LETTER SAVE AREA
    1400                 1400 4507 SVAVTC DS    CL1                       VARIABLE TYPE CODE SAVE AREA
    1401                 1401 4508 SVALNG DS    CL1                       VAR LNG SAVE AREA
    1402                 1402 4509 SVAMAG DS    CL1                       STR TYPE HOLDER           1-4
    1403                 1403 4510 SVAZRO DS    CL1                       STR ZERO                  1-4
```

```
                               4512 *
                               4513 * SVARAB CONSTANTS
                               4514 *
     1404 0001      1405 4515 SVAIO1 DC    IL2'01'                INTEGER OF 1
     1406 03        1406 4516 SVAFIL DC    IL1'03'                VALUE TO DECR FUNC TBL PT
                               4517 *
                               4518 * EMBEDDED STATEMENT KEYWORD IDENTIFIERS
                               4519 *
     1407 E3D6      1408 4520 SVAKTO DC    CL2'TO'                IDENTIFIER FOR KEYWORD 'TO'
     1409 E2E3      140A 4521 SVAKST DC    CL2'ST'                IDENTIFIER FOR KEYWORD 'STEP'
     140B E3C8      140C 4522 SVAKTH DC    CL2'TH'                IDENTIFIER FOR KEYWORD 'THEN'
     140D C7D6      140E 4523 SVAKGO DC    CL2'GO'                IDENTIFIER FOR KEYWORD *GOTO'
                               4524 *
                               4525 * DETERMINANT AND RANDOM NUMBER FUNCTION IDENTIFIERS
                               4526 *
     140F C4C5E3    1411 4527 SVADET DC    CL(B@LIFN)'DET'        DETERMINANT FUNCTION IDENTIFIER
     1412 D9D5C4    1414 4528 SVARND DC    CL(B@LIFN)'RND'        RND NUMBER FUNC IDENTIFIER
                               4529 *
                               4530 * USER DEFINED FUNCTION IDENTIFIER
                               4531 *
     1415 C6D5      1416 4532 SVAFNC DC    CL2'FN'                USER FUNCTION IDENTIFIER
     1417 E2E3D9    1419 4533 SVASTR DC    CL(B@LIFN)'STR'                                   1-4
                               4534 *
                               4535 * INTRINSIC FUNCTION TABLE
                               4536 *
                     141A 4537 SVAIFT EQU   *                     ADDR INTRINSIC FUNCTION TABLE
     141A C1C2E2    141C 4538        DC    CL(B@LIFN)'ABS'        FUNCTION SYMBOL FOR ABSOLUTE
     141D C9D5E3    141F 4539        DC    CL(B@LIFN)'INT'        FUNCTION SYMBOL FOR INTEGER
     1420 E2C7D5    1422 4540        DC    CL(B@LIFN)'SGN'        FUNCTION SYMBOL FOR SIGN
     1423 E2D8D9    1425 4541        DC    CL(B@LIFN)'SQR'        FUNCTION SYMBOL FOR SQ ROOT
     1426 D3D6C7    1428 4542        DC    CL(B@LIFN)'LOG'        FUNCTION SYMBOL FOR LOG E
     1429 D3C7E3    142B 4543        DC    CL(B@LIFN)'LGT'        FUNCTION SYMBOL FOR LOG 10
     142C D3E3E6    142E 4544        DC    CL(B@LIFN)'LTW'        FUNCTION SYMBOL FOR LOC 2
     142F C5E7D7    1431 4545        DC    CL(B@LIFN)'EXP'        FUNCTION SYMBOL FOR EXPONENTIAL
     1432 E3C1D5    1434 4546        DC    CL(B@LIFN)'TAN'        FUNCTION SYMBOL FOR TSNGENT
     1435 C3D6E3    1437 4547        DC    CL(B@LIFN)'COT'        FUNCTION SYMBOL FOR COTANGENT
     1438 E2C9D5    143A 4548        DC    CL(B@LIFN)'SIN'        FUNCTION SYMBOL FOR SINE
     143B C3D6E2    143D 4549        DC    CL(B@LIFN)'COS'        FUNCTION SYMBOL FOR COSINE
     143E E2C5C3    1440 4550        DC    CL(B@LIFN)'SEC'        FUNCTION SYMBOL FOR SECANT
     1441 C3E2C3    1443 4551        DC    CL(B@LIFN)'CSC'        FUNCTION SYMBOL FOR COSECANT
     1444 C1E3D5    1446 4552        DC    CL(B@LIFN)'ATN'        FUNCTION SYMBOL FOR ARCTANGENT
     1447 C1E2D5    1449 4553        DC    CL(B@LIFN)'ASN'        FUNCTION SYMBOL FOR ANCSINE
     144A C1C3E2    144C 4554        DC    CL(B@LIFN)'ACS'        FUNCTION SYMBOL FOR ARCCOSINE
     144D C8E3D5    144F 4555        DC    CL(B@LIFN)'HTN'        HYPERBOLIC MODEM FUNC SYM
     1450 C8E2D5    1452 4556        DC    CL(B@LIFN)'HSN'        HYPERBOLIC SINE FUNC SYM
     1453 C8C3E2    1455 4557        DC    CL(B@LIFN)'HCS'        HYPERBOLIC COSINE FUNC SYM
     1456 C4C5C7    1458 4558        DC    CL(B@LIFN)'DEG'        CONVERT RAD TO DEG FUNC SYM
     1459 D9C1C4    145B 4559        DC    CL(B@LIFN)'RAD'        CONVERT DEG TO RAD FUMC SYM
                               4560 *
                               4561 * INVERSE AND TRANSPOSE IDENTIFIERS
                               4562 *
     145C C9D5E5    145E 4563 SVAINV DC    CL(B@LIFN)'INV'        FUNCTION SYMBOL FOR INVERSE
     145F E3D9D5    1461 4564 SVATRN DC    CL(B@LIFN)'TRN'        FUNCTION SYMBOL FOR TRANSPOSE
```

```
                             4566 *
                             4567 * IDENTIFIERS FOR CONSTANT, IDENTITY AND ZERO
                             4568 *
     1462 C3D6D5        1464 4569 SVACON DC    CL(B@LIFN)'CON'          FUNCTION SYMBOL FOR CONSTANT
     1465 C9C4D5        1467 4570 SVAIDN DC    CL(B@LIFN)'IDN'          FUNCTION SYMBOL FOR IDENTITY
     1468 E9C5D9        146A 4571 SVAZER DC    CL(B@LIFN)'ZER'          FUNCTION SYMBOL FOR ZERO
                             4572 *
                             4573 * SVARAB EQUATES REFERENCING PROGRAM
                             4574 *
                        13FD 4575 SVALS1 EQU   SVALSA-2                 1ST LETTER SAVE BYTE
                        13FE 4576 SVALS2 EQU   SVALSA-1                 2ND LETTER SAVE BYTE
                        13FF 4577 SVALS3 EQU   SVALSA                   3RD LETTER SAVE BYTE
```

```
                              4579 ********************************************************************
                              4580 ********************************************************************
                              4581 *                                                                  *
                              4582 * SVARAB DISTRIBUTOR TABLE                                         *
                              4583 *                                                                  *
                              4584 ********************************************************************
                              4585 ********************************************************************
                              4586 *
                         146B 4587 SVABRT EQU   *                           CADDR 1ST BYTE DISTRIBUTLA TBL
                              4588 *
    146B 03               146B 4589        DC    AL1(B@LREM)                LNG OF KEYWORD REM
    146C 1021             146D 4590        DC    AL(@CADDR)(SVA050)         RTN FOR LINES WITH NO YARS
                              4591 *
    146E 04               146E 4592        DC    AL1(B@LDAT)                LNG OF KEYWORD DATA
    146F 1021             1470 4593        DC    AL(@CADDR)(SVA050)         RTN FOR LINES WITH NO VARS
                              4594 *
    1471 03               1471 4595        DC    AL1(B@LDEF)                LNG OF KEYWORD DEF
    1472 115B             1473 4596        DC    AL(@CADDR)(SVA200)         RTN FOR REMAINING NON-MATS
                              4597 *
    1474 03               1474 4598        DC    AL1(B@LDIM)                INC OF KEYWORD DIN
    1475 115B             1476 4599        DC    AL(@CADDR)(SVA200)         RTN FOR REMAINING NON-MATS
                              4600 *
    1477 03               1477 4601        DC    AL1(B@LLET)                LNG OF KEYWORD LET
    1478 115B             1479 4602        DC    AL(@CADDR)(SVA200)         RTN FOR REMAINING NON-MATS
                              4603 *
    147A 00               147A 4604        DC    XL1'00'                    LNG ASSIGN SIMPLE
    147B 1030             147C 4605        DC    AL(@CADDR)(SVA070)         RTN FOR ASSIGNMENT SIMLE
                              4606 *
    147D 03               147D 4607        DC    AL1(B@LLET)                LNG OF KEYWORD LET
    147E 115B             147F 4608        DC    AL(@CADDR)(SVA200)         RTN FOR REMAINING U-MATS
                              4609 *
    1480 00               1480 4610        DC    XL1'00'                    LNG ASSIGN SIMPLE
    1481 1030             1482 4611        DC    AL(@CADDR)(SVA070)         RTN FOR ASSIGNMENT SIMPLE
                              4612 *
    1483 03               1483 4613        DC    AL1(B@LLET)                LNG OF KEYWORD LET
    1484 115B             1485 4614        DC    AL(@CADDR)(SVA200)         RTN FOR REMAINING U-MATS
                              4615 *
    1486 00               1486 4616        DC    XL1'00'                    LNG ASSIGN SIMPLE
    1487 1030             1488 4617        DC    AL(@CADDR)(SVA070)         RTN FOR ASSIGNMD4T SIMPLE
                              4618 *
    1489 03               1489 4619        DC    AL1(B@LKFR)                LNG OF KEYWORD FOR
    148A 115B             148B 4620        DC    AL(@CADDR)(SVA200)         RTN FOR REMAINING NON-MATS
                              4621 *
    148C 04               148C 4622        DC    AL1(B@LNEX)                LNG OF KEYWORD NM
    148D 115B             148E 4623        DC    AL(@CADDR)(SVA200)         RTN FOR REMAINING NON-MATS
                              4624 *
    148F 02               148F 4625        DC    AL1(B@LKIF)                LNG OF KEYWORD IF
    1490 115B             1491 4626        DC    AL(@CADDR)(SVA200)         RTN FOR REMAINING ION-MATS
                              4627 *
    1492 02               1492 4628        DC    AL1(B@LKIF)                LNG OF KEYWORD IF
    1493 115B             1494 4629        DC    AL(@CADDR)(SVA200)         RTN FOR REMAINING NON-MATS
                              4630 *
    1495 04               1495 4631        DC    AL1(B@LGTO)                LNG OF KEYWORD GOTO SIMPLE
    1496 115B             1497 4632        DC    AL(@CADDR)(SVA200)         RTN FOR REMAINING NON-MATS
                              4633 *
    1498 04               1498 4634        DC    AL1(B@LGTO)                LNG OF KEYWORD GOTO
```

```
     1499 106A          149A 4635      DC    AL(@CADDR)(SVA090)     RTN FOR COMPUTED GOTO
                             4636 *
     149B 05            149B 4637      DC    AL1(B@LGSB)            LNG OF KEYWORD GO SUB
     149C 1021          149D 4638      DC    AL(@CADDR)(SVA050)     RTN FOR LINES WITH NO VARS
                             4639 *
     149E 06            149E 4640      DC    AL1(B@LRTN)            LNG OF KEYWORD RETURN
     149F 1021          14A0 4641      DC    AL(@CADDR)(SVA050)     RTN FOR LINES WITH NO VARS
                             4642 *
     14A1 03            14A1 4643      DC    AL1(B@LKGT)            LNG OF KEYWORD GET
     14A2 112C          14A3 4644      DC    AL(@CADDR)(SVA170)     RTN FOR GET AND PUT
                             4645 *
     14A4 03            14A4 4646      DC    AL1(B@LKPT)            LNG OF KEYWORD PUT
     14A5 112C          14A6 4647      DC    AL(@CADDR)(SVA170)     RTN FOR GET ANF PUT
                             4648 *
     14A7 05            14A7 4649      DC    AL1(B@LKRT)            LNG OF KEYWORD RESET
     14A8 112C          14A9 4650      DC    AL(@CADDR)(SVA170)     RTN TO PROCESS VAR FILE REF
                             4651 *
     14AA 05            14AA 4652      DC    AL1(B@LKCL)            LNG OF KEYWORD CLOSE
     14AB 112C          14AC 4653      DC    AL(@CADDR)(SVA170)     RTN TO PROCESS VAR FILE REF
                             4654 *
     14AD 05            14AD 4655      DC    AL1(B@LINP)            LNG OF KEYWORD INPUT
     14AE 115B          14AF 4656      DC    AL(@CADDR)(SVA200)     RTN FOR REMAINING NON-MATS
                             4657 *
     14B0 04            14B0 4658      DC    AL1(B@LREA)            LNG OF KEYWORD READ
     14B1 115B          14B2 4659      DC    AL(@CADDR)(SVA200)     RTN FOR REMAINING NON-MATS
                             4660 *
     14B3 07            14B3 4661      DC    AL1(B@LKRR)            LNG OF KEYWORD RESTORE
     14B4 1021          14B5 4662      DC    AL(@CADDR)(SVA050)     RTN FOR LINES WITH NO VARS
                             4663 *
     14B6 05            14B6 4664      DC    AL1(B@LPRT)            LNG OF KEYWORD PRINT
     14B7 115B          14B8 4665      DC    AL(@CADDR)(SVA200)     RTN FOR REMAINING NON-MATS
                             4666 *
     14B9 0A            14B9 4667      DC    AL1(B@LKPU)            LNG OF KEYWORD PRINT USING
     14BA 115B          14BB 4668      DC    AL(@CADDR)(SVA200)     RTN FOR REMAINING NON-MATS
                             4669 *
     14BC 01            14BC 4670      DC    AL1(B@LIMG)            LNG OF KEYWORD IMAGE
     14BD 1021          14BE 4671      DC    AL(@CADDR)(SVA050)     RTN FOR LINES WITH NO VARS
                             4672 *
     14BF 03            14BF 4673      DC    AL1(B@LMAT)            LNG OF KEYWORD MAT
     14C0 108E          14C1 4674      DC    AL(@CADDR)(SVA120)     RTN FOR MAT ASSIGNMENT
                             4675 *
     14C2 06            14C2 4676      DC    AL1(B@LMGT)            LNG OF KEYWORD MAT GET
     14C3 1173          14C4 4677      DC    AL(@CADDR)(SVA250)     RTN FOR REMAINING MAT STMNTS
                             4678 *
     14C5 08            14C5 4679      DC    AL1(B@LMIN)            LNG OF KEYWORD MAT INPUT
     14C6 1144          14C7 4680      DC    AL(@CADDR)(SVA192)     RTN FOR REMAINING MAT STMNTS
                             4681 *
     14C8 07            14C8 4682      DC    AL1(B@LMRD)            LNG OF KEYWORD MAT READ
     14C9 1144          14CA 4683      DC    AL(@CADDR)(SVA192)     RTN FOR REMAINING MAT STINTS
                             4684 *
     14CB 06            14CB 4685      DC    AL1(B@LMPT)            LNG OF KEYWORD MAT PUT
     14CC 1173          14CD 4686      DC    AL(@CADDR)(SVA250)     RTN FOR REMAINING MAT STMNTS
                             4687 *
     14CE 08            14CE 4688      DC    AL1(B@LMPR)            LNG OF KEYWORD MAT PRINT
     14CF 1144          14D0 4689      DC    AL(@CADDR)(SVA192)     RTN FOR REMAINING MAT STMNTS
                             4690 *
```

```
      #KSYMB - SYMBOL KEYWORD MODULE


ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT            VER 15, MOD 00  04/07/22  PAGE  49

    14D1 0D              14D1 4691        DC    AL1(B@LMPU)             LNG OF KEYWORD MAT PRINT USING
    14D2 1144            14D3 4692        DC    AL(@CADDR)(SVA192)      RTN FOR REMAINING MAT STMNTS
                              4693 *
    14D4 05              14D4 4694        DC    AL1(B@LPSE)             LNG OF KEYWORD PAUSE
    14D5 1021            14D6 4695        DC    AL(@CADDR)(SVA050)      RTN FOR LINES WITH NO VARS
                              4696 *
    14D7 04              14D7 4697        DC    AL1(B@LSTP)             LNG OF KEYWORD STOP
    14D8 1021            14D9 4698        DC    AL(@CADDR)(SVA050)      RTN FOR LINES WITH NO VARS
                              4699 *
    14DA 03              14DA 4700        DC    AL1(B@LEND)             LNG OF KEYWORD END
    14DB 1021            14DC 4701        DC    AL(@CADDR)(SVA050)      RTN FOR LINES WITH NO VARS
                              4702 *
    14DD                 14DD 4703        DS    CL1                     DUMMY BYTE
    14DE 1021            14DF 4704        DC    AL(@CADDR)(SVA050)      RTN FOR LINES WITH NO VARS
                              4705 *
    14E0                 14E0 4706        DS    CL1                     DUMMY BYTE
    14E1 1021            14E2 4707        DC    AL(@CADDR)(SVA050)      RTN FOR LINES KITH NO VARS
                              4708 *
    14E3 03              14E3 4709        DC    AL1(B@LLET)             LNG OF LET(STR)            1-4
    14E4 115B            14E5 4710        DC    AL(@CADDR)(SVA200)      RTN (STR)---SIMPLE         1-4
                              4711 *                                                              1-4
    14E6 03              14E6 4712        DC    AL1(B@LLET)             LNG OF LET(STR)            1-4
    14E7 115B            14E8 4713        DC    AL(@CADDR)(SVA200)      RTN (STR)--MULTIPLE        1-4
                              4714 *                                                              1-4
    14E9 00              14E9 4715        DC    XL1'00'                 LNG OF ASSIGN(STR)         1-4
    14EA 1030            14EB 4716        DC    AL(@CADDR)(SVA070)      RTN (STR)---SIMPLE         1-4
                              4717 *                                                              1-4
    14EC 00              14EC 4718        DC    XL1'00'                 LNG OF ASSIGN(STR)         1-4
    14ED 1030            14EE 4719        DC    AL(@CADDR)(SVA070)      RTN (STR)---MULTIPLE       1-4
                              4720 *                                                              1-4
    14EF 02              14EF 4721        DC    AL1(B@LKIF)             LNG OF IF(STR)             1-4
    14F0 115B            14F1 4722        DC    AL(@CADDR)(SVA200)      RTN (STR) IF STATEMENT     1-4
                              4723 *                                                                *
                              4724 ***********************************************************************
                              4725 ***********************************************************************
                              4726 *
                         1E00 4727 GRBFR1 EQU   X'1E00'                 LABEL FOR FIRST BRAGIT BUFFER
                              4728 *
```

```
                    4730 ***********************************************************************
                    4731 * 5703-XM1      COPYRIGHT IBM CORP. 1970                              *
                    4732 *                        REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083   *
                    4733 *                                                                     *
                    4734 ***********************************************************************
                    4735 *STATUS                                                               *
                    4736 *    VERSION 1 MODIFICATION 0                                          *
                    4737 *                                                                     *
                    4738 *FUNCTION                                                             *
                    4739 *    GRABIT LOCATES SEQUENTIAL STATEMENTS IN THE FILE SPECIFIED BY THE *
                    4740 *    USER, AND, DEPENDING UPON THE OPTION CHOSEN, PASSES BACK THE      *
                    4741 *    STATEMENT OR SKIPS TO THE NEXT.                                  *
                    4742 *    AFTER BEING PRIMED BY THE CALLING PROGRAM, GRABIT READS LOGICALLY *
                    4743 *    CONSECUTIVE BLOCKS OF SEGMENTED STATEMENTS, FROM THE FILE          *
                    4744 *    SPECIFIED BY THE USER, INTO CORE.  GRABIT RETURNS WITH @XR        *
                    4745 *    POINTING TO THE BINARY LINE NUMBER OF THE NEXT STATEMENT.         *
                    4746 *    IN ADDITION TO @XR, GRABIT PARAMETERS CAN BE SET TO CAUSE THE     *
                    4747 *    BINARY LINE NR, THE TYPE CODE AND THE UNPACKED, NON-SEGMENTED     *
                    4748 *    TEXT OF THE NEXT STMT TO BE PLACED IN AREAS DEFINED BY THE USER.  *
                    4749 *    IF GRABIT IS USED TO SKIP THROUGH THE STMTS WITHOUT UNPACKING     *
                    4750 *    THEM OR CHANGING THEIR LENGTH OR SEGMENTED CONDITION, GRABIT CAN  *
                    4751 *    BE INSTRUCTED TO RETURN THE BLOCKS TO THEIR ORIGINAL DISK ADDRESS *
                    4752 *    IF THE SPECIFIED FILE IS ACCESSED BY DL4ICS.                      *
                    4753 *                                                                     *
                    4754 *NOTES                                                                *
                    4755 *    THIS VERSION OF GRABIT USES ONLY DL4ICS TO ACCESS THE NEXT DATA   *
                    4756 *    BLOCK.  GRABIT IN THE SUBROUTINE LIBRARY USES DL4ICS AND DL2ICS.  *
                    4757 ***********************************************************************
```

```
      GRABIT - RETRIEVE FILE STATEMENTS


ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT          VER 15, MOD 00  04/07/22  PAGE  51

                          15D0 4759          USING GRABSE,@BR
                          14F2 4760 GRABIT EQU    *                      ENTRY POINT TO ROUTINE
     14F2 34 01 156F           4761        ST     GRASBR,@BR             SAVE CALLING PROG'S BASE REG.
     14F6 C2 01 15D0           4762        LA     GRABSE,@BR             LOAD LOCAL BASE TO BASE REG.
     14FA 34 08 1573           4763        ST     GRASAR,@ARR            SAVE RETURN ADDR.
     14FE 7D 00 A7             4764        CLI    GRWHAT(,@BR),GRAEFI    IS FUNC REQ'D INITIALIZATION ?
     1501 F2 81 13             4765        JE     GRA100                 YES, GO TO INITIALIZATION RTN
                               4766 * THE ADDRESS OF THE NEXT SEGMENT IN THE CURRENT BUFFER IS INITLZ'D
                               4767 * AND MAINTAINED IN THE NEXT INST, WHICH LOADS IT TO THE @XR.
     1504 C2 02 0000           4768 GRA020 LA     *-*,@XR                LOAD NEXT STMNT CADDR TO @XR
     1508 7D 01 A7             4769        CLI    GRWHAT(,@BR),GRAEFR    IS FUNC REQ'D RETURN TEXT ?
     150B F2 81 87             4770        JE     GRA300                 YES, GO RETURN STMNT ROUTINE
     150E 7D 02 A7             4771        CLI    GRWHAT(,@BR),GRAEFS    IS FUNC REQ'D SKIP STATEMENT
     1511 F2 81 35             4772        JE     GRA200                 YES, GO TO SKIP STMNT ROUTINE
     1514 F2 87 38             4773        J      GRA210                 GO TO SKIP SEGMENT RTN
                               4774 *
                               4775 *                INITIALIZATION ROUTINE
                               4776 *
     1517 75 02 A0             4777 GRA100 L      GRBFRA(,@BR),@XR       LOAD 1ST BFR ADDR TO DB
     151A 74 02 A6             4778        ST     GRANCA(,@BR),@XR       PROPIGATE IT TO NEXT BFR DPL
     151D 5C 01 A3 9D          4779        MVC    GRANDA(@DADDR,@BR),GRSRDA(,@BR)  INITLZ NEXT BRF DADDR
     1521 7C FF AC             4780        MVI    GRASIZ(,@BR),GRAEBS    INITLZ BUFFER SIZE COUNTER
     1524 5C 00 9E A4          4781        MVC    GRACSC(1,@BR),GRSCTR(,@BR)  INITLZ SCTR COUNT IN DPL
     1528 C0 87 0025           4782        B      $DISKN                 WAIT FOR FIRST DATA BLOCKS TO
     152C 057F          152D 4783        DC     AL2($WAITF)            * GET INTO CORE
     152E 7C 97 B5             4784        MVI    GRAERR+@Q(,@BR),@@E550 SET ERR CODE TO SPECIFY WRKFILE
     1531 5E 01 A6 A9          4785        ALC    GRANCA(@CADDR,@BR),GRASSZ(,@BR)  SET CADDR OF NEXT BFR
     1535 BD 00 00             4786 GRA140 CLI    GRAELK(,@XR),GRAELN    IS 1ST DB LINK CODE = 0 ?
     1538 F2 81 07             4787        JE     GRA150                 YES, GO INCR TO NEXT LOGICAL DB
     153B 7C 02 A3             4788        MVI    GRANDA(,@BR),GRAEDB    SET DADDR OF NEXT DB
     153E 6E 00 A3 00          4789        ALC    GRANDA(1,@BR),GRAELK(,@XR)  *
     1542 5E 00 A3 AB          4790 GRA150 ALC    GRANDA(1,@BR),GRANPB(,@BR)  INCR TO NEXT BFR DADDR
     1546 F2 87 2E             4791        J      GRA260                 GO ACCESS FIRST STATEMENT
                               4792 *
                               4793 *                ACCESS NEXT STATEMENT OR NEXT SEGMENT ROUTINE
                               4794 *
     1549 BD 75 07             4795 GRA200 CLI    GRAEDT(,@XR),GRAEET    END-OF-FILE RECORD ?
     154C F2 81 16             4796        JE     GRA230                 YES, RESET OR TO THIS RECORD
     154F 6F 00 AC 02          4797 GRA210 SLC    GRASIZ(1,@BR),GRAES1(,@XR)  DECR BFR CT BY SEGMENT LENGTH
     1553 B6 02 02             4798        A      GRAES1(,@XR),@XR       INCR OR BY SEGMENT LENGTH
     1556 7D 00 AC             4799 GRA220 CLI    GRASIZ(,@BR),@ZERO     IS BUFFER EMPTY ?
     1559 D0 82 B4             4800        BL     GRAERR(,@BR)           GONE NEG, GO TO BAD ERR
     155C F2 81 15             4801        JE     GRA250                 YES, GO TO GET NEXT BFR
     155F BD 80 01             4802        CLI    GRAES0(,@XR),@SNULL    IS SEGMENT NULL ?
     1562 F2 81 0F             4803        JE     GRA250                 YES, GO TO GET NEXT BFR
     1565 34 02 1507           4804 GRA230 ST     GRA020+@OP1,@XR        SAVE CADDR OF NEXT SEG.IN INST.
     1569 E2 02 06             4805        LA     GRAEDL(,@XR),@XR       POINT @XR TO LINE NUMBER
     156C C2 01 0000           4806 GRA240 LA     *-*,@BR                RESTORE THE BASE REGISTER
                          156F 4807 GRASBR EQU    GRA240+@OP1            * STORED IN INST AT GRA240
     1570 C0 87 0000           4808 GRA245 B      *-*                    RETURN TO USER
                          1573 4809 GRASAR EQU    GRA245+@OP1            * TO CADDR SAVED IN GRA245
     1574 D0 87 67             4810 GRA250 B      GRA500(,@BR)           ACCESS NEXT BUFFER
     1577 BD 80 01             4811 GRA260 CLI    GRAES0(,@XR),@SNULL    IS 1ST SEG. NULL ?
     157A D0 81 B4             4812        BE     GRAERR(,@BR)           YES, GO TO BAD ERR
     157D B9 02 03             4813        TBF    GRAES2(,@XR),GRAETP    PRIMARY SEGMENT
     1580 C0 10 1565           4814        BT     GRA230                 YES, SAVE LOCATION
```

```
     1584 7D 01 A7          4815        CLI   GRWHAT(,@BR),GRAEFR    ACTION REQ'D = RETURN TEXT ?
     1587 D0 81 B4          4816        BE    GRAERR(,@BR)          YES, GO TO BAD ERR
     158A 7D 04 A7          4817        CLI   GRWHAT(,@BR),GRAEFG   ACTION REQ'D = SKIP SEGMENT ?
     158D C0 81 1565        4818        BE    GRA230                YES, GO SAVE LOCATION
     1591 C0 87 154F        4819        B     GRA210                NO, GO SKIP THIS SEGMENT
                            4820 *
                            4821 *                 RETURN TEXT ROUTINE
                            4822 *
     1595 2C 01 1B97 06     4823 GRA300 MVC   GRLINE,GRAEDL(GRAELL,@XR)  SET BINARY LINE NO.IN O/P FIELD
     159A 2C 00 1B96 07     4824        MVC   GRTYPE,GRAEDT(1,@XR)    SET TYPE CODE IN OUTPUT FIELD
     159F 4C 01 58 167E     4825        MVC   GRTEND(@CADDR,@BR),GRATXT  INITLZ TEXT 0/P CADDR IN INST.
     15A4 BD 75 07          4826        CLI   GRAEDT(,@XR),GRAEET   END OF FILE STATEMENT ?
     15A7 F2 01 08          4827        JNE   GRA303                NO - GO RESET SEGMENT SWITCH
     15AA 3C 1C 1B98        4828        MVI   GRTEXT,@EOF           MOVE EOF CODE TO GRTEXT
     15AE C0 87 1565        4829        B     GRA230                GO GET OUT

     15B2 7C 87 01          4831 GRA303 MVI   GRA310+@Q(,@BR),@UCB   INITLZ BRANCH FOR ONLY SEGMENT
     15B5 BD 00 03          4832        CLI   GRAES2(,@XR),@SONLY    IS IT AN ONLY SEGMENT ?
     15B8 F2 81 03          4833        JE    GRA305                YES, BYPASS BRANCH RESET
     15BB 7C 80 01          4834        MVI   GRA310+@Q(,@BR),@NOP   SET FOR MORE SEGMENTS
     15BE 6F 00 AC 02       4835 GRA305 SLC   GRASIZ(1,@BR),GRAES1(,@XR)  DECR BFR CT BY SEG LENGTH
     15C2 9F 00 02 B0       4836        SLC   GRAES1(1,@BR),GRAPSG(,@BR)  DECR SEG CT BY SDF-HDR LENGTH
     15C6 6C 00 B3 02       4837        MVC   GRASEG(1,@BR),GRAES1(,@XR)  MOVE TEXT LENGTH TO TEXT CTR
     15CA E2 02 07          4838        LA    GRAELP(,@XR),@XR       INCR TO TYPE CODE
     15CD F2 87 2A          4839        J     GRA317                GO TEST FILE TYPE
     15D0 C0 87 1556        4840 GRA310 B     GRA220                GO ACCESS NEXT STATEMENT
     15D0                   4841        ORG   GRA310                * UNLESS CURRENT STATEMENT
     15D0 C0 87 1556        4842        BC    GRA220,@UCB           * HAS MORE SEGMENTS
     15D4 6C 00 24 00       4843        MVC   GRASVC(,@BR),@ZERO(1,@XR) SAVE CURR CHAR IN RESTORE INST
     15D8 D0 87 67          4844        B     GRA500(,@BR)          ACCESS NEXT BUFFER
     15DB BD 02 03          4845        CLI   GRAES2(,@XR),@SLAST   LAST SEGMENT ?
     15DE F2 01 03          4846        JNE   GRA313                NO, GO RESET SEG COUNTER
     15E1 7C 87 01          4847        MVI   GRA310+@Q(,@BR),@UCB  RESET BRANCH OUT
     15E4 6F 00 AC 02       4848 GRA313 SLC   GRASIZ(1,@BR),GRAES1(,@XR)  DECR BUFFER COUNTER
     15E8 9F 00 02 B2       4849        SLC   GRAES1(1,@XR),GRASSG(,@BR)  DECR SEG COUNT BY SDF LENGTH
     15EC 6C 00 B3 02       4850        MVC   GRASEG(1,@BR),GRAES1(,@XR)  MOVE TEXT LNG TO SEG COUNTER
     15F0 E2 02 04          4851        LA    GRAELS(,@XR),@XR       INCR @XR PAST SECONDARY SDF
     15F3 BC 00 00          4852 GRA315 MVI   @ZERO(,@XR),*-*        RESTORE CHAR SAVED IN Q-CODE
          15F4 4853 GRASVC  EQU   GRA315+@Q              SAVED CHAR HOLD AREA
     15F6 5E 01 58 AB       4854 GRA316 ALC   GRTEND(@CADDR,@BR),GRABOA(,@BR)  INCR RECEIVING CADDR
          15FA 4855 GRA317  EQU   *                      MOVE TEXT TO GRTEXT
     15FA 38 80 03D4        4856        TBN   $INDR1,$BASIC         IS FILE TYPE = BASIC ?
     15FE F2 90 24          4857        JF    GRA350                NO, BYPASS REPITION CODE CHECK
     1601 BD 1B 01          4858        CLI   GRAENC(,@XR),GRAEMR   IS CHAR REF A REPITITION CODE ?
     1604 F2 84 1E          4859        JH    GRA350                NO, GO RETURN REF'D CHAR
     1607 5C 01 3E 58       4860        MVC   GRATND(@CADDR,@BR),GRTEND(,@BR)  SET RCV'G CADDR IN INSTR
     160B 2C 00 0000 00     4861 GRA320 MVC   *-*,@ZERO(1,@XR)       RETURN REPEATED CHAR TO OUTPUT
          160E 4862 GRATND  EQU   GRA320+@OP1            * ADDR SUPPLIED
     1610 9F 00 01 AB       4863        SLC   GRAENC(1,@XR),GRAONE(,@BR)  DECR. REPITITION COUNTER
     1614 F2 01 07          4864        JNZ   GRA330                IF <> 0, GO INCR O/P CADDR
     1617 5C 01 58 3E       4865        MVC   GRTEND(@CADDR,@BR),GRATND(,@BR)  RESTORE NEW O/P CADDR
     161B F2 87 0C          4866        J     GRA360                GO INCR @XR
     161E 5E 01 3E AB       4867 GRA330 ALC   GRATND(@CADDR,@BR),GRABOA(,@BR) INCR O/P CADDR IN INSTR
     1622 D0 87 3B          4868        B     GRA320(,@BR)          GO MOVE CHAR TO OUTPUT
     1625 2C 00 0000 01     4869 GRA350 MVC   *-*,GRAENC(1,@XR)     MOVE NON-REPEAT CHAR TO OUTPUT
          1628 4870 GRTEND  EQU   GRA350+@OP1           * ADDR SUPPLIED
```

```
      162A E2 02 01              4871 GRA360 LA     GRAENC(,@XR),@XR          INCR @XR TO NEXT CHAR.
      162D 5F 00 B3 AB           4872        SLC    GRASEG(1,@BR),GRABOA(,@BR)  DECR BFR SPACE CTR
      1631 D0 81 00              4873        BZ     GRA310(,@BR)              NO MORE TEXT IN SEG, CHK MORE
      1634 D0 87 26              4874        B      GRA316(,@BR)              MORE TEXT, GO INCR RECV CADDR
                                 4875 *
                                 4876 *              ACCESS NEXT BUFFER ROUTINE
                                 4877 *
      1637 74 08 9A              4878 GRA500 ST     GRA5SA(,@BR),@ARR
      163A C0 87 0025            4879        B      $DISKN                    WAIT FOR PRIOR READ TO COMPLETE
      163E 057F             163F 4880        DC     AL2($WAITF)               *
                            1640 4881 GRA600 EQU    *
                                 4882 *
                                 4883 *              DL4ICS BEING USED - ACCESS NEXT DATA BLOCK
                                 4884 *
      1640 75 02 A0              4885        L      GRBFRA(,@BR),@XR          SAVE CURR BFR STARTING CADDR
      1643 5C 04 A0 A6           4886        MVC    GRBFRA(GRAEDS,@BR),GRANCA(,@BR)  MOVE NEXT DPL TO CURR DPI
      1647 74 02 A6              4887        ST     GRANCA(,@BR),@XR          RESTORE NEXT BFR STARTING CADDR
      164A 75 02 A0              4888        L      GRBFRA(,@BR),@XR          POINT EN TO CURR BFR CADDR
      164D BD 00 00              4889        CLI    GRAELK(,@XR),GRAELN       NEXT LOGICAL DB = NEXT PHYS DB ?
      1650 F2 81 07              4890        JE     GRA620                    YES, GO INCR SCTR DISP.
      1653 7C 02 A3              4891        MVI    GRANDA(,@BR),GRAEDB       SET DADDR OF NEXT DB
      1656 6E 00 A3 00           4892        ALC    GRANDA(1,@BR),GRAELK(,@XR)  *
      165A 5E 00 A3 AB           4893 GRA620 ALC    GRANDA(1,@BR),GRANPB(,@BR)  INCR SCTR DISP FOR NEXT PHYS D
      165E C0 87 1690            4894 GRA640 B      DL4ICS                    GO READ NEXT DB
      1662 1671             1663 4895        DC     AL2(GRANPL)               * CADDR OF DPL
      1664 7C FF AC              4896 GRA660 MVI    GRASIZ(,@BR),GRAEBS       RE-INITLZ BFR SPACE COUNT
      1667 C0 87 0000            4897 GRA680 B      *-*                       RETURN TO
                            166A 4898 GRA5SA EQU    GRA680+@OP1               * CADDR SUPPLIED
                            166B 4899 GRACPL EQU    *                         DPL FOR CURRENT BUFFER
      166B 02               166B 4900 GRACFN DC     AL1(@DPUT)                WRITE FUNCTION CODE
      166C                  166D 4901 GRSRDA DS     CL2                       RELATIVE DADDR OF CURR. BFR
                            166C 4902 GRACCA EQU    GRSRDA-@B1                CYLINDER BYTE OF DISK ADDR.
      166C                       4903        ORG    *-2                       * INITIALIZED TO THE
      166C 0503             166D 4904        DC     AL2(@WSTBL)               * 1ST DB OF THE WORK FILE
      166E                  166E 4905 GRACSC DS     CL1                       SECTOR COUNT
      166F 1E00             1670 4906 GRBFRA DC     AL2(GRBFR1)               CADDR OF CURRENT BUFFER
                            1671 4907 GRANPL EQU    *                         DPL FOR NEXT BUFFER
      1671 01               1671 4908        DC     AL1(@DGET)                READ FUNCTION CODE
      1672                  1673 4909 GRANDA DS     CL2                       RELATIVE DADDR OF NEXT BFR.
      1674                  1674 4910 GRSCTR DS     CL1                       SECTOR COUNT
      1674                       4911        ORG    *-1                       * INITIALIZE TO 1
      1674 01               1674 4912        DC     XL1'01'
      1675                  1676 4913 GRANCA DS     CL2                       CADDR OF NEXT BUFFER
      1677                  1677 4914 GRWHAT DS     CL1                       USER SPEC'D FUNCTION CODE
      1677                       4915        ORG    *-1                       SET TO ZERO FOR
      1677 00               1677 4916        DC     XL1'00'                   * INITIALIZATION CALL
      1678 0100             1679 4917 GRASSZ DC     XL2'0100'                 SECTOR SIZE
      167A 0001             167B 4918 GRANPB DC     XL2'01'                   DISP TO NEXT PHYS BFR DADDR
                            0002 4919 GRAEDB EQU    2                         DB DADDR ADJUSTMENT FACTOR
      167C                  167C 4920 GRASIZ DS     CL1                       BUFFER SPACE COUNTER
      167D 1B98             167E 4921 GRATXT DC     AL2(GRTEXT)               ADDRESS OF TEXT OUTPUT AREA
      167F 0007             1680 4922 GRAPSG DC     XL2'07'                   SIZE OF PRIMARY SEG. HEADER
      1681 0004             1682 4923 GRASSG DC     XL2'04'                   SIZE OF 2NDARY SEG. HEADER
                            167B 4924 GRAONE EQU    GRANPB                    DECR FACTOR FOR REPITITION CTR
                            167B 4925 GRABOA EQU    GRANPB                    INCR FACTOR FOR NEXT TEXT CHAR
                            167B 4926 GRANXC EQU    GRANPB                    CYL ADJ FACTOR
```

```
     1683                    1683 4927 GRASEG DS   CL1                        SEGMENT TEXT COUNTER
                             0000 4928 GRAEFI EQU  X'00'                      INITIALIZATION FUNC. CODE
                             0003 4929 GRAEFW EQU  X'03'                      WRITE BACK ONLY FUNC. CODE
                             0001 4930 GRAEFR EQU  X'01'                      RETURN TEXT FUNC. CODE
                             0002 4931 GRAEFS EQU  X'02'                      SKIP STATEMENT FUNC. CODE
                             0004 4932 GRAEFG EQU  X'04'                      SKIP SEGMENT FUNC. CODE
                             00FF 4933 GRAEBS EQU  X'FF'                      BUFFER TEXT AREA SIZE
                             0001 4934 GRAESC EQU  X'01'                      SCTR COUNT IF DL4ICS USED
                             0000 4935 GRAELK EQU  X'00'                      DISP TO LINK CODE WITHIN DB
                             0000 4936 GRAELN EQU  X'00'                      LINK CODE TO NEXT PHYS DB
                             0001 4937 GRAEXA EQU  X'01'                      ADJ TO '@' EQU'S FOR @XR ADDRG
                             0006 4938 GRAEDL EQU  @SBLN+GRAEXA               DISP TO STMT BINARY LINE NO.
                             0007 4939 GRAEDT EQU  @STYPE+GRAEXA              DISP TO STMNT TYPE CODE
                             0002 4940 GRAELL EQU  X'02'                      LENGTH OF BINARY LINE NUMBER
                             0075 4941 GRAEET EQU  @EOFTC                     TYPE CODE OF END-OF-FILE STMT
                             0001 4942 GRAES0 EQU  @SDF0+GRAEXA               DISP TO SDF0 - NULL INDR
                             0002 4943 GRAES1 EQU  @SDF1+GRAEXA               DISP TO SDF1 - LENGTH
                             0003 4944 GRAES2 EQU  @SDF2+GRAEXA               DISP TO SDF2 - SEGMENTATION CDE
                             0002 4945 GRAETP EQU  X'02'                      MASK FOR A PRIMARY SEGMENT
                             0007 4946 GRAELP EQU  X'07'                      LENGTH OF PRIMARY SEG.
                             0004 4947 GRAELS EQU  X'04'                      LENGTH OF SECONDARY SEG.
                             001B 4948 GRAEMR EQU  27                         MAX. REPITITION CODE
                             0001 4949 GRAENC EQU  X'01'                      DISP TO NEXT TEXT CHARACTER
                             0001 4950 GRAEDC EQU  X'01'                      DISP TO CYL IN DADDR
                             15D0 4951 GRABSE EQU  GRA310                     BASE ADDRESS OF GRABIT
                             0005 4952 GRAEDS EQU  X'05'                      LNG OF DPL DADDR, SCTR-CT.
                             0006 4953 GRAEW2 EQU  6                          SECOND CYL OF WORK FILE
                                  4954 *
                                  4955 *          ERROR ROUTINE
                                  4956 *
     1684 3C 98 03CD              4957 GRAERR MVI  $CAERR,@@E551              SET BAD FILE ERROR CODE
                                  4958 *          THE ABOVE ERROR CODE IS INITIALLY SET FOR A SAVED FILE,
                                  4959 *          BUT IS MODIFIED TO THE WORK FILE IF DL4ICS IS USED
     1688 3A 04 03D6              4960        SBN  $INDR3,$ERHRD              SET INDR FOR HARD ERROR
     168C C0 87 0469              4961        B    $CAERK                     GO TO ERRPGM INTERFACE
                                  4962 *
                                  4963 *     $DL4P
```

```
      DL4ICS - FOUR TRACK LOGICAL IOCR
```

```
                        4965+*******************************************************************************
                        4966+* 5703-XM1  COPYRIGHT IBM CORP. 1970                            *
                        4967+*            REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083 *
                        4968+*                                                               *
                        4969+*******************************************************************************
                        4970+*STATUS                                                         *
                        4971+*   VERSION 1 MODIFICATION 0                                    *
                        4972+*                                                               *
                        4973+*FUNCTION                                                       *
                        4974+*   * DL4ICS WILL CONVERT A RELATIVE DISK ADDRESS TO A PHYSICAL  *
                        4975+*     DISK ADDRESS AND CALL $DISKN TO PERFORM THE SPECIFIED FUNCTION *
                        4976+*   * THE DISK ADDRESS IS A ONE BYTE CYLINDER ADDRESS AND A ONE BYTE *
                        4977+*     SECTOR DISPLACEMENT RELATIVE TO SECTOR 0 ON A CYLINDER     *
                        4978+*     BOUNDARY                                                   *
                        4979+*   * WHEN MORE THAN 1 SECTOR IS PROCESSED, DL4ICS WILL MAKE MULTIPLE *
                        4980+*     CALLS TO $DISKN TO CROSS CYLINDER BOUNDARIES IF REQUIRED.  *
                        4981+*   * IF 1 SECTOR ONLY IS TO BE PROCESSED, THE USER MAY OVERLAY THE *
                        4982+*     UNUSED CODE BY ORGING HIS NEXT MODULE AT DL4SPT            *
                        4983+*                                                               *
                        4984+*ENTRY POINTS                                                   *
                        4985+*     DL4ICS - ENTRY TO PROCESS A 4 SURFACE FILE. THE CALLING    *
                        4986+*              SEQUENCE IS AS FOLLOWS                            *
                        4987+*                 DSKL4  DPL                                     *
                        4988+*              WHERE DPL IS THE LABEL OF A SIX BYTE DISK PARAMETER *
                        4989+*              LIST AS DESCRIBED FOR $DISKN EXCEPT FOR THE SECTOR *
                        4990+*              ADDRESS BYTE.                                     *
                        4991+*                                                               *
                        4992+*INPUT                                                          *
                        4993+*   * INPUT TO DL4ICS IS THE ADDRESS OF THE DPL TO BE PROCESSED. *
                        4994+*                                                               *
                        4995+*OUTPUT                                                         *
                        4996+*   * N/A                                                        *
                        4997+*                                                               *
                        4998+*EXTERNAL REFENECES                                             *
                        4999+*     $DISKN - ENTRY TO SYSTEM DISK ROUTINE                      *
                        5000+*                                                               *
                        5001+*EXITS, NORMAL                                                  *
                        5002+*   * NORMAL RETURN IS TO THE 1ST INSTRUCTION FOLLOWING THE TWO BYTE *
                        5003+*     ADDRESS POINTING TO THE DPL.                              *
                        5004+*                                                               *
                        5005+*EXITS, ERROR                                                   *
                        5006+*   * N/A                                                        *
                        5007+*                                                               *
                        5008+*TABLES/WORK AREAS                                              *
                        5009+*   * N/A                                                        *
                        5010+*                                                               *
                        5011+*ATTRIBUTES                                                     *
                        5012+*   * RELOCATABLE                                                *
                        5013+*   * REUSABLE                                                   *
                        5014+*                                                               *
                        5015+*CHARACTER CODE DEPENDENCY                                       *
                        5016+*   * THE OPERATION OF THIS MODULE DOES NOT DEPEND UPON A PARTICULAR *
                        5017+*     INTERNAL REPRESENTATION OF THE EXTERNAL CHARACTER SET.    *
                        5018+*                                                               *
                        5019+*NOTES                                                          *
                        5020+*   ERROR PROCEDURES                                            *
```

```
                              5021+*     N/A                                                    *
                              5022+*                                                            *
                              5023+*   REGISTER USAGE                                           *
                              5024+*     @BR IS SAVED AND RESTORED ON EXIT, @XR IS NOT USED.  @ARR IS  *
                              5025+*     USED TO PROVIDE THE ADDRESS OF THE PARAMETER.  THE @ARR IS    *
                              5026+*     INCREMENTED BT TWO AND SAVED AS THE RETURN ADDRESS.    *
                              5027+*                                                            *
                              5028+*   SAVED/RESTORED AREAS                                     *
                              5029+*     N/A                                                    *
                              5030+*                                                            *
                              5031+*   MODIFICATION CONSIDERATIONS                              *
                              5032+*     N/A                                                    *
                              5033+*                                                            *
                              5034+*   REQUIRED MODULES                                         *
                              5035+*     @SYSEQ - SYSTEM SOFTWARE EQUATES                        *
                              5036+*     @FXDEQ - SYSTEM NUCLEUS EQUATES                         *
                              5037+*                                                            *
                              5038+*   OTHER                                                    *
                              5039+*     NONE                                                   *
                              5040+***********************************************************************
```

```
     DL4ICS - FOUR TRACK LOGICAL IOCR
```

ERR LOC   OBJECT CODE   ADDR STMT SOURCE STATEMENT   VER 15, MOD 00 04/07/22 PAGE 57

```
                          1690 5042+DL4ICS EQU    *                        ENTRY TO DL4ICS
                          1694 5043+       USING  DL4010,@BR               ESTABLISH BASE REGISTER USAGE
     1690 34 01 1700           5044+       ST     DL4900+@OP1,@BR          SAVE BASE REGISTER FOR EXIT
                          1694 5045+DL4010 EQU    *                        BASE ADDRESSABILITY
     1694 C2 01 1694           5046+       LA     DL4010,@BR               ESTABLISH BASE
     1698 76 08 78             5047+       A      DL4C01(,@BR),@ARR        BUMP TO HIGH END OF ADDR
     169B 74 08 14             5048+       ST     DL4020+@DOP2(,@BR),@ARR  SET UP MOVE INSTRUCTION
     169E 76 08 78             5049+       A      DL4C01(,@BR),@ARR        BUMP TO RETURN ADDR
     16A1 74 08 70             5050+       ST     DL4920+@OP1(,@BR),@ARR   SAVE RETURN ADDR
                               5051+*
     16A4 4C 01 1D 0000        5052+DL4020 MVC    DL4030+@DOP2(@DADDR,@BR),*-* MOVE DPL ADDR INTO MOVE
     16A9 5E 01 1D 7A          5053+       ALC    DL4030+@DOP2(@CADDR,@BR),DL4C05(,@BR) BUMP TO RIGHT END
     16AD 4C 05 76 0000        5054+DL4030 MVC    DL4DPL(@DPLNG,@BR),*-*   MOVE USER DPL TO WORK AREA
                               5055+*
     16B2 7C 00 5E             5056+DL4035 MVI    DL4100+@Q(,@BR),@ZERO    CLEAR TRACK, DISK SET INST
     16B5 7C 80 67             5057+       MVI    DL4200+@Q(,@BR),@NOP     TURN OFF TWICE INDICATOR
                               5058+*
     16B8 7D 60 73             5059+DL4040 CLI    DL4SCD(,@BR),DL4E96      TEST IF DISPLACEMENT OVER 95 ?
     16BB F2 82 0B             5060+       JL     DL4050                   JUMP IF NOT OVER 95
     16BE 5E 00 72 78          5061+       ALC    DL4CYL(1,@BR),DL4C01(,@BR) INCREMENT CYLINDER COUNT
     16C2 5F 00 73 25          5062+       SLC    DL4SCD(1,@BR),DL4C96(,@BR) DECREMENT DISP BY 96
     16C6 D0 87 24             5063+       B      DL4040(,@BR)             GO BACK CHECK FOR NEXT CYLINDER
                               5064+*
     16C9 7D 30 73             5065+DL4050 CLI    DL4SCD(,@BR),DL4E48      TEST IF DISP ON NEXT DISK ?
     16CC F2 82 07             5066+       JL     DL4060                   JUMP IF NOT OVER 48
     16CF 7A 01 5E             5067+       SBN    DL4100+@Q(,@BR),DL4EFD   TURN ON BIT FOR FIXED DISK
     16D2 5F 00 73 36          5068+       SLC    DL4SCD(1,@BR),DL4C48(,@BR) DECREMENT DISP 1 DISK
     16D6 7D 01 74             5069+DL4060 CLI    DL4SCT(,@BR),DL4E01      IS SECTOR COUNT GREATER THEN 1 ?
     16D9 F2 84 33             5070+       JH     DL4SPT                   GO TO SPLIT CALL
     16DC 7D 18 73             5071+DL4070 CLI    DL4SCD(,@BR),DL4E24      DISPLACEMENT OVER 23 ?
     16DF F2 82 07             5072+       JL     DL4080                   JUMP NOT OVER 24
     16E2 7A 80 5E             5073+       SBN    DL4100+@Q(,@BR),DL4ETB   SET TRACK BIT ON
     16E5 5F 00 73 49          5074+       SLC    DL4SCD(1,@BR),DL4C24(,@BR) DECR DISP TO NEXT TRACK
     16E9 5E 00 73 73          5075+DL4080 ALC    DL4SCD(1,@BR),DL4SCD(,@BR) SHIFT LEFT 1 PLACE
     16ED 5E 00 73 73          5076+       ALC    DL4SCD(1,@BR),DL4SCD(,@BR) SHIFT LEFT 1 PLACE
     16F1 7A 00 73             5077+DL4100 SBN    DL4SCD(,@BR),*-*         SET TRACK, DISK BIT
                               5078+*
     16F4 C0 87 0025           5079+       B      $DISKN                   GO PERFORM DISK I/O
     16F8 1705            16F9 5080+       DC     AL2(DL4LST)              ADDR OF DISK PARAM LIST
                               5081+*
     16FA F2 00 3C             5082+DL4200 JC     DL4600,*-*               BRANCH OR NOP IF TWICE SET
                               5083+*
     16FD C2 01 0000           5084+DL4900 LA     *-*,@BR                  RESTORE OLD BASE TO RETURN
     1701 C0 87 0000           5085+DL4920 B      *-*                      RETURN TO CALLER

                          1705 5087+DL4LST EQU    *                        LEFT END OF DPL
     1705                 170A 5088+DL4DPL DS     CL(@DPLNG)               DPL SAVE AREA
                          1706 5089+DL4CYL EQU    DL4LST+@DCYL             CYLINDER COUNT BYTE
                          1707 5090+DL4SCD EQU    DL4LST+@DSAD             DISPLACEMENT SECTOR COUNT
                          0060 5091+DL4E96 EQU    96                       TWO DISK SECTOR COUNT PER CYL
                          0030 5092+DL4E48 EQU    48                       ONE DISK SECTOR COUNT PER CYL
                          0018 5093+DL4E24 EQU    24                       TRACK SECTOR COUNT
                          0001 5094+DL4E01 EQU    01                       VALUE TO TEST SECTOR COUNT
                          0001 5095+DL4EFD EQU    01                       VALUE TO SET FIXED DISK BIT
                          0080 5096+DL4ETB EQU    X'80'                    VALUE TO SET TRACK BIT
     170B 0001            170C 5097+DL4C01 DC     IL2'1'                   VALUE TO INCR TO CYLINDER
```

```
          DL4ICS - FOUR TRACK LOGICAL IOCR


ERR LOC   OBJECT CODE       ADDR STMT SOURCE STATEMENT            VER 15, MOD 00  04/07/22  PAGE  58

    170D 0005              170E 5098+DL4C05 DC    IL2'5'                       DISP TO RIGHT END OF DPL
                           16B9 5099+DL4C96 EQU   DL4040+@Q                    VALUE TO DECR DISPLACEMENT
                           16DD 5100+DL4C24 EQU   DL4070+@Q                    VALUE OF 1 TRACK
                           1708 5101+DL4SCT EQU   DL4LST+@DCNT                 POINTER TO DPL SECTOR COUNT
                           16CA 5102+DL4C48 EQU   DL4050+@Q                    VALUE TO DECR DISP BY 1 DISK

    170F 5C 00 14 74            5104+DL4500 MVC   DL4WRK(1,@BR),DL4SCT(,@BR) PICKUP SECTOR COUNT
                           170F 5105+DL4SPT EQU   DL4500                       POSSIBLE OVERLAY REFERENCE
    1713 5E 00 14 73            5106+       ALC   DL4WRK(1,@BR),DL4SCD(,@BR) BUMP BY DISPLACEMENT
    1717 7D 30 14              5107+       CLI   DL4WRK(,@BR),DL4E48         TEST FOR CYLINDER OVERLAP
    171A D0 04 48              5108+       BNH   DL4070(,@BR)               BRANCH BACK IF NO OVERLAY
    171D 5F 00 14 36           5109+       SLC   DL4WRK(1,@BR),DL4C48(,@BR) DECREMENT WORK BY 48
    1721 5F 00 74 14           5110+       SLC   DL4SCT(1,@BR),DL4WRK(,@BR) SUBTRACT WORK FROM COUNT
    1725 7C 87 67              5111+       MVI   DL4200+@Q(,@BR),@UCB       SET TWICE SWITCH
    1728 5C 00 13 73           5112+       MVC   DL4SAV(1,@BR),DL4SCD(,@BR) SAVE SECTOR DISP IN WORK AREA
    172C 78 01 5E              5113+       TBN   DL4100+@Q(,@BR),DL4EFD     DISK BIT ON IN Q CODE ?
    172F D0 90 48              5114+       BF    DL4070(,@BR)               BRANCH NOT ON
    1732 5E 00 13 36           5115+       ALC   DL4SAV(1,@BR),DL4C48(,@BR) BUMP TO NEXT DISK
    1736 D0 87 48              5116+       B     DL4070(,@BR)               RETURN TO CALL I/O
                               5117+*
    1739 5C 00 73 13           5118+DL4600 MVC   DL4SCD(1,@BR),DL4SAV(,@BR) PICKUP NEXT HALF OF I/O
    173D 5E 00 75 74           5119+       ALC   DL4LST+@DBFR1(1,@BR),DL4SCT(,@BR) BUMP CORE ADDRESS
    1741 5E 00 73 74           5120+       ALC   DL4SCD(1,@BR),DL4SCT(,@BR)
    1745 5C 00 74 14           5121+       MVC   DL4SCT(1,@BR),DL4WRK(,@BR) MOVE IN NEW SECTOR COUNT
    1749 D0 87 1E              5122+       B     DL4035(,@BR)               RETURN FOR SECOND PASS
                               5123+*
                           16A8 5124+DL4WRK EQU   DL4020+@DOP2                 1 BYTE WORK AREA FOR SPLIT CALL
                           16A7 5125+DL4SAV EQU   DL4020+@DOP2-1               1 BYTE WORK AREA FOR SPLIT CALL
                           174C 5126+DL4END EQU   *                            DEFINE END OF CODE
                               5127+***                          END OF DL4ICS                          ***
                               5128 *
                               5129 *       $DLPR
```

```
                          5131+******************************************************************************
                          5132+* 5703-XM1      COPYRIGHT IBM CORP. 1970                             *
                          5133+*                   REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083  *
                          5134+*                                                                    *
                          5135+******************************************************************************
                          5136+*STATUS                                                              *
                          5137+*    VERSION 1 MODIFICATION 0                                        *
                          5138+*                                                                    *
                          5139+*FUNCTION                                                            *
                          5140+*    * DLPRNT PROVIDES FOR DEVICE INDEPENDENCE FOR OUTPUT FROM        *
                          5141+*      LIST ORIENTED PROGRAMS.                                       *
                          5142+*    * FOR CRT OUTPUT, ROLL SPEED AND POP FEATURES ARE SUPPORTED.     *
                          5143+*      IN ADDITION DLPRNT WILL FLASH COMMAND LIGHT 13 WHEN IN         *
                          5144+*      STOP MODE.                                                    *
                          5145+*    * IF A 50LMP MATRIX PRINTER IS TO BE USED, ALL PRINTED LINES     *
                          5146+*      ARE ANALYZED FOR LENGTH TO PROVIDE MAXIMUM LINE THROUGHPUT.    *
                          5147+*      THIS IS DONE BY PRINTING RIGHT ONLY AS FAR AS REQUIRED TO      *
                          5148+*      PRINT THE NEXT LINE FROM RIGHT TO LEFT.  THE 50LMP I/O         *
                          5149+*      INTERFACE IS SUPPLIED BY DLPRNT.                               *
                          5150+*    * OUTPUT MAY BE DIRECTED TO THE CRT, THE MATRIX PRINTER, OR       *
                          5151+*      THE CURRENT SYSTEM OUTPUT DEVICE(S).                           *
                          5152+*                                                                    *
                          5153+*ENTRY POINTS                                                         *
                          5154+*    DLPRNT HAS ONE ENTRY POINT.  THIS ENTRY POINT IS USED WHEN A      *
                          5155+*    LINE IS TO BE PRINTED FOLLOWED BY A NORMAL CARRIER RETURN.        *
                          5156+*    THE CALLING SEQUENCE IS:                                         *
                          5157+*                                                                    *
                          5158+*         B     DLPRNT                                                *
                          5159+*         DC    AL2(PPLA)                                             *
                          5160+*    WHERE PPLA IS A TWO BYTE ADDRESS OF THE LEFT BYTE OF A PRINT      *
                          5161+*    PARAMETER LIST.                                                  *
                          5162+*                                                                    *
                          5163+*INPUT                                                                *
                          5164+*    * BEFORE USING DLPRNT THE ONE BYTE INDICATOR, DLPTYP, MUST        *
                          5165+*      BE SET TO INDICATE WHICH DEVICE IS TO BE USED FOR OUTPUT.       *
                          5166+*      THE CORRESPONDING VALUES AND THEIR FUNCTION FOLLOWS:           *
                          5167+*         DLPMPR - MATRIX PRINTER IS TO BE USED FOR OUTPUT.           *
                          5168+*         DLPCRT - THE DISPLAY STATION IS TO BE USED FOR OUTPUT.       *
                          5169+*                  ROLL SPEED AND POP FUNCTIONS WILL BE CONTROLED.    *
                          5170+*         DLPSPT - THE SYSTEM PRINTER(S) IS TO OE USED FOR OUTPUT.     *
                          5171+*                  THIS IS THE DEFAULT VALUE.                         *
                          5172+*    * A 244 BYTE BUFFER MUST BE ALLOCATED FOR DLPRNTS USE STARTING    *
                          5173+*      AT LOCATION DLIBUF.                                            *
                          5174+*    * A FOUR BYTE PRINT PARAMETER LIST (PPL) MUST BE PASSED VIA       *
                          5175+*      A TWO BYTE COME ADDRESS FOLLOWING THE CALL.  THIS PPL IS OF     *
                          5176+*      THE SAME FORMAT AS THE PPL SENT TO DPRINT WITH THE FOLLOWING    *
                          5177+*      RESTRICTIONS:                                                 *
                          5178+*         * ONLY 'PRINT AND RETURN' CONTROL CODES ARE ALLOWED FOR      *
                          5179+*           PRINTING.                                                *
                          5180+*         * WAIT FUNCTIONS SHOULD NOT BE USED EXCEPT AFTER THE LAST     *
                          5181+*           LINE HAS BEEN PRINTED.  IT IS THEN REQUIRED TO TERMINATE   *
                          5182+*           DLPRNT'S FUNCTION.                                        *
                          5183+*OUTPUT                                                               *
                          5184+*    UPON COMPLETION THE GENERAL REGISTERS AND PPL WILL BE THE SAME    *
                          5185+*    AS AT ENTRY, THE LINE TO BE PRINTED WILL BE PRINTED (OR BUFFERED  *
                          5186+*    IN THE CASE OF THE LINE PRINTER).  THE CALLING PROGRAM MAY        *
```

```
                            5187+*   MODIFY THE LINE UPON RETURN.                              *
                            5188+*                                                             *
                            5189+*EXTERNAL REFERENCES                                          *
                            5190+*        $PRDEV - SYSTEM PRINTER INDICATOR.                   *
                            5191+*        DLIBUF - LOCATION OF BUFFER.                         *
                            5192+*        $$PLYN - ENTRY TO DSPLYN.                            *
                            5193+*        $$PRNT - ENTRY TO DPRINT.                            *
                            5194+*        $CRTIN - ROLL INDICATORS.                            *
                            5195+*        $IOIND - LINE PRINTER INDICATOR.                     *
                            5196+*        $UNMSK - ENTRY TO UNMASK INQUIRY REQUEST.            *
                            5197+*        $$PSIO - LOCATION OF CONTROL BYTE IN DPRINT SIG.     *
                            5198+*        $$PCNT - LOCATION OF COUNT BYTE IN DPRINT I/O LIST.  *
                            5199+*                                                             *
                            5200+*EXITS, NORMAL                                                *
                            5201+*   EXIT IS TO THE CALLING PROGRAM FOLLOWING THE PPL ADDRESS. *
                            5202+*                                                             *
                            5203+*EXITS, ERROR                                                 *
                            5204+*   N/A                                                       *
                            5205+*                                                             *
                            5206+*TABLES/WORK AREAS                                            *
                            5207+*   N/A                                                       *
                            5208+*                                                             *
                            5209+*ATTRIBUTES                                                   *
                            5210+*   RELOCATABLE                                               *
                            5211+*   REUSABLE                                                  *
                            5212+*                                                             *
                            5213+*CHARACTER CODE DEPENDENCY                                    *
                            5214+*   N/A                                                       *
                            5215+*                                                             *
                            5216+*NOTES                                                        *
                            5217+*   ERROR PROCEDURES                                          *
                            5218+*     N/A                                                     *
                            5219+*                                                             *
                            5220+*   REGISTER USAGE                                            *
                            5221+*     REGISTERS 1 AND 2 ARE USED FOR BASE ADDRESSING.         *
                            5222+*                                                             *
                            5223+*   SAVED/RESTORED AREAS                                       *
                            5224+*     N/A                                                     *
                            5225+*                                                             *
                            5226+*   MODIFICATION CONSIDERATIONS                               *
                            5227+*     DLPRNT DIRECTLY MODIFIES DPRINT WHEN USING THE LINE PRINTER *
                            5228+*     FUNCTION.  CARE MUST BE TAKEN WHEN MODIFING EITHER DLPRNT OR *
                            5229+*     DPRINT.                                                  *
                            5230+*                                                             *
                            5231+*   REQUIRED MODULES                                          *
                            5232+*        @SYSEQ - GENERAL SYSTEM EQUATES                       *
                            5233+*        @FXDEQ - NUCLEUS LOCATION EQUATES                     *
                            5234+*        @HDWEQ - HARDWARE VALUE EQUATES                       *
                            5235+*        @CANEQ - TRANSCIENT LOCATION EQUATES                 *
                            5236+*                                                             *
                            5237+*   OTHER                                                     *
                            5238+*     N/A                                                     *
                            5239+*****************************************************************
```

```
                              1785 5241+        USING DLPBSE,@BR               BASE SPECIFICATION
                              174C 5242+DLPRNT EQU   *                         ENTRY
     174C 34 01 1856              5243+        ST    DLP480+@OP1,@BR           SAVE BR
     1750 C2 01 1785              5244+        LA    DLPBSE,@BR                LOAD BASE REG
     1754 74 02 D5                5245+        ST    DLP500+@OP1(,@BR),@XR     SAVE XR
     1757 76 08 ED                5246+        A     DLPONE(,@BR),@ARR         CALCULATE PPL ADDR POINTER
     175A 34 08 1767              5247+        ST    DLP100+@OP1,@ARR          GET PARM ADDR
     175E 76 08 ED                5248+        A     DLPONE(,@BR),@ARR         CALCULATE RETURN ADDR
     1761 74 08 DD                5249+        ST    DLP520+@OP1(,@BR),@ARR    SAVE RETURN ADDR
     1764 35 02 0000              5250+DLP100 L     *-*,@XR                   XR POINTS TO PPL
     1768 6C 03 EA 03             5251+        MVC   DLPWK2+@PDATA(@PPLNG,@BR),@PDATA(,@XR) MOVE IN PPL
     176C 7C 20 0F                5252+        MVI   DLPEXT-1(,@BR),X'20'      INITIALIZE DSPLYN ADDR    *****
     176F 4E 00 0F 043B           5253+        ALC   DLPEXT-1(1,@BR),$EXFTR    GET DSPLYN ADDR
     1774 F2 87 00                5254+        J     *-*                       GO TO CORRECT INTERFACE
                              1776 5255+DLPTYP EQU   *-1                       I/O DEVICE INDR LOCATION
     1776                         5256+        ORG   DLPTYP                    SET INSTR CNTR
     1776 00                  1776 5257+        DC    AL1(DLPSPT)               SET DEFAULT TO SYSTEM PRINTER
                              1777 5258+DLPBSD EQU   *                         DISPLACEMENT BASE
                                   5259+**
                              1777 5260+DLPSPI EQU   *                         SYSTEM PRINTER INTERFACE
     1777 3D 07 044A              5261+        CLI   $PRDEV-1,X'07'            SYSPRINT = MATRIX PRINT   *****
     177B F2 81 7E                5262+        JE    DLPNPT                    DO LIME PRINTER INTERFACE
     177E 5C 01 00 10             5263+        MVC   DLP120+@OP1(@CADDR,@BR),DLPEXT(,@BR) GET DSPLYN ADDR
     1782 C0 87 0000             5264+DLP120 B     *-*                       GO TO DSPLYN
     1786 186C                1787 5265+        DC    AL2(DLPWK2)               PPL ADDRESS
     1788 3D 00 044B             5266+        CLI   $PRDEV,X'00'              IS PRINTER REQUIRED TOO   *****
     178C F2 81 6D                5267+        JE    DLPNPT                    DO LINE PRINTER INTERFACE
     178F F2 87 C1                5268+        J     DLP480                    EXIT INTERFACE
                              1785 5269+DLPBSE EQU   DLP120+@OP1               BASE ADDRESS
```

```
                            1792 5271+DLPTIF EQU    *                            ENTRY
     1792 C0 87 0000             5272+       B      *-*                          GO TO DSPLYN
     1794                        5273+       ORG    *-2                          INITIALIZE ADDR
     1794 2004             1795 5274+DLPEXT DC     AL2($$PLYN)                   DSPLYN ENTRY ADDR
     1796 186C             1797 5275+       DC     AL2(DLPWK2)                   PPL ADDRESS
     1798 7D FF E7              5276+       CLI    DLPWK2+@PCTRL(,@BR),@PWAIT WAIT FUNCTION ?
     179B F2 81 57              5277+       JE     DLP360                        GO TURN OFF CMD LIGHTS
     179E 71 11 E2              5278+DLP140 LIO    DLPK13(,@BR),@KEYBD+@CMLON TURN ON CMD LITE 13
     17A1 38 08 03D3            5279+       TBN    $CRTIN,$CRTSP                 IN STOP MODE?
     17A5 F2 90 1D              5280+       JF     DLP240                        NO ? CONTINUE ROLL
     17A8 F2 80 09              5281+DLP160 JC     DLP180,@NOP                   JUMP IF LIGHT ON
     17AB 71 10 E2              5282+       LIO    DLPK13(,@BR),@KEYBD+@CMOFF TURN POP LITE OFF
     17AE 7C 87 24              5283+       MVI    DLP160+@Q(,@BR),@UCB          SET FOR TURN ON
     17B1 F2 87 03              5284+       J      DLP200                        GO DO TIME OUT
     17B4 7C 80 24              5285+DLP180 MVI    DLP160+@Q(,@BR),@NOP          SET TO TURN OFF
     17B7 5C 01 E0 E1           5286+DLP200 MVC    DLPLPC(2,@BR),DLPLIN(,@BR) SET UP TIME COUNT
     17BB 5F 01 E0 ED           5287+DLP220 SLC    DLPLPC(2,@BR),DLPONE(,@BR) DECREMENT TIME COUNT
     17BF D0 84 36              5288+       BH     DLP220(,@BR)                  LOOP UNTIL TIME OUT
     17C2 D0 87 19              5289+       B      DLP140(,@BR)                  GO TEST STOP MODE
     17C5 38 04 03D3            5290+DLP240 TBN    $CRTIN,$CRTPU                 POP UP INDR ON ?
     17C9 F2 90 07              5291+       JF     DLP260                        SKIP LINE CNT INITIALIZATION
     17CC 3B 04 03D3            5292+       SBF    $CRTIN,$CRTPU                 SET POP UP INDR
     17D0 7C 00 DE              5293+       MVI    DLPCNT(,@BR),@ZERO            ZERO LINES DISPLAYED CNT
     17D3 7D 0D DE              5294+DLP260 CLI    DLPCNT(,@BR),DLPMAX           HAVE MAX NO. OF LINES BEEN ?
                                5295+*                                          * DISPLAYED ?
     17D6 F2 01 04              5296+       JNE    DLP280                        JUMP IF NOT
     17D9 3A 08 03D3            5297+       SBN    $CRTIN,$CRTSP                 SET ROLL STOP INDR
     17DD F2 04 0E              5298+DLP280 JNH    DLP320                        JUMP IF MAX LINES NOT DISPLAYED
     17E0 5C 01 E0 E1           5299+       MVC    DLPLPC(2,@BR),DLPLIN(,@BR) SET UP TIMING LOOP
     17E4 5F 01 E0 ED           5300+DLP300 SLC    DLPLPC(2,@BR),DLPONE(,@BR) DECREMENT COUNT
     17E8 D0 84 5F              5301+       BH     DLP300(,@BR)                  BRANCH IF TIME NOT UP
     17EB F2 87 04              5302+       J      DLP340                        GO EXIT
     17EE 5E 00 DE ED           5303+DLP320 ALC    DLPCNT(1,@BR),DLPONE(,@BR) BUMP LINE COUNT
     17F2 F2 87 5E              5304+DLP340 J      DLP480                        GO EXIT
     17F5 C0 87 0B44            5305+DLP360 B      $$COFF                        TURN OFF CMD LIGHTS
     17F9 F2 87 57              5306+       J      DLP480                        GO EXIT
```

```
                            17FC 5308+DLPNPT EQU   *                           ENTRY
    17FC 38 80 03D2              5309+        TBN   $IOIND,$LNPTR              LINE PRINTER AVAILABLE
    1800 F2 10 0F                5310+        JT    DLP400                     JUMP IF YES
    1803 C0 87 0707              5311+DLP380 B     $$PRNT                     DO NORMAL PRINT IF NOT
    1807 186C               1808 5312+        DC    AL2(DLPWK2)                PPL ADDR
    1809 C0 87 0707              5313+        B     $$PRNT                     WAIT FOR OP COMPLETION
    180D 057F               180E 5314+        DC    AL2($WAITF)                WAIT PPL ADDRESS
    180F F2 87 41                5315+        J     DLP480                     GO EXIT
    1812 7D FF E7                5316+DLP400 CLI   DLPWK2+@PCTRL(,@BR),@PWAIT IS THIS A WAIT FUNCTION ?
    1815 F2 01 03                5317+        JNE   DLP420                     JUMP IF NO
    1818 7C 00 E8                5318+        MVI   DLPWK2+@PRCNT(,@BR),@ZERO ZERO NEXT LINE CNT
    181B 7D FF E3                5319+DLP420 CLI   DLPWK1(,@BR),@PWAIT        IS THERE A LINE TO PRINT ?
    181E F2 01 59                5320+        JNE   DLPPRT                     JUMP IF YES
    1821 C0 87 0707              5321+        B     $$PRNT                     INSURE PRINT HEAD IS AT LEFT
    1825 1878               1826 5322+        DC    AL2(DLPRTN)                * MARGIN
    1827 5C 01 E4 E8             5323+DLP440 MVC   DLPWK1+@PRCNT(2,@BR),DLPWK2+@PRCNT(,@BR) SET NEXT PPL
    182B 5C 01 E8 F4             5324+        MVC   DLPWK2+@PRCNT(2,@BR),DLPRTN+@PRCNT(,@BR) SET CARRIER RTN
    182F 7D FF E3                5325+        CLI   DLPWK1(,@BR),@PWAIT        WAS THIS A WAIT FUNCTION ?
    1832 D0 81 7E                5326+        BE    DLP380(,@BR)               DO CARRIER RETURN IF YES
    1835 C2 02 1C8C              5327+        LA    DLIBUF,@XR                 POINT XR TO BUFFER
    1839 BC 40 F3                5328+        MVI   DLPBLN-1(,@XR),@BLANK      SET BLANK FOR CLEAR BUF
    183C AC F2 F2 F3             5329+        MVC   DLPBLN-2(DLPBLN-1,@XR),DLPBLN-1(,@XR) CLEAR BUF TO BLNKS
    1840 5C 00 CD E4             5330+        MVC   DLP460+@DD2(1,@BR),DLPWK1+@PRCNT(,@BR) SET DATA CNT
    1844 5F 00 CD ED             5331+        SLC   DLP460+@DD2(1,@BR),DLPONE(,@BR) GET TRUE DISPLACMENT
    1848 5C 01 CC CD             5332+        MVC   DLP460+@D1(2,@BR),DLP460+@DD2(,@BR) SET 0 AND DI VALUES
    184C 75 01 EA                5333+        L     DLPWK2+@PDATA(,@BR),@BR    BR POINTS TO DATA
    184F 9C 00 00 00             5334+DLP460 MVC   *-*(@VQ,@XR),*-*(,@BR)      MOVE DATA TO BUFFER
                                 5335+*
    1853 C2 01 0000              5336+DLP480 LA    *-*,@BR                    RESTORE BR
    1857 C2 02 0000              5337+DLP500 LA    *-*,@XR                    RESTORE XR
    185B C0 87 048D              5338+        B     $UNMSK                     GO CHECK FOR INQUIRY REQUEST
    185F C0 87 0000              5339+DLP520 B     *-*                        RETURN
```

```
      DLPRNT -- LIST OUTPUT INTERFACE


ERR LOC  OBJECT CODE      ADDR STMT SOURCE STATEMENT          VER 15, MOD 00  04/07/22  PAGE  64

                          5341+********************************************************************
                          5342+* CONSTANTS, WORK AREAS AND EQUATES
                          5343+********************************************************************
                          5344+*
                     0085 5345+DLPMPR EQU   DLPNPT-DLPBSD               MATRIX PRINTER INDR VALUE
                     0000 5346+DLPSPT EQU   DLPSPI-DLPBSD               SYSTEM PRINTER INDR VALUE
                     001B 5347+DLPCRT EQU   DLPTIF-DLPBSD               CRT INOR VALUE
     1863            1863 5348+DCRCNT DS    CL1                         DISPLAYED LINE CNTR
                     1863 5349+DLPCNT EQU   DCRCNT                      COMMUNICATIONS LABEL
     1863                 5350+       ORG   DLPCNT                      SET INST CNTR
     1863 01         1863 5351+       DC    XL1'01'                     INITIAL VALUE
     1864            1865 5352+DLPLPC DS    CL2                         TIMING LOOP CNTR
     1866 3B         1866 5353+DLPLIN DC    XL1'3B'                     INITIAL LOOP CNT
     1867 0D         1867 5354+DLPK13 DC    AL1(@CKY13)                 CMD LIGHT 13 CONTROL
                     000D 5355+DLPMAX EQU   13                          MAX LINES TO BE DISPLAYED
                     1868 5356+DLPWK1 EQU   *                           CURRENT PPL
     1868 FFFF       1869 5357+       DC    2XL1'FF'                    CTRL AND DATA CNT
     186A 1C8C       186B 5358+       DC    AL2(DLIBUF)                 BUFFER ADDR
                     186C 5359+DLPWK2 EQU   *                           NEXT PPL
     186C            186F 5360+       DS    CL(@PPLNG)
     1870 01         1870 5361+DLPNDX DC    AL1(@INDEX)                 INDEX PPL
     1871 0001       1872 5362+DLPONE DC    XL2'0001'                   CONSTANT OF ONE
     1873            1873 5363+DLPRES DS    CL1                         RESIDUAL CNT
     1874 0000       1875 5364+DLPWTH DC    XL2'00'                     WIDTH OF PRINT LINE
     1876            1876 5365+DLPNXT DS    CL1                         NEXT LINE CNT
     1877            1877 5366+DLPREM DS    CL1                         ADDITIONAL CNT FOR NEXT LINE
                     1878 5367+DLPRTN EQU   *                           ADDR OF RETURN PPL
     1878 8080       1879 5368+       DC    2AL1(@RETRN)                RETURN CARRIER PPL
                     0001 5369+DLPPNT EQU   X'01'                       LINE PRINTER CONTROL BYTE
```

```
                              5371+********************************************************************
                              5372+* THIS ROUTINE PRINTS THE CURRENT LINE IN THE CORRECT DIRECTION AND
                              5373+* SETS UP THE NEXT LINE CNT.
                              5374+********************************************************************
                         1868 5375+      USING DLPBS2,@BR                    NEW BASE VALUE
                         187A 5376+DLPPRT EQU   *                            ENTRY TO PRINT
     187A C2 01 1868          5377+      LA    DLPBS2,@BR                    LOAD BASE REGISTER
     187E C0 87 0707          5378+      B     $$PRNT                        WAIT FOR PRINTER READY
     1882 057F           1883 5379+      DC    AL2($WAITF)                   WAIT PPL
     1884 3C 80 0476          5380+      MVI   $CIMSK,@NOP                   MASK IR FOR THIS FUNCTION
     1888 4C 00 0D 03C0       5381+      MVC   DLPWTH(1,@BR),$RMRGN          SET RIGHT MARGIN VALUE
     188D 4F 00 0D 03C1       5382+      SLC   DLPWTH(1,@BR),$LMRGN          CALCULATE WIDTH
     1892 5C 00 0E 05         5383+      MVC   DLPNXT(1,@BR),DLPWK2+@PRCNT(,@BR) SET NEXT LINE CNT
     1896 7C 00 0B            5384+      MVI   DLPRES(,@BR),@ZERO            ZERO RESIDUAL CNT
     1899 5D 00 01 0D         5385+      CLC   DLPWK1+@PRCNT(1,@BR),DLPWTH(,@BR) CNT > WIDTH ?
     189D F2 04 10            5386+      JNH   DLP540                        JUMP IF NO
     18A0 5C 00 0B 01         5387+      MVC   DLPRES(1,@BR),DLPWK1+@PRCNT(,@BR) SAVE CNT
     18A4 5F 00 0B 0D         5388+      SLC   DLPRES(1,@BR),DLPWTH(,@BR) CALCULATE RESIDUAL CNT
     18A8 5C 00 01 0B         5389+      MVC   DLPWK1+@PRCNT(1,@BR),DLPRES(,@BR) SET CNT TO WIDTH
     18AC 5C 00 0E 0B         5390+      MVC   DLPNXT(1,@BR),DLPRES(,@BR) SET NEXT LINE CNT = RESIDUAL
     18B0 0D 00 03C1 03C2     5391+DLP540 CLC   $LMRGN(1),$PRPOS             ARE WE AT LEFT MARGIN ?
     18B6 F2 01 19            5392+      JNE   DLPPRL                        JUMP TO PRINT LEFT IF NOT
                              5393+*
                              5394+* SET UP FOR PRINT RIGHT OPERATION
                              5395+*
     18B9 5D 00 01 0E         5396+      CLC   DLPWK1+@PRCNT(1,@BR),DLPNXT(,@BR) CNT > NEXT CNT ?
     18BD F2 02 24            5397+      JNL   DLP560                        JUMP IF CURRENT CNT > NEXT CNT
                              5398+*                                        * NEXT LINE
     18C0 5C 00 01 0D         5399+      MVC   DLPWK1+@PRCNT(1,@BR),DLPWTH(,@BR) SET CURRENT CNT TO MAX
     18C4 5D 00 0E 0D         5400+      CLC   DLPNXT(1,@BR),DLPWTH(,@BR) NEXT LINE LESS THAN WIDTH ?
     18C8 F2 02 19            5401+      JNL   DLP560                        JUMP IF NOT
     18CB 5C 00 01 0E         5402+      MVC   DLPWK1+@PRCNT(1,@BR),DLPNXT(,@BR) SET CURRENT CNT TO
                              5403+*                                        * NEXT LINE CNT
     18CF F2 87 12            5404+      J     DLP560                        GO DO PRINTING
                              5405+*
                              5406+* SET UP FOR PRINT LEFT OPERATION
                              5407+*
                         18D2 5408+DLPPRL EQU   *                            ENTRY TO PRINT LEFT
     18D2 3C 01 07CE          5409+      MVI   $$PSIO,DLPPNT                 SET DPRINT FOR LINE MODE
     18D6 4C 00 01 03C2       5410+      MVC   DLPWK1+@PRCNT(1,@BR),$PRPOS SET CURRENT PRINT POSITION
     18DB 4F 00 01 03C1       5411+      SLC   DLPWK1+@PRCNT(1,@BR),$LMRGN GET RETURN PRINT CNT
     18E0 5F 00 01 0A         5412+      SLC   DLPWK1+@PRCNT(1,@BR),DLPONE(,@BR) SET UP FOR HARDWARE
                              5413+*
                              5414+* DO THE PRINT OPERATION
                              5415+*
     18E4 7C 40 00            5416+DLP560 MVI   DLPWK1+@PCTRL(,@BR),@PRINT SET NO CARRIER RETURN
                              5417+*                                        * PRINT LENGTH = WIDTH
     18E7 C0 87 0707          5418+      B     $$PRNT                        GO PRINT THE LINE
     18EB 1868           18EC 5419+      DC    AL2(DLPWK1)                   PPL ADDR
     18ED 3C 00 07CE          5420+      MVI   $$PSIO,@ZERO                  RESET SIO CTRL FOR NORMAL OPS
     18F1 3C 00 07E9          5421+      MVI   $$PCNT,@ZERO                  SET DPRINT PPL CNT ZERO
     18F5 C0 87 0707          5422+      B     $$PRNT                        INDEX A LINE
     18F9 1870           18FA 5423+      DC    AL2(DLPNDX)                   INDEX PPL ADDRESS
                         1785 5424+      USING DLPBSE,@BR                    USE MAINLINE BASE VALUE
     18FB C2 01 1785          5425+      LA    DLPBSE,@BR                    RESTORE MAINLINE BR
     18FF 7D 00 EE            5426+      CLI   DLPRES(,@BR),@ZERO            ANY RESIDUAL DATA ?
```

DLPRNT -- LIST OUTPUT INTERFACE

```
     1902 D0 81 A2              5427+          BE   DLP440(,@BR)              EXIT TO MAINLINE IF NOT
                                5428+*
                           1868 5429+          USING DLPBS2,@BR             USE PRINT BASE ADDR
     1905 C2 01 1868           5430+          LA   DLPBS2,@BR               SET BR
     1909 7C F4 0F             5431+          MVI  DLPREM(,@BR),DLPBLN      SET REMAINDER TO BUF LENGTH
     190C 5F 00 0F 0B          5432+          SLC  DLPREM(1,@BR),DLPRES(,@BR) GET REMAINDER FOR BLANK CNT
     1910 C2 02 1C8C           5433+          LA   DLIBUF,@XR               XR POINTS TO BUFFER
     1914 74 02 B7             5434+          ST   DLP580+@DOP2(,@BR),@XR   SET MOVE INSTR TO BUF ADDR
     1917 5E 01 B7 0D          5435+          ALC  DLP580+@DOP2(@CADDR,@BR),DLPWTH(,@BR) POINT TO RESIDUAL
     191B 8C 00 00 0000        5436+DLP580 MVC  0(1,@XR),*-*             MOVE A BYTE OF RESIDUAL DATA
     1920 E2 02 01             5437+          LA   1(,@XR),@XR              INCREMENT DATA POINTER
     1923 5E 01 B7 0A          5438+          ALC  DLP580+@DOP2(@CADDR,@BR),DLPONE(,@BR) INCREMENT DATA ADDR
     1927 5F 00 0B 0A          5439+          SLC  DLPRES(1,@BR),DLPONE(,@BR) DECREMENT RESIDUAL CNT
     192B D0 84 B3             5440+          BH   DLP580(,@BR)             DO IT AGAIN TILL DONE
     192E BC 40 00             5441+DLP600 MVI  0(,@XR),@BLANK           SET REMAINING BLANKS
     1931 E2 02 01             5442+          LA   1(,@XR),@XR              INCREMENT
     1934 5F 00 0F 0A          5443+          SLC  DLPREM(1,@BR),DLPONE(,@BR) REMAINDER ?
     1938 D0 84 C6             5444+          BH   DLP600(,@BR)             SET ANOTHER BLANK
     193B 5C 00 01 0E          5445+          MVC  DLPWK1+@PRCNT(1,@BR),DLPNXT(,@BR) SET NEXT CNT
     193F D0 87 12             5446+          B    DLPPRT(,@BR)             GO FINISH LINE

                           1868 5448+DLPBS2 EQU  DLPWK1                    BASE VALUE FOR PRINT OP
                           00F4 5449+DLPBLN EQU  244                       LENGTH OF PRINT BUFFER
                                5450+***                    END OF DLPRNT                         ***
                                5451 *
     1942 1A00            1943 5452 KSYASL DC   AL(@CADDR)(KSYLVC)        ADDR OF LETTER VAR TBL
     1944 1A1D            1945 5453 KSYALD DC   AL(@CADDR)(KSYLDC)        ADDR OF LETTER DIGIT VAR TBL
     1946 1B3F            1947 5454 KSYAAA DC   AL(@CADDR)(KSYNAC)        ADDR OF ARITH ARRAY TBL
     1948 1B5C            1949 5455 KSYACV DC   AL(@CADDR)(KSYCVC)        ADDR OF CHARACTER VAR TBL
     194A 1B79            194B 5456 KSYACA DC   AL(@CADDR)(KSYCAC)        ADDR OF CHARACTER ARRAY TBL
                                5457 *       PATCH
                                5458 ***********************************************************************
                                5459 * PATCH AREA 1                                                        *
                                5460 ***********************************************************************
                                5461 *
                                5462 * CALCULATE AREA LEFT IN THIS SECTOR
                                5463 *
                           194C 5464 $$$$L1 EQU  *                         START OF PATCH AREA 1
     1A00                       5465         ORG  *,256,0                   SET LOC CNTR TO NEXT SECTOR
                           1A00 5466 $$$$T1 EQU  *                         DEFINE ADDR OF SETR DNDRY
     194C                       5467         ORG  $$$$L1                    SET LOC CNTR TO START OF
                                5468 *                                     * PATCH AREA
     194C                  19FF 5469 $$$$$1 DS   CL($$$$T1-$$$$L1)          PATCH AREA
                                5470 ***********************************************************************
                           1A00 5471 KSYLVC EQU  *                         LETTER VAR TBL
                           1A1D 5472 KSYLDC EQU  KSYLVC+29                 LETTER DIGIT VAR TBL
                           1B3F 5473 KSYNAC EQU  KSYLDC+290                ARITH ARRAY TBL
                           1B5C 5474 KSYCVC EQU  KSYNAC+29                 CHARACTER VAR TBL
                           1B79 5475 KSYCAC EQU  KSYCVC+29                 CHARACTER ARRAY TBL
                           1B96 5476 GRTYPE EQU  KSYCAC+29                 ADDR OF STATEMENT TYPE CODE
                           1B97 5477 GRLINE EQU  GRTYPE+1                  ADDR OF STATEMENT LINE NO.
                           1B98 5478 GRTEXT EQU  GRLINE+1                  ADDR OF LEFT BYTE OF STMNT TEXT
                           1C8C 5479 DLIBUF EQU  GRTEXT+244                BUFFER FOR DIPRNT
                           1B98 5480 KSYBFR EQU  GRTEXT                    PRINT BUFFER
                                5481         PRINT  ON
                           FFFF 5482         END
```

TOTAL STATEMENTS IN ERROR IN THIS ASSEMBLY =   0

SYMBOL	LEN	VALUE	DEFN	REFERENCES								
$$$$$$	001	0C00	3012									
$$$$$1	180	19FF	5469									
$$$$L1	001	194C	5464	5467	5469							
$$$$T1	001	1A00	5466	5469								
$$$CMD	001	0020	1309									
$$$DAT	001	0040	1308									
$$$EPL	001	0091	1305									
$$$ERN	001	0080	1359									
$$$FUN	001	0010	1310									
$$$NLN	001	00A0	1355									
$$$STD	001	0081	1304									
$$BNLN	001	0605	1285	1287								
$$CDBS	001	08C0	1335									
$$CDND	001	0666	1294									
$$CDRD	001	0890	1333	1335								
$$CKEY	001	0603	1283									
$$CKFF	001	0B3D	1315									
$$COFF	001	0B44	1314	5305								
$$CSNS	001	209C	1344									
$$DATB	001	0BBF	1316									
$$EOSA	001	0AFE	1313									
$$ERSK	001	1C00	1354									
$$FITS	001	1D00	1362									
$$FLIB	001	06FF	1361									
$$ILEN	001	0601	1279	1281	1285							
$$ILHD	001	0600	1277	1279								
$$INLN	001	0607	1292	1294	1296							
$$INND	001	06FA	1296									
$$KBDT	001	09E1	1303	1307								
$$KBSN	001	09E2	1307	1312								
$$KLD1	001	0600	1367									
$$KLD2	001	0700	1369									
$$KLD3	001	0C00	1371									
$$LPOS	001	09EB	1312									
$$PCNT	001	07E9	1328	5421*								
$$PLYN	001	2004	1342	5274								
$$PRES	001	0890	1301	1303	1313	1314	1315	1316	1333	3674		
$$PRFL	001	2143	1346									
$$PRNT	001	0707	1322	1323	1327	1328	5311	5313	5321	5378	5418	5422
$$PRTN	001	0782	1323									
$$PSIO	001	07CE	1327	5409*	5420*							
$$PYCD	001	2200	1348	3671								
$$PYMP	001	2000	1340	1342	1344	1346	1348					
$$SLIB	001	1C00	1357									
$$TPCD	001	0606	1287	1292								
$$UPAR	001	0602	1281	1283								
$$WSPB	001	1E00	1360									
$$XIND	001	06FF	1358	1361								
$$ZERO	001	0000	0223	0224	0226	0227	0228	0232	1340	3263		
$ABORT	001	0010	0336									
$BASIC	001	0080	0394	4856								
$BIGCD	001	0080	0470									
$BLDPL	001	0579	0603	0605								
$BLNOE	001	0569	0593									
$BLOAD	001	0522	0584	0586	0589	0602	0603					
$BLRTN	001	0550	0592	0593								

```
$BRSAV  001  03C5 0281   0282
$BSADR  001  0587 0608   0610
$BUFPT  001  03E3 0489   0490
$CABLD  001  04B4 0562   0563
$CAERK  001  0469 0539   0542   3248   3314   3571   4961
$CAERR  001  03CD 0287   0289   3033*  3240*  3243*  3246*  3416*  3605*  3608*  3611*  3648*  3657*  3661*
                         4957*
$CAIPL  001  049D 0558   0560
$CALLI  001  0008 0479
$CARDI  001  0001 0250   3654
$CARPL  001  04A1 0560   0562   3225
$CIENT  001  0483 0549   0550
$CIEXT  001  0480 0548   0549
$CIMSK  001  0476 0545   0548   5380*
$CISUS  001  0496 0553   0558
$CLBFR  001  0010 0437
$CMDKY  001  0008 0349   3668
$CMODE  001  0002 0399
$CONFG  001  03DD 0462   0472
$CRPOS  001  03E2 0488   0489
$CRTAD  001  044D 0527   0528
$CRTAV  001  0002 0343   3651
$CRTDN  001  0002 0367
$CRTIN  001  03D3 0364   0371   3637*  5279   5290   5292*  5297*
$CRTNO  001  0004 0346
$CRTPU  001  0004 0368   5290   5292
$CRTSP  001  0008 0369   5279   5297
$CRTUP  001  0001 0366   3637
$CRUSH  001  0080 0475
$CSDPL  001  050E 0574   0575
$C0001  001  0464 0531   0537
$DATE   001  043A 0512   0513
$DBGUF  001  03E0 0474   0483
$DBLOK  001  0001 0424
$DFDET  001  03E8 0495   0496
$DISKN  001  0025 0226   4782   4879   5079
$DKERR  001  0008 0405
$DKSIZ  001  03D7 0449   0457   0498
$DK100  001  0001 0451
$DK200  001  0002 0452
$DK400  001  0004 0453
$DK600  001  0008 0454
$DK800  001  0010 0455
$DPLSV  001  0449 0523   0525
$DTNMB  001  0040 0270
$DTRDR  001  0040 0358
$ENDNU  001  0600 0617   1277   1301   1322   1358   1367   1369   1371   2005
$ERDPL  001  046F 0542   0544
$ERFIL  001  0040 0297
$ERHRD  001  0004 0429   4960
$ERKEY  001  0080 0301
$ERLOG  001  0345 0231
$ERMAD  001  0472 0544   0545
$ERPND  001  0004 0402
$ERRCT  001  03CF 0303
$ERRPG  001  03CE 0291
```

```
$ERSFL   001  0035 0296
$ERSTK   001  0030 0294
$ER050   001  0363 0232
$ER1N2   001  0050 0299
$EXADR   001  0517 0577   0579
$EXCMD   001  0001 0331
$EXFTR   001  043B 0513   0518   3670   3672   5253
$FCIND   001  0010 0409
$FDIND   001  0040 0416
$FEARR   001  0004 0224
$FEMAP   001  0588 0610   0611
$FILIB   001  03DA 0460   0461
$FITIN   001  0010 0385
$FUIND   001  0020 0414
$GUFIO   001  0583 0607   0608
$GUFIR   001  0008 0259
$HISTE   001  042E 0510   0511
$HIST1   001  0435 0511   0512
$HRDER   001  0020 0355
$INDR1   001  03D4 0371   0397   4856
$INDR2   001  03D5 0397   0422
$INDR3   001  03D6 0422   0449   4960*
$INLNO   001  03CF 0289   0291   0303   0310
$INRPT   001  0020 0267
$IOIND   001  03D2 0338   0364   3645   3651   3668* 5309
$IOPGS   001  0010 0478
$IOYES   001  0002 0253
$IPLDV   001  05FF 0614   0617
$IRKEY   001  0020 0477
$KEYBD   001  03E1 0483   0488
$KEYCD   001  03C3 0247   0281   3654
$KEYDT   001  0040 0391
$KE090   001  00DE 0227
$KE130   001  01D5 0228
$KYBSY   001  0010 0264
$LDRTN   001  0571 0602
$LEVEL   001  03DF 0472   0474
$LIST    001  0002 0426
$LMRGN   001  03C1 0242   0244   5382   5391   5411
$LNPTR   001  0080 0361   5309
$LOADB   001  054A 0586
$LOADR   001  051A 0579   0582
$LPRIO   001  03EA 0496
$LPROS   001  03E5 0491   0493
$LPRP3   001  03E4 0490   0491
$MOUNT   001  0020 0440
$MPDWN   001  0001 0340   3645
$NEXTB   001  03E6 0493   0494
$NEXTL   001  03E7 0494   0495
$NOENB   001  0008 0432
$NOLST   001  0004 0256
$NUCBS   001  03C0 0239   0240
$NWRKF   001  0080 0445
$NWRKR   001  0040 0442
$PASWD   001  042D 0509   0510
$PAUSD   001  04BA 0563   0565
```

SYMBOL	LEN	VALUE	DEFN	REFERENCES						VER 15, MOD 00 04/07/22 PAGE 70
$PAUSE	001	0002	0333							
$PGMDT	001	0020	0388							
$PGMST	001	0010	0352							
$PKERT	001	0419	0507	0509						
$PLST1	001	0454	0528	0529						
$PLST2	001	045B	0529	0530						
$PLST3	001	0462	0530	0531						
$PRDEV	001	044B	0525	0527	5261	5266				
$PRESN	001	0002	0376							
$PROCI	001	0001	0373							
$PRPOS	001	03C2	0244	0247	5391	5410				
$PSDBR	001	04FA	0568							
$PSDXR	001	04F2	0567	0568						
$PSTEP	001	0004	0334							
$PSTMT	001	0008	0335							
$PTCH1	001	03F5	0498	0502						
$READY	001	0080	0418							
$REORD	001	0040	0476							
$RLOAD	001	051E	0582	0584						
$RMRGN	001	03C0	0240	0242	5381					
$RSTR	001	04D6	0565	0567	0569	0574				
$RUNIT	001	0001	0312							
$SFAID	001	050D	0570							
$SPRNT	001	0465	0537	0539						
$SRTRN	001	04FE	0569	0570						
$STEPT	001	0002	0313							
$SWPCR	001	0511	0575	0577						
$TABLN	001	03CB	0284	0287						
$TFLOW	001	0008	0319							
$TRACE	001	0004	0314							
$TRALL	001	0010	0320							
$TROVR	001	054E	0589	0592						
$TRUNK	001	0080	0272							
$TRVAR	001	0020	0321							
$UNMSK	001	048D	0550	0553	5338					
$USRDR	001	03DC	0461	0462						
$VMDEF	001	0080	0325							
$VOLF1	001	03FE	0504	0505						
$VOLF2	001	040E	0506							
$VOLID	001	03F6	0502	0503	0507					
$VOLR1	001	03F6	0503	0504						
$VOLR2	001	0406	0505	0506						
$WAITF	001	057F	0605	0607	3223	4783	4880	5314	5379	
$WFDEF	001	0040	0519							
$WFLOK	001	0008	0382							
$WFNME	001	0443	0518	0523						
$WSIND	001	0004	0379							
$XIND1	001	03D0	0310	0329						
$XIND2	001	03D1	0329	0338						
$XIND3	001	03D8	0457	0460						
$XPREC	001	0040	0322							
$XRSAV	001	03C7	0282	0284	3024					
$ZTRAD	001	05A2	0611							
$12K	001	0004	0466							
$16CKY	001	0008	0468							
$16K	001	0002	0465							

SYMBOL	LEN	VALUE	DEFN
$22IMP	001	0001	0463
#$$#BL	001	0000	0945
#$$#CK	001	0000	1073
#$$#CN	001	0000	1041
#$$#CO	001	0000	0833
#$$#CS	001	0000	0893
#$$#DR	001	0000	0637
#$$#ER	001	0000	0837
#$$#FS	001	0000	0933
#$$#IN	001	0000	1077
#$$#PW	001	0000	1081
#$$#RS	001	0000	0913
#$$#SA	001	0000	0901
#$$#SS	001	0000	0897
#$$#VU	001	0600	0857
#$$#0T	001	0700	0629
#$$#1T	001	0000	0633
#$$BCO	001	0600	0645
#$$BOV	001	0800	0917
#$$DPR	001	0700	0653
#$$DRE	001	0889	0669
#$$DSP	001	2800	0689
#$$ECM	001	0C00	0949
#$$EFK	001	0C00	0969
#$$ERR	001	0C00	0941
#$$EXM	001	0C00	0829
#$$FIL	001	0E00	0909
#$$FIS	001	0E00	0905
#$$FML	001	0200	1037
#$$FMS	001	0200	0877
#$$GRA	001	0889	0801
#$$GUF	001	0C00	0937
#$$INL	001	0600	1017
#$$INS	001	0600	0641
#$$KAL	001	0C00	0805
#$$KCA	001	0C00	1021
#$$KCH	001	0C00	0773
#$$KCN	001	0C00	0889
#$$KCT	001	0C00	0741
#$$KDE	001	0C00	0737
#$$KDI	001	0D00	0817
#$$KDN	001	0C00	0725
#$$KDO	001	0E00	0821
#$$KED	001	0C00	0661
#$$KEN	001	0C00	0665
#$$KEX	001	0C00	0685
#$$KGO	001	0C00	0657
#$$KHE	001	0C00	0841
#$$KKE	001	0C00	1069
#$$KLI	001	0C00	0745
#$$KLL	001	0920	1045
#$$KLO	001	0C00	0749
#$$KME	001	0D00	0729
#$$KMO	001	0C00	0673
#$$KNA	001	0C00	0785
#$$KOV	001	0E00	0705

SYMBOL	LEN	VALUE	DEFN	REFERENCES	VER 15, MOD 00 04/07/22 PAGE 72
#$$KPA	001	0C00	0681		
#$$KPO	001	0C00	0769		
#$$KPR	001	0C00	0793		
#$$KRE	001	0C00	0713		
#$$KRL	001	0700	0809		
#$$KRM	001	0C00	0677		
#$$KRN	001	0700	0697		
#$$KRO	001	0D00	0701		
#$$KRS	001	0C00	1025		
#$$KRU	001	0C00	0721		
#$$KRV	001	0800	0813		
#$$KSA	001	0C00	0757		
#$$KSE	001	0E00	0797		
#$$KSO	001	0C20	0849		
#$$KSS	001	0C00	0781		
#$$KSV	001	0980	0777		
#$$KSY	001	0C00	0789	3011	
#$$KWI	001	0C00	0717		
#$$KWR	001	0C00	0709		
#$$LOA	001	0600	0649		
#$$MIP	001	0C00	0845		
#$$SDS	001	0C00	0957		
#$$SFF	001	0E00	0961		
#$$SFL	001	0F00	0953		
#$$SFO	001	1500	0925		
#$$SFS	001	0C00	0921		
#$$SPA	001	0C00	0761		
#$$SPO	001	0806	0765		
#$$SPS	001	0C00	0753		
#$$STR	001	1600	0929		
#$$TDC	001	1000	0733		
#$$TSY	001	1000	0693		
#$$TVK	001	0FC0	0869		
#$$UAL	001	0C00	0885		
#$$UAT	001	0900	0981		
#$$UCD	001	0900	0989		
#$$UCN	001	0C00	0973		
#$$UCP	001	0700	0977		
#$$UDE	001	0C00	0993		
#$$UDI	001	0C00	0997		
#$$UEX	001	0C00	0881		
#$$UIN	001	0C00	0985		
#$$UPA	001	0C00	0965		
#$$UPO	001	0C00	1033		
#$$UPT	001	0C00	1029		
#$$VCR	001	2000	0825		
#$$VLO	001	0600	0861		
#$$VOD	001	0600	0865		
#$$VVM	001	0000	0873		
#$$VXI	001	0600	0853		
#$$ZDU	001	1100	1005		
#$$ZLB	001	1100	1049		
#$$ZLO	001	1100	1009		
#$$ZLV	001	0F00	1065		
#$$ZL1	001	0F00	1053		
#$$ZL2	001	0F00	1057		

```
                                  CROSS REFERENCE

SYMBOL  LEN VALUE DEFN  REFERENCES                    VER 15, MOD 00  04/07/22  PAGE  73

#$$ZL3  001  0C00 1061
#$$ZTR  001  1000 1001
#$$ZUT  001  0C00 1013
#$#BLN  001  18D4 0944
#$#CKT  001  2118 1072
#$#CNF  001  2000 1040
#$#COR  001  0800 0832
#$#CSA  001  1000 0892
#$#DRT  001  0000 0636
#$#ERM  001  0928 0836
#$#FSP  001  1880 0932
#$#INV  001  212C 1076
#$#PWR  001  2300 1080
#$#RSP  001  1780 0912
#$#SAV  001  1180 0900
#$#SSA  001  1128 0896
#$#VUF  001  0B08 0856
#$#0TR  001  0000 0628
#$#1TR  001  0080 0632
#$@#BL  001  0001 0946
#$@#CK  001  0004 1074
#$@#CN  001  0001 1042
#$@#CO  001  003A 0834
#$@#CS  001  003A 0894
#$@#DR  001  0008 0638
#$@#ER  001  0032 0838
#$@#FS  001  0030 0934
#$@#IN  001  003A 1078
#$@#PW  001  00C0 1082
#$@#RS  001  0030 0914
#$@#SA  001  0108 0902
#$@#SS  001  0001 0898
#$@#VU  001  0002 0858
#$@#0T  001  0018 0630
#$@#1T  001  0018 0634
#$@BCO  001  0018 0646
#$@BOV  001  0018 0918
#$@DPR  001  0005 0654
#$@DRE  001  0001 0670
#$@DSP  001  0004 0690
#$@ECM  001  0006 0950
#$@EFK  001  0002 0970
#$@ERR  001  0003 0942
#$@EXM  001  0003 0830
#$@FIL  001  0009 0910
#$@FIS  001  0009 0906
#$@FML  001  0052 1038
#$@FMS  001  0052 0878
#$@GRA  001  0003 0802
#$@GUF  001  0010 0938
#$@INL  001  0010 1018
#$@INS  001  0010 0642
#$@KAL  001  000F 0806
#$@KCA  001  000C 1022
#$@KCH  001  000C 0774
#$@KCN  001  0010 0890
```

SYMBOL	LEN	VALUE	DEFN	REFERENCES
#$@KCT	001	0009	0742	
#$@KDE	001	0010	0738	
#$@KDI	001	0005	0818	
#$@KDN	001	0010	0726	
#$@KDO	001	000C	0822	
#$@KED	001	000E	0662	
#$@KEN	001	0006	0666	
#$@KEX	001	0003	0686	
#$@KGO	001	0002	0658	
#$@KHE	001	000C	0842	
#$@KKE	001	0006	1070	
#$@KLI	001	0011	0746	
#$@KLL	001	0001	1046	
#$@KLO	001	0008	0750	
#$@KME	001	0003	0730	
#$@KMO	001	0004	0674	
#$@KNA	001	0008	0786	
#$@KOV	001	0009	0706	
#$@KPA	001	0005	0682	
#$@KPO	001	000D	0770	
#$@KPR	001	0009	0794	
#$@KRE	001	0002	0714	
#$@KRL	001	0004	0810	
#$@KRM	001	0003	0678	
#$@KRN	001	0003	0698	
#$@KRO	001	000A	0702	
#$@KRS	001	000A	1026	
#$@KRU	001	0003	0722	
#$@KRV	001	000D	0814	
#$@KSA	001	0011	0758	
#$@KSE	001	0004	0798	
#$@KSO	001	0005	0850	
#$@KSS	001	000B	0782	
#$@KSV	001	0002	0778	
#$@KSY	001	000F	0790	
#$@KWI	001	0002	0718	
#$@KWR	001	0002	0710	
#$@LOA	001	0013	0650	
#$@MIP	001	000D	0846	
#$@SDS	001	0004	0958	
#$@SFF	001	0008	0962	
#$@SFL	001	0005	0954	
#$@SFO	001	0003	0926	
#$@SFS	001	0011	0922	
#$@SPA	001	0004	0762	
#$@SPO	001	0003	0766	
#$@SPS	001	0001	0754	
#$@STR	001	0002	0930	
#$@TDC	001	0003	0734	
#$@TSY	001	0003	0694	
#$@TVK	001	0001	0870	
#$@UAL	001	0011	0886	
#$@UAT	001	000C	0982	
#$@UCD	001	000B	0990	
#$@UCN	001	0009	0974	
#$@UCP	001	000F	0978	

SYMBOL	LEN	VALUE	DEFN	REFERENCES	VER 15, MOD 00 04/07/22 PAGE 75
#$@UDE	001	000E	0994		
#$@UDI	001	0008	0998		
#$@UEX	001	000E	0882		
#$@UIN	001	000F	0986		
#$@UPA	001	0004	0966		
#$@UPO	001	0005	1034		
#$@UPT	001	0012	1030		
#$@VCR	001	0008	0826		
#$@VLO	001	0002	0862		
#$@VOD	001	0016	0866		
#$@VVM	001	0030	0874		
#$@VXI	001	0002	0854		
#$@ZDU	001	0008	1006		
#$@ZLB	001	0002	1050		
#$@ZLO	001	000C	1010		
#$@ZLV	001	0006	1066		
#$@ZL1	001	0007	1054		
#$@ZL2	001	000D	1058		
#$@ZL3	001	000A	1062		
#$@ZTR	001	0001	1002		
#$@ZUT	001	0014	1014		
#$BCOM	001	0080	0644		
#$BOLV	001	1780	0916		
#$DPRI	001	014C	0652		
#$DREA	001	0200	0668		
#$DSPL	001	0240	0688		
#$ECMA	001	1900	0948		
#$EFKE	001	1990	0968		
#$ERRP	001	18C0	0940		
#$EXMS	001	07D4	0828		
#$FILN	001	1724	0908		
#$FIST	001	1700	0904		
#$FMLN	001	1E00	1036		
#$FMST	001	0D00	0876		
#$GRAP	001	0690	0800		
#$GUFU	001	1880	0936		
#$INLN	001	1C84	1016		
#$INST	001	0020	0640		
#$KALL	001	06A4	0804		
#$KCAL	001	1CC4	1020		
#$KCHA	001	053C	0772		
#$KCND	001	0F80	0888		
#$KCTL	001	03BC	0740		
#$KDEL	001	035C	0736		
#$KDIS	001	0744	0816		
#$KDNT	001	0300	0724		
#$KDOV	001	0780	0820		
#$KEDI	001	0188	0660		
#$KENA	001	01C4	0664		
#$KEXT	001	0234	0684		
#$KGOS	001	0180	0656		
#$KHEL	001	0A30	0840		
#$KKEY	001	2100	1068		
#$KLIS	001	0400	0744		
#$KLLA	001	2004	1044		
#$KLOG	001	0444	0748		

```
#$KMER   001   030C 0728
#$KMOU   001   0204 0672
#$KNAM   001   05C0 0784
#$KOVM   001   0290 0704
#$KPAS   001   0220 0680
#$KPOO   001   0508 0768
#$KPRT   001   063C 0792
#$KREA   001   02BC 0712
#$KRLA   001   0700 0808
#$KRMO   001   0214 0676
#$KRNU   001   0280 0696
#$KROV   001   028C 0700
#$KRSU   001   1D24 1024
#$KRUN   001   02CC 0720
#$KRVL   001   0710 0812
#$KSAV   001   0488 0756
#$KSET   001   0680 0796
#$KSOV   001   0AC8 0848
#$KSSP   001   0594 0780
#$KSVL   001   058C 0776
#$KSYM   001   0600 0788
#$KWID   001   02C4 0716
#$KWRI   001   02B4 0708
#$LOAD   001   0100 0648
#$MIPP   001   0A80 0844
#$SDSY   001   192C 0956
#$SFFI   001   193C 0960
#$SFLO   001   1918 0952
#$SFOV   001   1844 0924
#$SFSY   001   1800 0920
#$SPAC   001   04CC 0760
#$SPOV   001   04DC 0764
#$SPSY   001   0484 0752
#$STRO   001   1850 0928
#$TDCK   001   0350 0732
#$TSYK   001   0250 0692
#$TVKB   001   0BAC 0868
#$UALL   001   0F00 0884
#$UATR   001   1A38 0980
#$UCDI   001   1AD8 0988
#$UCNF   001   19B8 0972
#$UCPL   001   19DC 0976
#$UDEL   001   1B24 0992
#$UDIS   001   1B5C 0996
#$UEXL   001   0EA8 0880
#$UINI   001   1A88 0984
#$UPAC   001   1980 0964
#$UPOV   001   1D24 1032
#$UPTF   001   1D5C 1028
#$VCRT   001   07B4 0824
#$VLOA   001   0B80 0860
#$VODK   001   0B88 0864
#$VVMR   001   0C00 0872
#$VXIT   001   0B00 0852
#$ZDUM   001   1BA4 1004
#$ZLBM   001   2008 1048
```

```
#$ZLOA   001  1BC4 1008
#$ZLVR   001  20B0 1064
#$ZL1M   001  2010 1052
#$ZL2M   001  2030 1056
#$ZL3M   001  2088 1060
#$ZTRA   001  1B9C 1000
#$ZUTM   001  1C14 1012
#KSYM    001  0C07 3015
#KSYMB   001  0000 0001
@@E001   001  0000 1909   1911
@@E003   001  0001 1911   1913
@@E004   001  0002 1913   1915
@@E005   001  0003 1915   1917
@@E006   001  0004 1917   1919
@@E007   001  0005 1919   1921
@@E008   001  0006 1921   1923
@@E009   001  0007 1923   1925
@@E010   001  0008 1925   1927
@@E011   001  0009 1927   1929
@@E012   001  000A 1929   1931
@@E013   001  000B 1931   1933
@@E014   001  000C 1933   1935
@@E015   001  000D 1935   1937
@@E016   001  000E 1937   1939
@@E017   001  000F 1939   1941
@@E018   001  0010 1941   1943
@@E019   001  0011 1943   1945
@@E020   001  0012 1945   1947
@@E021   001  0013 1947   1949
@@E023   001  0014 1949   1951
@@E024   001  0015 1951   1953
@@E025   001  0016 1953   1955
@@E026   001  0017 1955   1957
@@E027   001  0018 1957   1959
@@E028   001  0019 1959   1961
@@E029   001  001A 1961   1963
@@E030   001  001B 1963   1965
@@E031   001  001C 1965   1967
@@E032   001  001D 1967   1969
@@E035   001  001E 1969   1971
@@E036   001  001F 1971   1973
@@E037   001  0020 1973   1975
@@E038   001  0021 1975   1977
@@E039   001  0022 1977   1979
@@E040   001  0023 1979   1981
@@E041   001  0024 1981   1983
@@E042   001  0025 1983   1985
@@E043   001  0026 1985   1987
@@E044   001  0027 1987   1989
@@E045   001  0028 1989   1991
@@E046   001  0029 1991   1993
@@E060   001  002A 1993   1995
@@E080   001  002B 1995
@@E100   001  0000 1381   1383
@@E101   001  0001 1383   1385
@@E102   001  0002 1385   1387
```

```
@@E103   001  0003 1387   1389
@@E110   001  0004 1389   1391   3416
@@E112   001  0005 1391   1393
@@E113   001  0006 1393   1395
@@E114   001  0007 1395   1397
@@E115   001  0008 1397   1399
@@E116   001  0009 1399   1401
@@E117   001  000A 1401   1403
@@E120   001  000B 1403   1405
@@E122   001  000C 1405   1407
@@E123   001  000D 1407   1409
@@E124   001  000E 1409   1411
@@E129   001  000F 1411   1413
@@E130   001  0010 1413   1415
@@E131   001  0011 1415   1417   3033   3605
@@E133   001  0012 1417   1419   3246
@@E134   001  0013 1419   1421   3611
@@E135   001  0014 1421   1423
@@E136   001  0015 1423   1425   3608
@@E137   001  0016 1425   1427
@@E138   001  0017 1427   1429
@@E139   001  0018 1429   1431   3240
@@E142   001  0019 1431   1433
@@E143   001  001A 1433   1435
@@E150   001  001B 1435   1437
@@E151   001  001C 1437   1439
@@E160   001  001D 1439   1441
@@E162   001  001E 1441   1443
@@E163   001  001F 1443   1445
@@E164   001  0020 1445   1447
@@E200   001  0021 1447   1449
@@E205   001  0022 1449   1451
@@E210   001  0023 1451   1453
@@E211   001  0024 1453   1455
@@E212   001  0025 1455   1457
@@E213   001  0026 1457   1459
@@E215   001  0027 1459   1461
@@E216   001  0028 1461   1463
@@E217   001  0029 1463   1465
@@E220   001  002A 1465   1467
@@E221   001  002B 1467   1469
@@E222   001  002C 1469   1471
@@E223   001  002D 1471   1473
@@E225   001  002E 1473   1475
@@E226   001  002F 1475   1477
@@E227   001  0030 1477   1479
@@E228   001  0031 1479   1481
@@E229   001  0032 1481   1483
@@E230   001  0033 1483   1485
@@E232   001  0034 1485   1487
@@E234   001  0035 1487   1489
@@E237   001  0036 1489   1491
@@E240   001  0037 1491   1493
@@E241   001  0038 1493   1495   3661
@@E242   001  0039 1495   1497
@@E248   001  003A 1497   1499   3657
```

```
@@E249  001  003B 1499   1501
@@E250  001  003C 1501   1503
@@E251  001  003D 1503   1505
@@E252  001  003E 1505   1507
@@E253  001  003F 1507   1509
@@E254  001  0040 1509   1511
@@E255  001  0041 1511   1513  3243
@@E256  001  0042 1513   1515
@@E300  001  0043 1515   1517
@@E301  001  0044 1517   1519
@@E302  001  0045 1519   1521
@@E303  001  0046 1521   1523
@@E304  001  0047 1523   1525
@@E305  001  0048 1525   1527
@@E308  001  0049 1527   1529
@@E310  001  004A 1529   1531
@@E315  001  004B 1531   1533
@@E316  001  004C 1533   1535
@@E320  001  004D 1535   1537
@@E325  001  004E 1537   1539
@@E330  001  004F 1539   1541
@@E335  001  0050 1541   1543
@@E338  001  0051 1543   1545
@@E340  001  0052 1545   1547
@@E350  001  0053 1547   1549
@@E351  001  0054 1549   1551
@@E352  001  0055 1551   1553
@@E360  001  0056 1553   1555
@@E361  001  0057 1555   1557
@@E362  001  0058 1557   1559
@@E371  001  0059 1559   1561
@@E380  001  005A 1561   1563
@@E390  001  005B 1563   1565
@@E400  001  005C 1565   1567
@@E410  001  005D 1567   1569
@@E415  001  005E 1569   1571
@@E417  001  005F 1571   1573
@@E420  001  0060 1573   1575
@@E430  001  0061 1575   1577
@@E432  001  0062 1577   1579
@@E433  001  0063 1579   1581
@@E450  001  0064 1581   1583
@@E451  001  0065 1583   1585
@@E460  001  0066 1585   1587
@@E461  001  0067 1587   1589
@@E464  001  0068 1589   1591
@@E465  001  0069 1591   1593
@@E466  001  006A 1593   1595
@@E467  001  006B 1595   1597
@@E469  001  006C 1597   1599
@@E470  001  006D 1599   1601
@@E471  001  006E 1601   1603
@@E473  001  006F 1603   1605
@@E474  001  0070 1605   1607
@@E475  001  0071 1607   1609
@@E476  001  0072 1609   1611
```

```
@@E477  001  0073 1611    1613
@@E478  001  0074 1613    1615
@@E479  001  0075 1615    1617
@@E480  001  0076 1617    1619
@@E481  001  0077 1619    1621
@@E482  001  0078 1621    1623
@@E483  001  0079 1623    1625
@@E484  001  007A 1625    1627
@@E485  001  007B 1627    1629
@@E486  001  007C 1629    1631
@@E487  001  007D 1631    1633
@@E488  001  007E 1633    1635
@@E489  001  007F 1635    1637
@@E490  001  0080 1637    1639
@@E491  001  0081 1639    1641
@@E492  001  0082 1641    1643
@@E493  001  0083 1643    1645
@@E494  001  0084 1645    1647
@@E495  001  0085 1647    1649
@@E496  001  0086 1649    1651
@@E497  001  0087 1651    1653
@@E498  001  0088 1653    1655
@@E500  001  0089 1655    1657
@@E501  001  008A 1657    1659
@@E530  001  008B 1659    1661
@@E531  001  008C 1661    1663
@@E535  001  008D 1663    1665
@@E540  001  008E 1665    1667
@@E541  001  008F 1667    1669
@@E542  001  0090 1669    1671
@@E543  001  0091 1671    1673
@@E544  001  0092 1673    1675
@@E545  001  0093 1675    1677
@@E546  001  0094 1677    1679
@@E547  001  0095 1679    1681
@@E548  001  FFFF 1885
@@E549  001  0096 1681    1683  3648
@@E550  001  0097 1683    1685  4784
@@E551  001  0098 1685    1687  4957
@@E552  001  0099 1687    1689
@@E553  001  009A 1689    1691
@@E554  001  009B 1691    1693
@@E555  001  009C 1693    1695
@@E556  001  009D 1695    1697
@@E558  001  009E 1697    1699
@@E570  001  009F 1699    1701
@@E571  001  00A0 1701    1703
@@E572  001  00A1 1703    1705
@@E573  001  00A2 1705    1707
@@E574  001  00A3 1707    1709
@@E575  001  FFFF 1887
@@E578  001  00A4 1709    1711
@@E579  001  FFFF 1889
@@E580  001  FFFF 1891
@@E585  001  00A5 1711    1713
@@E595  001  FFFF 1893
```

SYMBOL	LEN	VALUE	DEFN	REFERENCES
@@E597	001	FFFF	1895	
@@E598	001	FFFF	1897	
@@E600	001	00A6	1713	1715
@@E601	001	00A7	1715	1717
@@E602	001	00A8	1717	1719
@@E603	001	00A9	1719	1721
@@E604	001	00AA	1721	1723
@@E606	001	00AB	1723	1725
@@E607	001	00AC	1725	1727
@@E608	001	00AD	1727	1729
@@E609	001	00AE	1729	1731
@@E610	001	00AF	1731	1733
@@E611	001	00B0	1733	1735
@@E612	001	00B1	1735	1737
@@E613	001	00B2	1737	1739
@@E614	001	00B3	1739	1741
@@E700	001	00B4	1741	1743
@@E701	001	00B5	1743	1745
@@E710	001	00B6	1745	1747
@@E712	001	00B7	1747	1749
@@E713	001	00B8	1749	1751
@@E714	001	00B9	1751	1753
@@E715	001	00BA	1753	1755
@@E716	001	00BB	1755	1757
@@E717	001	00BC	1757	1759
@@E718	001	00BD	1759	1761
@@E720	001	00BE	1761	1763
@@E721	001	00BF	1763	1765
@@E723	001	00C0	1765	1767
@@E724	001	00C1	1767	1769
@@E725	001	00C2	1769	1771
@@E726	001	00C3	1771	1773
@@E727	001	00C4	1773	1775
@@E728	001	00C5	1775	1777
@@E729	001	00C6	1777	1779
@@E730	001	00C7	1779	1781
@@E732	001	00C8	1781	1783
@@E752	001	00C9	1783	1785
@@E753	001	00CA	1785	1787
@@E754	001	00CB	1787	1789
@@E755	001	00CC	1789	1791
@@E756	001	00CD	1791	1793
@@E757	001	00CE	1793	1795
@@E758	001	00CF	1795	1797
@@E759	001	00D0	1797	1799
@@E760	001	00D1	1799	1801
@@E761	001	00D2	1801	1803
@@E762	001	00D3	1803	1805
@@E763	001	00D4	1805	1807
@@E764	001	00D5	1807	1809
@@E765	001	00D6	1809	1811
@@E766	001	00D7	1811	1813
@@E767	001	00D8	1813	1815
@@E768	001	00D9	1815	1817
@@E769	001	00DA	1817	1819
@@E770	001	00DB	1819	1821

SYMBOL	LEN	VALUE	DEFN	REFERENCES										

SYMBOL	LEN	VALUE	DEFN	REFERENCES										
@@E771	001	00DC	1821	1823										
@@E772	001	00DD	1823	1825										
@@E773	001	00DE	1825	1827										
@@E774	001	00DF	1827	1829										
@@E775	001	00E0	1829	1831										
@@E776	001	00E1	1831	1833										
@@E777	001	00E2	1833	1835										
@@E778	001	00E3	1835	1837										
@@E779	001	00E4	1837	1839										
@@E780	001	00E5	1839	1841										
@@E781	001	00E6	1841	1843										
@@E782	001	00E7	1843	1845										
@@E783	001	00E8	1845	1847										
@@E784	001	00E9	1847	1849										
@@E785	001	00EA	1849	1851										
@@E786	001	00EB	1851	1853										
@@E790	001	00EC	1853	1855										
@@E791	001	00ED	1855	1857										
@@E792	001	00EE	1857	1859										
@@E793	001	00EF	1859	1861										
@@E794	001	00F0	1861	1863										
@@E795	001	00F1	1863	1865										
@@E796	001	00F2	1865	1867										
@@E797	001	00F3	1867	1869										
@@E798	001	00F4	1869	1871										
@@E800	001	FFFF	1899											
@@E801	001	FFFF	1901											
@@E802	001	FFFF	1903											
@@E803	001	FFFF	1905											
@@E804	001	FFFF	1907											
@@E900	001	00F5	1871	1873										
@@E901	001	00F6	1873	1875										
@@E902	001	00F7	1875	1877										
@@E903	001	00F8	1877	1879										
@@E905	001	00F9	1879	1881										
@@E906	001	00FA	1881	1883										
@@E910	001	00FB	1883											
@ALTFL	001	0001	1121											
@ARR	001	0008	0016	3227	3414	3545	3636	3803	4098	4340	4363	4443	4763	4878	5047*
				5048	5049*	5050	5246*	5247	5248*	5249				
@ASIGN	001	007C	0071	3855	4138	4414								
@ASTER	001	005C	0069											
@BCRDL	001	0050	0088											
@BE	001	0081	0043	3231										
@BF	001	0090	0052											
@BH	001	0084	0041											
@BKSPC	001	0010	1218											
@BL	001	0082	0042											
@BLANK	001	0040	0065	3419	3425	4448	5328	5441						
@BM	001	0082	0054											
@BNE	001	0001	0046	3212	3410									
@BNH	001	0004	0044											
@BNL	001	0002	0045											
@BNM	001	0002	0057											
@BNOL	001	0020	0050											
@BNOZ	001	0008	0049											

```
@BNP    001  0004 0056
@BNZ    001  0001 0058
@BOL    001  00A0 0048
@BOZ    001  0088 0047
@BP     001  0084 0053
@BR     001  0001 0013    3054* 3061  3065  3065* 3068* 3072* 3090* 3097* 3103* 3105* 3107  3115
                          3125* 3135  3140  3145  3172  3178  3180  3214  3229  3229* 3237* 3804
                          3805  3806* 3807  3808  3815  3815  3817  3817  3818  3818  3819  3819
                          3820* 3821* 3822  3823  3823* 3824  3846* 3847  3862  3872* 3876  3881
                          3884  3899* 3903  3909  3910  3916  3917  3918  3928  3930  3931  3932
                          3933  3934  3938  3938  3940  3940  3945  3948  3949  3950  3956  3956
                          3958  3958  3960  3960  3977* 3981  3983  3998* 4002  4007  4008  4019*
                          4023  4029  4043* 4047  4052  4053  4060  4068  4069  4071  4072* 4084
                          4094  4103  4108  4112  4113  4147  4149  4156  4156  4157  4158  4165
                          4176  4176  4177  4189  4195  4207  4207  4209  4209  4213  4213  4215
                          4215  4220  4221  4226  4226  4228  4228  4237  4239  4244  4250  4250
                          4255  4262  4262  4267  4270  4271  4272  4279  4281  4282  4291  4297
                          4298  4308  4309  4309  4310  4349  4349  4364  4373  4378  4379  4380
                          4386  4389  4401  4401  4405  4405  4422  4424  4430  4431  4443  4444
                          4453  4454  4454  4759  4761  4762* 4764  4769  4771  4777  4778  4779
                          4779  4780  4781  4781  4784  4785  4785  4788  4789  4790  4790  4797
                          4799  4800  4806* 4810  4812  4815  4816  4817  4825  4831  4834  4835
                          4836  4837  4843  4844  4847  4848  4849  4850  4854  4854  4860  4860
                          4863  4865  4865  4867  4867  4868  4872  4872  4873  4874  4878  4885
                          4886  4886  4887  4888  4891  4892  4893  4893  4896  5043  5044  5046*
                          5047  5048  5049  5050  5052  5053  5053  5054  5056  5057  5059  5061
                          5061  5062  5062  5063  5065  5067  5068  5068  5069  5071  5073  5074
                          5074  5075  5075  5076  5076  5077  5084* 5104  5104  5106  5106  5107
                          5108  5109  5109  5110  5110  5111  5112  5112  5113  5114  5115  5115
                          5116  5118  5118  5119  5119  5120  5120  5121  5121  5122  5241  5243
                          5244* 5245  5246  5248  5249  5251  5252  5253  5263  5263  5276  5278
                          5282  5283  5285  5286  5286  5287  5287  5288  5289  5293  5294  5299
                          5299  5300  5300  5301  5303  5303  5316  5318  5319  5323  5323  5324
                          5324  5325  5326  5330  5330  5331  5331  5332  5332  5333  5333* 5334
                          5336* 5375  5377* 5381  5382  5383  5383  5384  5385  5385  5387  5387
                          5388  5388  5389  5389  5390  5390  5396  5396  5399  5399  5400  5400
                          5402  5402  5410  5411  5412  5412  5416  5424  5425* 5426  5427  5429
                          5430* 5431  5432  5432  5434  5435  5435  5438  5438  5439  5439  5440
                          5443  5443  5444  5445  5445  5446
@BT     001  0010 0051
@BZ     001  0081 0055
@BZ37B  001  00F2 1231
@B1     001  0001 0063    4211  4292  4902
@CADDR  001  0002 0142    2754  2755  2756  3081  3086  3126  3147  3195  3204  3217  3223  3234
                          3256  3262  3264  3266  3268  3270  4278  4590  4593  4596  4599  4602
                          4605  4608  4611  4614  4617  4620  4623  4626  4629  4632  4635  4638
                          4641  4644  4647  4650  4653  4656  4659  4662  4665  4668  4671  4674
                          4677  4680  4683  4686  4689  4692  4695  4698  4701  4704  4707  4710
                          4713  4716  4719  4722  4785  4825  4854  4860  4865  4867  5053  5263
                          5435  5438  5452  5453  5454  5455  5456
@CARDL  001  0060 0087    1294
@CC37B  001  0000 1227
@CD37B  001  00F0 1245
@CHARA  001  00C1 0072    3849  4132  4416
@CHARF  001  00C6 0073
@CHARR  001  00D9 0074
```

```
@CHARZ   001  00E9 0075   3851   4134   4418
@CKY01   001  0001 1179
@CKY02   001  0002 1180
@CKY03   001  0003 1181
@CKY04   001  0004 1182
@CKY05   001  0005 1183
@CKY06   001  0006 1184
@CKY07   001  0007 1185
@CKY08   001  0008 1186
@CKY09   001  0009 1187
@CKY10   001  000A 1188
@CKY11   001  000B 1189
@CKY12   001  000C 1190
@CKY13   001  000D 1191   5354
@CKY14   001  000E 1192
@CKY15   001  000F 1193
@CKY16   001  0010 1194
@CLOFF   001  0010 0094
@CLON    001  0011 0093
@CMLON   001  0001 1197   5278*
@CMOFF   001  0000 1196   5282*
@COMMA   001  006B 0066   3421
@CPLUS   001  004E 0079
@CP37B   001  0004 1258
@CRERR   001  0090 1213
@CRPRY   001  0004 1217
@CRTDS   001  0092 1210
@CRTQ    001  0090 1212
@CURSR   001  0040 1214
@DADDR   001  0002 0140   4779   5052
@DBFR1   001  0004 0129   5119*
@DBFR2   001  0005 0130
@DBUSY   001  0002 1115
@DCALK   001  0001 0081
@DCBCY   001  0009 0115   2583
@DCBT1   001  0050 0117   2586
@DCFLN   001  0004 1099
@DCNT    001  0003 0128   5101
@DCRID   001  0001 1113
@DCST1   001  0040 0116   2584
@DCTRL   001  0000 0125
@DCTRW   001  0000 1112
@DCWID   001  0001 1109
@DCYL    001  0001 0126   5089
@DCYMV   001  0001 1100
@DD2     001  0003 0030   5330* 5331* 5332
@DEFLG   001  0002 1122
@DERCE   001  0020 1152
@DERD2   001  0008 1144
@DEREQ   001  0010 1143
@DERIN   001  0040 1141
@DERMA   001  0020 1142
@DERNR   001  0004 1145
@DERR    001  0000 1116
@DERSC   001  0001 1147
@DERTC   001  0002 1146
```

```
SYMBOL   LEN VALUE DEFN   REFERENCES                          VER 15, MOD 00  04/07/22  PAGE  85

@DFCR    001 0006 1102
@DFDR    001 0004 1103
@DGET    001 0001 0134   4908
@DHARD   001 0000 1130
@DLNCT   001 000F 1216
@DLNLG   001 0040 1215
@DOLAR   001 005B 0068   3857   4140   4410
@DOP2    001 0004 0028   3195*  3204*  5048*  5052*  5053*  5124   5125   5434*  5435*  5438*
@DPLNG   001 0006 0132   5054   5088
@DPOS    001 0000 0133
@DPUT    001 0002 0135   4900
@DREAD   001 0001 1106
@DSAD    001 0002 0127   5090
@DSBCY   001 0004 0106   2521
@DSBSY   001 0092 1211
@DSCS1   001 0000 0107   2522
@DSEEK   001 0000 1105
@DSIVF   001 0003 0138
@DSPIN   001 0002 0131
@DTRSZ   001 0018 0085
@DUNSF   001 0080 1148
@DVBCY   001 0007 0108   2580
@DVERY   001 0003 1111
@DVRFY   001 0031 0136
@DVST1   001 0002 1117
@DVST2   001 0003 1118
@DWAIT   001 00FF 0137
@DWBCY   001 0005 0103   2577
@DWRIT   001 0002 1107
@DWSIZ   001 00C0 0105
@DWTB1   001 0003 0104   2578
@DZERO   001 00F0 0064
@D1      001 0002 0026   3193*  3202*  3670*  3672*  4236*  4244*  5332*
@EOF     001 001C 0077   3049   4828
@EOFTC   001 0075 0162   4941
@EOS     001 001E 0076   2593   3029   3036   3058   3427   3574   4394
@ER37B   001 00F0 1232
@FDDBC   001 0000 0195
@FDE1    001 000C 0200
@FDFNA   001 000B 0198
@FDHLN   001 0002 0208
@FDLNC   001 0002 0193
@FDNSC   001 0003 0210
@FDSD    001 0000 0206
@FLACE   001 0009 0197
@FLDBC   001 0001 0196
@FLDIN   001 0012 1204
@FLENT   001 0004 0201
@FLFNA   001 0002 0199
@FLHLN   001 0002 0209
@FLLNC   001 0002 0194
@FLNSC   001 0001 0211
@FLSD    001 0001 0207
@HDRLN   001 0007 0092   1322
@HSTAD   001 0009 1128
@HSTEN   001 0007 1127
```

```
@HSTPE  001  0006 1126
@HSTQR  001  0001 1124
@HSTSN  001  0005 1125
@HSTVI  001  000F 1129
@IAR    001  0010 0017
@ID37B  001  0040 1268
@INDEX  001  0001 0156   0157  5361
@INST3  001  0003 0032
@INST4  001  0004 0033
@INST5  001  0005 0034
@INST6  001  0006 0035
@IP37B  001  00C0 1267
@I1IAR  001  00C0 0020
@KCMDK  001  0020 1178
@KELOK  001  001B 1177
@KENAB  001  001E 1175
@KEXIT  001  001F 1176
@KEYBD  001  0010 1195   5278* 5282*
@KFUNK  001  0010 1198
@KHARD  001  0011 1203
@KLEAR  001  000D 1199
@LINSZ  001  00F4 0084   1296
@LO37B  001  00F0 1236
@MAPEN  001  0005 0089
@MINCR  001  2000 0083
@MINUS  001  0060 0080
@NOP    001  0080 0040   3191  3200  3228  3577  3618  4834  5057  5281  5285  5380
@NORFL  001  0000 1123
@NTRDY  001  00A0 1260
@NUMBR  001  007B 0070   3853  4136  4412
@OPD2   001  0004 0029
@OP1    001  0003 0027   3078* 3174* 3227* 3414* 3545* 3546* 3636* 3803* 3804* 3915* 3929* 3947*
                        4067* 4098* 4104* 4234* 4235* 4269* 4278  4278* 4340* 4363* 4443* 4804*
                        4807  4809  4862  4870  4898  5044* 5050* 5243* 5245* 5247* 5249* 5263*
                        5269
@OP2    001  0005 0031
@OVRUN  001  0004 1153
@PBUSY  001  00E2 1165
@PCAR   001  00E6 1162
@PCNT   001  0003 1097
@PCTRL  001  0000 0149   5276  5316  5416*
@PCYL   001  0001 1095
@PC37B  001  00F2 1252
@PDAR   001  00E4 1161
@PDATA  001  0003 0151   5251  5251* 5333
@PD37B  001  0080 1266
@PERR   001  00E0 1168
@PFLAG  001  0000 1094
@PFORM  001  00E1 1166
@PGCSZ  001  0020 0082   0083
@PLITE  001  00E2 1167
@PLNGH  001  0004 1158
@PMGCK  001  0020 1169
@PN37B  001  00F0 1251
@PPLNG  001  0004 0148   5251  5360
@PRCNT  001  0001 0150   5318* 5323  5323* 5324  5324* 5330  5383  5385  5387  5389* 5396  5399*
```

```
                                5402* 5410* 5411* 5412* 5445*
@PRETR  001  00C0 0154    3291
@PRINT  001  0040 0152    0154  5416
@PRITY  001  0080 1202
@PSAD   001  0002 1096
@PSIOQ  001  00E0 1164
@PSIOR  001  0000 1163
@PSNSQ  001  00E2 1170
@PSR    001  0004 0015    3559* 3600* 3614*
@PWAIT  001  00FF 0158    5276  5316  5319  5325
@P1IAR  001  0020 0018
@P2IAR  001  0040 0019
@Q      001  0001 0024    3194* 3197* 3203* 3206* 3212* 3228* 3433  3564* 3618* 4784* 4831* 4834*
                          4847* 4853  5056* 5057* 5067* 5073* 5099  5100  5102  5111* 5113  5283*
                          5285*
@RD37B  001  00F1 1246
@REGL   001  0002 0012
@RETRN  001  0080 0153    0154  5368
@RLDWN  001  004F 0159
@RTCNT  001  0003 1160
@RTRNC  001  0080 0161
@RT37B  001  0005 1259
@SBLN   001  0005 0170    4938
@SBLNL  001  0002 0184
@SCTSZ  001  0100 0100
@SDFLN  001  0007 0090
@SDF0   001  0000 0166    4942
@SDF1   001  0001 0167    4943
@SDF2   001  0002 0168    4944
@SDF3   001  0003 0169
@SECCY  001  0030 0086
@SIST   001  0001 0181
@SKCTL  001  0000 1110
@SLASH  001  0061 0067
@SLAST  001  0002 0183    4845
@SMIDL  001  0003 0182
@SNSB0  001  0000 1134
@SNSB1  001  0001 1135
@SNSB2  001  0002 1136
@SNSB3  001  0003 1137
@SNULL  001  0080 0173    4802  4811
@SN37B  001  00F2 1240
@SONLY  001  0000 0180    4832
@SPINA  001  00A0 1119
@SPINB  001  00B0 1120
@STEXT  001  0007 0172
@STYPE  001  0006 0171    4939
@SYCNT  001  0002 1159
@TBCNT  001  0000 0160
@TBLEF  001  0010 0155    0157
@TBLIX  001  0011 0157
@TJ37B  001  0040 1257
@TYPAM  001  0002 1201
@TYPO   001  001C 1200
@UCB    001  0087 0039    3197  3206  3221  3411  3422  3564  4831  4842  4847  5111  5283
@UPARW  001  005A 0078
```

SYMBOL	LEN	VALUE	DEFN	REFERENCES
@VADDR	001	0002	0141	2314 2750 2762 2763 2764 2764 2778 2781 2783 2807 2808 2809
				2847 2850 2853 2856 2859 2862 2865 2874 2877 2880 2883 2886
@VENTA	001	0056	0113	2581 2836
@VMDDV	001	00FE	0114	
@VMFD1	001	0000	0109	
@VMFD2	001	0001	0110	
@VMRS3	001	0002	0112	
@VMTRL	001	0001	0111	
@VOLID	001	0006	0091	
@VQ	001	0001	0025	5334
@WA37B	001	00FF	1265	
@WSFIT	001	0500	0101	
@WSTBL	001	0503	0102	4904
@XR	001	0002	0014	3024* 3025 3029 3036 3052* 3058 3061 3078 3079 3079* 3083 3087
				3091* 3110* 3129* 3130* 3132 3137 3154* 3155* 3156 3164* 3166 3169
				3174 3176* 3177* 3178 3182* 3185 3185* 3215* 3216 3415 3418 3418*
				3419 3421 3424 3424* 3425 3427 3429 3546 3551 3554 3565 3565*
				3568 3568* 3574 3613* 3663* 3834 3836* 3849 3851 3853 3855 3857
				3859 3859* 3915 3919* 3924 3926 3929 3930 3932 3934 3945* 3947
				3951* 3962 3962* 3967* 4067 4070* 4103 4104 4113 4117 4122 4127
				4132 4134 4136 4138 4140 4181 4221 4234 4235 4237* 4238 4238*
				4239 4249* 4256 4267* 4269 4280* 4295 4311 4321* 4330* 4345 4345*
				4368 4374 4376 4384 4386* 4387 4389* 4394 4399 4403 4410 4412
				4414 4416 4418 4420 4430 4444* 4448 4453* 4768* 4777* 4778 4786
				4789 4795 4797 4798 4798* 4802 4804 4805 4805* 4811 4813 4823
				4824 4826 4832 4835 4836 4837 4838 4838* 4843 4845 4848 4849
				4850 4851 4851* 4852 4858 4861 4863 4869 4871 4871* 4885* 4887
				4888* 4889 4892 5245 5250* 5251 5327* 5328 5329 5329 5334 5337*
				5433* 5434 5436 5437 5437* 5441 5442 5442*
@ZERO	001	0000	0062	3574 3808 4060 4147 4294 4799 4843 4852* 4861 5056 5293 5318
				5384 5420 5421 5426
@4K	001	0010	1219	
B$ADMK	001	0001	2218	
B$ADSW	001	159D	2217	
B$ARMK	001	0001	2203	
B$ARSW	001	0A45	2202	
B$BABF	001	1D00	2008	
B$BCKT	001	1590	2130	
B$BDPL	001	19E8	2082	
B$BDSA	001	19EA	2083	
B$BINO	001	1A6A	2146	
B$BRLN	001	19F1	2081	
B$BROP	001	1AF7	2187	
B$BRVA	001	19EF	2080	
B$BRVP	001	19EE	2079	
B$BTAB	001	1996	2078	
B$CADR	001	1AF9	2188	
B$CASA	001	0000	2023	
B$CASC	001	0671	2027	
B$CASM	001	0608	2025	
B$CBAS	001	14BB	2153	
B$CBFA	001	0CBC	2108	
B$CCGT	001	0600	2033	
B$CCLS	001	0695	2039	
B$CCON	001	001F	2106	
B$CDAT	001	0600	2019	

```
SYMBOL   LEN VALUE DEFN   REFERENCES                         VER 15, MOD 00  04/07/22  PAGE  89

B$CDEF   001  0600 2020
B$CDIM   001  0673 2021
B$CDUM   001  0000 2057
B$CEND   001  0600 2055   2056
B$CEOF   001  0600 2056
B$CFOR   001  0600 2028
B$CGET   001  06A3 2036
B$CGSB   001  0690 2034
B$CGTO   001  06B3 2032
B$CIFA   001  0600 2030
B$CIFC   001  0600 2031
B$CIMG   001  0600 2045
B$CINP   001  0600 2040
B$CLTA   001  0000 2022
B$CLTC   001  0669 2026
B$CLTM   001  0600 2024
B$CMAT   001  0600 2046
B$CMGT   001  0665 2047
B$CMIN   001  06D3 2048
B$CMPR   001  069B 2051
B$CMPT   001  069B 2050
B$CMPU   001  0600 2052
B$CMRD   001  06D0 2049
B$CNXT   001  0600 2029
B$CPCT   001  0CA8 2111
B$CPRT   001  0600 2043
B$CPRU   001  0600 2044
B$CPSE   001  06E7 2053
B$CPUT   001  0600 2037
B$CPWA   001  0CA6 2182
B$CRAD   001  150D 2152
B$CRBS   001  1509 2154
B$CREA   001  06CF 2041
B$CREM   001  0000 2018
B$CRMK   001  0001 2230
B$CRSR   001  06E3 2042
B$CRST   001  06A6 2038
B$CRSW   001  0E42 2229
B$CRTN   001  06CF 2035
B$CSBF   001  0600 2005   2019  2020  2021  2024  2025  2026  2027  2028  2029  2030  2031  2032
                          2033  2034  2035  2036  2037  2038  2039  2040  2041  2042  2043  2044
                          2045  2046  2047  2048  2049  2050  2051  2052  2053  2054  2055  2058
                          2059  2060  2061  2062
B$CSCN   001  14B0 2127
B$CSMK   001  0007 2233
B$CSSW   001  14BC 2232
B$CSTP   001  06D6 2054
B$CSTR   001  14CC 2151
B$CSXA   001  2000 2011
B$CTYP   001  0A5F 2105
B$CVPD   001  0C5D 2110
B$CVPG   001  0CA5 2109
B$CWRK   001  F500 2179
B$DIST   001  0700 2071
B$DLNK   001  1B37 2177
B$DL4T   001  1A6B 2148
```

SYMBOL	LEN	VALUE	DEFN
B$DPWA	001	0E46	2183
B$DST2	001	073A	2072
B$ERMK	001	0007	2206
B$ERSW	001	0993	2205
B$FACA	001	0E53	2114
B$FAIS	001	15AC	2131
B$FAIW	001	15A0	2132
B$FCON	001	0A46	2104
B$FORT	001	1B0E	2173
B$FPWA	001	15AC	2184
B$FRMK	001	0007	2224
B$FRSW	001	16CC	2223
B$FSC1	001	0E4C	2115
B$FSC2	001	0E4D	2116
B$FSMK	001	0007	2215
B$FSSW	001	0E5C	2214
B$FSVA	001	0E4F	2117
B$FTND	001	1B0B	2175
B$FTPT	001	1B0D	2174
B$FVME	001	15A2	2136
B$FVMP	001	15A4	2137
B$FVMS	001	15A6	2138
B$FVPE	001	15A8	2133
B$FVPP	001	15AA	2134
B$FVPS	001	15AC	2135
B$GBSW	001	08AF	2208
B$GBWK	001	0001	2209
B$GETC	001	0867	2085
B$GPTR	001	0878	2087
B$GTBF	001	1E00	2009
B$IFMK	001	0007	2227
B$IFSW	001	16E5	2226
B$INVT	001	1B38	2167
B$KWMK	001	0001	2221
B$KWSW	001	159E	2220
B$LBAS	001	185E	2158
B$LBSV	001	18E7	2156
B$LDRP	001	1A00	2006
B$LINE	001	07D0	2073
B$LIST	001	1853	2140
B$LRTN	001	18EB	2157
B$LSTR	001	1862	2155
B$LTYP	001	18F2	2141
B$MATR	001	18F3	2143
B$MBMK	001	0007	2242
B$MBSW	001	1903	2241
B$MFBK	001	1B8F	2169
B$MGMK	001	0007	2239
B$MGSW	001	18FF	2238
B$MPMK	001	0007	2245
B$MPSW	001	1981	2244
B$MRMK	001	0007	2236
B$MRSW	001	0DDE	2235
B$NUMC	001	0873	2086
B$NXMK	001	0007	2212
B$NXSW	001	071D	2211

```
B$PARP   001  0A41 2094
B$PBNL   001  0A01 2100
B$PCAD   001  0A40 2095
B$PCDL   001  09D3 2099
B$PCPG   001  0A35 2098
B$PECT   001  0A44 2102
B$PERC   001  0A39 2101
B$PFAE   001  0033 2092
B$PFCL   001  009D 2093
B$PFNC   001  094E 2090
B$PFWP   001  0015 2091
B$PNBY   001  0A41 2096
B$PPWA   001  0A35 2181
B$PRM1   001  1AF3 2185
B$PTBF   001  1F00 2010
B$PUTC   001  093A 2089
B$PVAD   001  0A43 2097
B$RMRK   001  1AE6 2150
B$RTRN   001  1AF5 2186
B$SABF   001  1C00 2007
B$SCAN   001  1514 2129
B$SCAT   001  13C8 2124
B$SCON   001  001B 2107
B$SCVT   001  12E0 2122
B$SDPL   001  07DA 2075
B$SFAB   001  0E48 2119
B$SFNT   001  143C 2125
B$SLDT   001  109C 2121
B$SLVT   001  1062 2120
B$SNAT   001  131A 2123
B$SPAT   001  07E0 2076
B$SSTA   001  1BAC 2171
B$STAS   001  061B 2060
B$STIF   001  0606 2062
B$STMA   001  061B 2061
B$STML   001  0600 2059
B$STRL   001  0600 2058
B$SVRB   001  0E46 2118
B$SYMB   001  0DBC 2113
B$TCD2   001  0001 2191
B$TLTH   001  0002 2192   2193
B$TOD1   001  0000 2190
B$TOTB   001  1AF8 2193
B$TTAB   001  1AFA 2189   2193
B$TYPE   001  0739 2074
B$WORK   001  15A0 2178
B$ZDBN   001  19F2 2145
B@ABAS   001  0007 2778
B@ACD1   001  0001 2775   2776
B@ACD2   001  0003 2776   2777
B@AFLG   001  0000 2770
B@ALLA   001  005C 2595
B@AMAX   001  0005 2777   2778
B@BLNK   001  0040 2604   3926  4364
B@BLSZ   001  0100 2729   2868  2871  2874  2889  2892
B@BREQ   001  0084 2384
```

```
B@BRHI   001  0088 2385
B@BRLO   001  0082 2383
B@BRNE   001  0094 2387
B@BRNH   001  0098 2388
B@BRNL   001  0092 2386
B@CADD   001  0006 2253
B@CADF   001  0058 2294
B@CBAS   001  0003 2781
B@CBNX   001  004A 2287
B@CBRA   001  0046 2285
B@CBRC   001  0044 2284
B@CBRD   001  0048 2286
B@CBRS   001  004C 2288
B@CCLS   001  005E 2297
B@CCMC   001  0042 2283
B@CCMF   001  0040 2282
B@CCNT   001  001F 2707
B@CCSA   001  003E 2281
B@CDCA   001  006A 2303
B@CDDL   001  006C 2304
B@CDIV   001  000C 2256
B@CDMN   001  0001 2780   2781
B@CDWA   001  006E 2305
B@CEOF   001  0070 2306
B@CEOP   001  0068 2302
B@CFCI   001  0016 2261
B@CFN0   001  0012 2259
B@CFN1   001  0014 2260
B@CFOR   001  004E 2289
B@CGET   001  0052 2291
B@CHAR   001  0000 2720
B@CHLT   001  0004 2252
B@CIEX   001  00C5 2680   4374
B@CIMH   001  0066 2301
B@CINI   001  0056 2293
B@CIPI   001  00D7 2683   4376
B@CIS2   001  00E2 2686
B@CMF1   001  0018 2262
B@CMF2   001  001A 2263
B@CMF3   001  001C 2264
B@CMMA   001  006B 2615
B@CMPY   001  000A 2255
B@CMSM   001  001E 2265
B@CNEG   001  0010 2258
B@CNXT   001  0050 2290
B@COLN   001  007A 2617
B@CPMK   001  00FF 2525   2529  2533  2534  2568
B@CPRS   001  0060 2298
B@CPRU   001  0062 2299
B@CPUT   001  0054 2292
B@CPWR   001  000E 2257
B@CRSR   001  005A 2295
B@CRST   001  005C 2296
B@CSA1   001  0036 2277
B@CSA2   001  0038 2278
B@CSB1   001  003A 2279
```

```
B@CSC1   001  002A  2271
B@CSD0   001  002E  2273
B@CSD1   001  0030  2274
B@CSD2   001  0032  2275
B@CSF1   001  0022  2267
B@CSF2   001  0024  2268
B@CSTA   001  0034  2276
B@CSTC   001  0028  2270
B@CSTF   001  0020  2266
B@CSTH   001  0064  2300
B@CSTX   001  003C  2280
B@CSUB   001  0008  2254
B@CSVC   001  0002  2251
B@CTYP   001  0020  2705
B@CUSC   001  002C  2272
B@CUSF   001  0026  2269
B@CVAR   001  005B  2594    4127
B@DAMK   001  0080  2773
B@DASA   001  00FF  2534
B@DASC   001  0040  2538
B@DASM   001  0038  2536
B@DCGT   001  0050  2544
B@DCLS   001  0054  2550
B@DDAT   001  0024  2530
B@DDEF   001  0034  2531
B@DDIM   001  0004  2532
B@DDUM   001  00FF  2568
B@DEC0   001  00F0  2663    4117  4422
B@DEC1   001  00F1  2664
B@DEC2   001  00F2  2665
B@DEC3   001  00F3  2666
B@DEC4   001  00F4  2667
B@DEC5   001  00F5  2668
B@DEC6   001  00F6  2669
B@DEC7   001  00F7  2670
B@DEC8   001  00F8  2671
B@DEC9   001  00F9  2672
B@DEND   001  0058  2566    2567
B@DEOF   001  0058  2567
B@DFOR   001  0028  2539
B@DGET   001  0040  2547
B@DGSB   001  0020  2545
B@DGTO   001  0044  2543
B@DIFA   001  0048  2541
B@DIFC   001  004C  2542
B@DIGS   001  007B  2597
B@DIMG   001  003C  2556
B@DINP   001  0000  2551
B@DIVD   001  0061  2614
B@DLTA   001  00FF  2533
B@DLTC   001  0040  2537
B@DLTM   001  0038  2535
B@DL01   001  0001  2848    2851
B@DL02   001  0003  2851    2854
B@DL03   001  0005  2854    2857
B@DL04   001  0007  2857    2860
```

```
B@DL05  001  0009 2860    2863
B@DL06  001  000B 2863    2866
B@DL07  001  0045 2866    2869
B@DL08  001  0145 2869    2872
B@DL09  001  0245 2872    2875
B@DL10  001  0289 2875    2878
B@DL11  001  02C3 2878    2881
B@DL12  001  02FD 2881    2884
B@DL13  001  0337 2884    2887
B@DL14  001  0371 2887    2890
B@DL15  001  0471 2890    2893
B@DL16  001  0507 2893
B@DMAT  001  0008 2557
B@DMGT  001  0044 2558
B@DMIN  001  0038 2559
B@DMPR  001  0048 2562
B@DMPT  001  004C 2561
B@DMPU  001  0054 2563
B@DMRD  001  003C 2560
B@DNXT  001  0044 2540
B@DPNT  001  004B 2605    4424
B@DPRT  001  002C 2554
B@DPRU  001  0030 2555
B@DPSE  001  0050 2564
B@DPUT  001  0040 2548
B@DREA  001  000C 2552
B@DREM  001  00FF 2529
B@DRSR  001  005C 2553
B@DRST  001  0050 2549
B@DRTN  001  005C 2546
B@DSCY  001  0004 2521
B@DSIF  001  001C 2570
B@DSLT  001  0010 2569
B@DSML  001  0010 2571
B@DSNS  001  0018 2523
B@DSS1  001  0000 2522
B@DSTP  001  0054 2565
B@DTBN  001  0010 2587
B@DTB1  001  0050 2586
B@DTCY  001  0009 2583
B@DTSN  001  0010 2585
B@DTS1  001  0040 2584
B@DTYP  001  0040 2699
B@DVCY  001  0007 2580
B@DVC1  001  0056 2581
B@DWCY  001  0005 2577
B@DWT1  001  0003 2578
B@D1MK  001  0080 2771
B@D2MK  001  00C0 2772
B@EOST  001  001E 2593    3834
B@EQUL  001  007E 2619    3924
B@EXPC  001  00C5 2596    4295  4420
B@FOFL  001  005C 2598
B@FVAD  001  0001 2783
B@GETC  001  0001 2722
B@GETE  001  00FF 2723
```

```
B@GETS  001  0000 2721
B@GRTR  001  006E 2616
B@ICON  001  0050 2678    4368
B@LADD  001  0001 2322
B@LADF  001  0002 2363
B@LADV  001  0008 2807    2828
B@LBIN  001  0002 2732    2733   2739
B@LBNX  001  0003 2356
B@LBRA  001  0003 2354
B@LBRC  001  0004 2353
B@LBRD  001  0003 2355
B@LBRS  001  0001 2357
B@LCCA  001  0004 2763
B@LCCC  001  0001 2315    2353
B@LCDV  001  0004 2808    2829
B@LCER  001  0001 2313    2377
B@LCFN  001  0004 2764
B@LCLN  001  0002 2318    2369   2370   2377
B@LCLS  001  0001 2366
B@LCMC  001  0001 2352
B@LCMF  001  0001 2351
B@LCNA  001  0006 2762
B@LCNN  001  0001 2316    2341   2350   2362   2374
B@LCOP  001  0001 2312    2320   2321   2322   2323   2324   2325   2326   2327   2328   2329   2330   2331
                          2332   2333   2334   2335   2336   2337   2338   2339   2340   2341   2342   2343
                          2344   2345   2346   2347   2348   2349   2350   2351   2352   2353   2354   2355
                          2356   2357   2358   2359   2360   2361   2362   2363   2364   2365   2366   2367
                          2368   2369   2370   2371   2372   2373   2374   2375
B@LCRV  001  0013 2806    2826
B@LCSA  001  0002 2350
B@LCVA  001  0002 2314    2328   2329   2330   2331   2332   2333   2334   2335   2336   2337   2339   2340
                          2342   2343   2344   2345   2346   2347   2348   2353   2354   2355   2356   2358
                          2359   2360   2372   2373
B@LCXX  001  0001 2317    2349   2361   2363   2367   2368
B@LDAT  001  0004 2476    4592
B@LDCA  001  0003 2372
B@LDDL  001  0003 2373
B@LDDM  001  0004 2736
B@LDEF  001  0003 2477    4595
B@LDIM  001  0003 2478    4598
B@LDIN  001  0004 2735    2736   2737
B@LDIV  001  0001 2325
B@LDMN  001  0002 2733    2762   2763   2775   2776   2777   2780   2807   2808
B@LDSN  001  0004 2737
B@LDWA  001  0002 2374
B@LELP  001  0010 2805
B@LEND  001  0003 2505    4700
B@LEOF  001  0001 2375
B@LEOP  001  0001 2371
B@LERC  001  0003 2377
B@LESP  001  0008 2804
B@LESS  001  004C 2606
B@LET$  001  005B 2626
B@LET#  001  007B 2627
B@LET@  001  007C 2628
B@LETA  001  00C1 2630
```

```
B@LETB  001  00C2 2632
B@LETC  001  00C3 2633
B@LETD  001  00C4 2634
B@LETE  001  00C5 2635
B@LETF  001  00C6 2636
B@LETG  001  00C7 2637
B@LETH  001  00C8 2638
B@LETI  001  00C9 2639
B@LETJ  001  00D1 2640
B@LETK  001  00D2 2641
B@LETL  001  00D3 2642
B@LETM  001  00D4 2643
B@LETN  001  00D5 2644
B@LETO  001  00D6 2645
B@LETP  001  00D7 2646
B@LETQ  001  00D8 2647
B@LETR  001  00D9 2648
B@LETS  001  00E2 2649
B@LETT  001  00E3 2650
B@LETU  001  00E4 2651
B@LETV  001  00E5 2652
B@LETW  001  00E6 2653
B@LETX  001  00E7 2654
B@LETY  001  00E8 2655
B@LETZ  001  00E9 2656
B@LEXP  001  0008 2695
B@LFCI  001  0003 2330
B@LFNA  001  0002 2809   2830
B@LFN0  001  0003 2328
B@LFN1  001  0003 2329
B@LFOR  001  0003 2358
B@LFRT  001  0004 2750   2751
B@LGET  001  0003 2360
B@LGSB  001  0005 2484   4637
B@LGTO  001  0004 2483   4631  4634
B@LHLT  001  0001 2321
B@LIEX  001  0002 2681
B@LIFN  001  0003 2744   3938  3940  3956  3958  3960  4239  4250  4262  4527  4528  4533  4538
                        4539  4540  4541  4542  4543  4544  4545  4546  4547  4548  4549  4550
                        4551  4552  4553  4554  4555  4556  4557  4558  4559  4563  4564  4569
                        4570  4571
B@LILP  001  0009 2803   2821  2822  2823
B@LIMG  001  0001 2495   4670
B@LIMH  001  0003 2370
B@LINI  001  0002 2362
B@LINP  001  0005 2490   4655
B@LIPI  001  0003 2684
B@LISP  001  0005 2802   2810  2816  2817  2818
B@LIS2  001  0005 2687
B@LIVT  001  0001 2760
B@LKCL  001  0005 2489   4652
B@LKFR  001  0003 2480   4619
B@LKGT  001  0003 2486   4643
B@LKIF  001  0002 2482   4625  4628  4721
B@LKON  001  0002 2515   3884
B@LKPT  001  0003 2487   4646
```

SYMBOL	LEN	VALUE	DEFN	REFERENCES				
B@LKPU	001	000A	2494	4667				
B@LKRR	001	0007	2492	4661				
B@LKRT	001	0005	2488	4649				
B@LKTO	001	0002	2509					
B@LLET	001	0003	2479	4601	4607	4613	4709	4712
B@LL01	001	0002	2847	2848				
B@LL02	001	0002	2850	2851				
B@LL03	001	0002	2853	2854				
B@LL04	001	0002	2856	2857				
B@LL05	001	0002	2859	2860				
B@LL06	001	0002	2862	2863				
B@LL07	001	003A	2865	2866				
B@LL08	001	0100	2868	2869				
B@LL09	001	0100	2871	2872				
B@LL10	001	0044	2874	2875				
B@LL11	001	003A	2877	2878				
B@LL12	001	003A	2880	2881				
B@LL13	001	003A	2883	2884				
B@LL14	001	003A	2886	2887				
B@LL15	001	0100	2889	2890				
B@LL16	001	0096	2892	2893				
B@LMAT	001	0003	2496	4673				
B@LMF1	001	0003	2331					
B@LMF2	001	0003	2332					
B@LMF3	001	0003	2333					
B@LMGT	001	0006	2497	4676				
B@LMIN	001	0008	2498	4679				
B@LMPR	001	0008	2501	4688				
B@LMPT	001	0006	2500	4685				
B@LMPU	001	000D	2502	4691				
B@LMPY	001	0001	2324					
B@LMRD	001	0007	2499	4682				
B@LMSM	001	0003	2334					
B@LNEG	001	0001	2327					
B@LNEX	001	0004	2481	4622				
B@LNXT	001	0003	2359					
B@LPAR	001	004D	2607	4122	4181	4256	4311	4399
B@LPRS	001	0002	2367					
B@LPRT	001	0005	2493	4664				
B@LPRU	001	0002	2368					
B@LPSE	001	0005	2503	4694				
B@LPUT	001	0002	2361					
B@LPWR	001	0001	2326					
B@LREA	001	0004	2491	4658				
B@LREM	001	0003	2475	4589				
B@LRSR	001	0001	2364					
B@LRST	001	0001	2365					
B@LRTN	001	0006	2485	4640				
B@LSA1	001	0003	2346					
B@LSA2	001	0003	2347					
B@LSB1	001	0003	2348					
B@LSC1	001	0003	2340					
B@LSDF	001	0004	2730					
B@LSD0	001	0003	2342					
B@LSD1	001	0003	2343					
B@LSD2	001	0003	2344					

```
B@LSF1  001  0003 2336
B@LSF2  001  0003 2337
B@LSKW  001  0002 2746
B@LSNO  001  0002 2739
B@LSPT  001  0003 2754   2757
B@LSTA  001  0003 2345
B@LSTC  001  0003 2339
B@LSTE  001  0004 2510
B@LSTF  001  0003 2335
B@LSTH  001  0003 2369
B@LSTP  001  0004 2504   4697
B@LSTX  001  0002 2349
B@LSUB  001  0001 2323
B@LSVC  001  0001 2320
B@LTHN  001  0004 2511
B@LTYP  001  0001 2740
B@LUFN  001  0002 2747   4226
B@LUSC  001  0002 2341
B@LUSF  001  0001 2338
B@LVPG  001  0100 2834   2837
B@MINS  001  0060 2613
B@MULT  001  005C 2610
B@NAAR  001  001D 2798   2828   2880
B@NCAR  001  001D 2799   2829   2883
B@NCRV  001  001D 2797   2826   2877
B@NDGT  001  000A 2790   2796
B@NEQL  001  007F 2620
B@NFRT  001  000A 2749   2751
B@NICN  001  0006 2792   2794
B@NIEL  001  0007 2794   2810   2816   2821
B@NIFN  001  0018 2743
B@NIVR  001  0001 2793   2794
B@NIVT  001  0057 2759
B@NLDV  001  0122 2796   2818   2823   2874
B@NLRV  001  001D 2795   2817   2822   2865
B@NLTR  001  001D 2789   2795   2796   2797   2798   2799   2800
B@NSKW  001  0004 2745
B@NSPT  001  0028 2753
B@NUFN  001  001D 2800   2830   2886
B@NVPG  001  0100 2833   2837
B@NXHI  001  00E3 2714
B@NXLO  001  001E 2713
B@NXZR  001  0080 2712   2713   2714
B@PLUS  001  004E 2608
B@POWR  001  005A 2609
B@PREC  001  0020 2701
B@PROD  001  0023 2810
B@PRPL  001  0002 2397
B@PRPN  001  0001 2396
B@PRPR  001  0004 2399
B@PRPS  001  0003 2398
B@PRRC  001  0007 2402
B@PRRL  001  0008 2403
B@PRSL  001  0005 2400
B@PRSS  001  0006 2401
B@PTAB  001  0000 2755
```

```
B@PTAD   001  0001 2756
B@PTSA   001  0002 2757
B@PUD1   001  0006 2413
B@PUD2   001  0007 2414
B@PUI0   001  0001 2407
B@PUI1   001  0004 2408
B@PUI2   001  0005 2409
B@PUNL   001  0002 2411
B@PUNS   001  0003 2412
B@PURE   001  0020 2417
B@PUTM   001  0010 2416
B@RPAR   001  005D 2611   4403
B@SADV   001  00E8 2828   2831
B@SAVL   001  0B76 2824   2841
B@SAVS   001  065E 2819   2840
B@SCDV   001  0074 2829   2831
B@SCLN   001  005E 2612
B@SCRV   001  0227 2826   2840  2841
B@SDMK   001  0080 2741
B@SEXP   001  0004 2694
B@SFAT   001  0196 2831   2840  2841  2892
B@SFNA   001  003A 2830   2831
B@SFRT   001  0028 2751
B@SIEL   001  003F 2821   2824
B@SIES   001  0023 2816   2819
B@SIGN   001  0010 2703
B@SLDL   001  0A32 2823   2824
B@SLDS   001  05AA 2818   2819
B@SLVL   001  0105 2822   2824
B@SLVS   001  0091 2817   2819
B@SQUO   001  007D 2618   4384  4387
B@STAT   001  0000 2693
B@TASA   001  0012 2428
B@TASC   001  001E 2434
B@TASM   001  0018 2430
B@TASS   001  007B 2435
B@TCGT   001  0030 2443
B@TCLS   001  0042 2449
B@TDAT   001  0006 2424
B@TDEF   001  0009 2425
B@TDIM   001  000C 2426
B@TDUM   001  0078 2467   3807
B@TEND   001  0072 2465
B@TEOF   001  0075 2466
B@TFOR   001  0021 2437
B@TGET   001  0039 2446
B@TGSB   001  0033 2444
B@TGTO   001  002D 2442
B@TIFA   001  0027 2439
B@TIFC   001  002A 2440
B@TIFS   001  007D 2441
B@TIMG   001  0054 2455
B@TINP   001  0045 2450
B@TLTA   001  000F 2427
B@TLTC   001  001B 2431
B@TLTM   001  0015 2429
```

```
B@TLTS  001  0079 2432
B@TMAS  001  007C 2436
B@TMAT  001  0057 2456
B@TMGT  001  005A 2457
B@TMIN  001  005D 2458
B@TMLS  001  007A 2433
B@TMPR  001  0066 2461
B@TMPT  001  0063 2460
B@TMPU  001  0069 2462
B@TMRD  001  0060 2459
B@TNXT  001  0024 2438
B@TPRT  001  004E 2453
B@TPRU  001  0051 2454
B@TPSE  001  006C 2463
B@TPUT  001  003C 2447
B@TRAC  001  0080 2697
B@TREA  001  0048 2451
B@TREM  001  0003 2423
B@TRSR  001  004B 2452
B@TRST  001  003F 2448
B@TRTN  001  0036 2445
B@TSTP  001  006F 2464
B@VMC1  001  0056 2836
B@VMLB  001  F0CD 2841
B@VMSB  001  F5E5 2840
B@VMSZ  001  0000 2837    2839   2840   2841
B@VMTB  001  0000 2839
B@ZNEG  001  00D0 2710
B@ZPOS  001  00F0 2709
DCRCNT  001  1863 5348    5349
DLIBUF  001  1C8C 5479    5327   5358   5433
DLPBLN  001  00F4 5449    5328*  5329   5329   5329*  5431
DLPBSD  001  1777 5258    5345   5346   5347
DLPBSE  004  1785 5269    5241   5244   5424   5425
DLPBS2  001  1868 5448    5375   5377   5429   5430
DLPCNT  001  1863 5349    5293*  5294   5303*  5350
DLPCRT  001  001B 5347    3582   3593   3599   3639
DLPEXT  002  1795 5274    5252*  5253*  5263
DLPK13  001  1867 5354    5278   5282
DLPLIN  001  1866 5353    5286   5299
DLPLPC  002  1865 5352    5286*  5287*  5299*  5300*
DLPMAX  001  000D 5355    5294
DLPMPR  001  0085 5345    3585   3588   3596   3642
DLPNDX  001  1870 5361    5423
DLPNPT  001  17FC 5308    5262   5267   5345
DLPNXT  001  1876 5365    5383*  5390*  5396   5400   5402   5445
DLPONE  002  1872 5362    5246   5248   5287   5300   5303   5331   5412   5438   5439   5443
DLPPNT  001  0001 5369    5409
DLPPRL  001  18D2 5408    5392
DLPPRT  001  187A 5376    5320   5446
DLPREM  001  1877 5366    5431*  5432*  5443*
DLPRES  001  1873 5363    5384*  5387*  5388*  5389   5390   5426   5432   5439*
DLPRNT  001  174C 5242    3222   3233
DLPRTN  001  1878 5367    5322   5324
DLPSPI  001  1777 5260    5346
DLPSPT  001  0000 5346    5257
```

```
DLPTIF  001 1792 5271   5347
DLPTYP  001 1776 5255   3582   3585   3588*  3593   3596   3599*  3639   3642   5256
DLPWK1  001 1868 5356   5319   5323*  5325   5330   5385   5387   5389*  5396   5399*  5402*  5410*  5411*
                        5412*  5416*  5419   5445*  5448
DLPWK2  001 186C 5359   5251*  5265   5275   5276   5312   5316   5318*  5323   5324*  5333   5383
DLPWTH  002 1875 5364   5381*  5382*  5385   5388   5399   5400   5435
DLP100  004 1764 5250   5247*
DLP120  004 1782 5264   5263*  5269
DLP140  003 179E 5278   5289
DLP160  003 17A8 5281   5283*  5285*
DLP180  003 17B4 5285   5281
DLP200  004 17B7 5286   5284
DLP220  004 17BB 5287   5288
DLP240  004 17C5 5290   5280
DLP260  003 17D3 5294   5291
DLP280  003 17DD 5298   5296
DLP300  004 17E4 5300   5301
DLP320  004 17EE 5303   5298
DLP340  003 17F2 5304   5302
DLP360  004 17F5 5305   5277
DLP380  004 1803 5311   5326
DLP400  003 1812 5316   5310
DLP420  003 181B 5319   5317
DLP440  004 1827 5323   5427
DLP460  004 184F 5334   5330*  5331*  5332   5332*
DLP480  004 1853 5336   5243*  5268   5304   5306   5315
DLP500  004 1857 5337   5245*
DLP520  004 185F 5339   5249*
DLP540  006 18B0 5391   5386
DLP560  003 18E4 5416   5397   5401   5404
DLP580  005 191B 5436   5434*  5435*  5438*  5440
DLP600  003 192E 5441   5444
DL4CYL  001 1706 5089   5061*
DL4C01  002 170C 5097   5047   5049   5061
DL4C05  002 170E 5098   5053
DL4C24  003 16DD 5100   5074
DL4C48  003 16CA 5102   5068   5109   5115
DL4C96  003 16B9 5099   5062
DL4DPL  006 170A 5088   5054*
DL4EFD  001 0001 5095   5067   5113
DL4END  001 174C 5126
DL4ETB  001 0080 5096   5073
DL4E01  001 0001 5094   5069
DL4E24  001 0018 5093   5071
DL4E48  001 0030 5092   5065   5107
DL4E96  001 0060 5091   5059
DL4ICS  001 1690 5042   4894
DL4LST  001 1705 5087   5080   5089   5090   5101   5119*
DL4SAV  005 16A7 5125   5112*  5115*  5118
DL4SCD  001 1707 5090   5059   5062*  5065   5068*  5071   5074*  5075   5075*  5076   5076*  5077*  5106
                        5112   5118*  5120*
DL4SCT  001 1708 5101   5069   5104   5110*  5119   5120   5121*
DL4SPT  004 170F 5105   5070
DL4WRK  005 16A8 5124   5104*  5106*  5107   5109*  5110   5121
DL4010  001 1694 5045   5043   5046
DL4020  005 16A4 5052   5048*  5124   5125
```

```
DL4030   005  16AD 5054    5052* 5053*
DL4035   003  16B2 5056    5122
DL4040   003  16B8 5059    5063  5099
DL4050   003  16C9 5065    5060  5102
DL4060   003  16D6 5069    5066
DL4070   003  16DC 5071    5100  5108  5114  5116
DL4080   004  16E9 5075    5072
DL4100   003  16F1 5077    5056* 5067* 5073* 5113
DL4200   003  16FA 5082    5057* 5111*
DL4500   004  170F 5104    5105
DL4600   004  1739 5118    5082
DL4900   004  16FD 5084    5044*
DL4920   004  1701 5085    5050*
GRABIT   001  14F2 4760    3043  3047
GRABOA   002  167B 4925    4854  4867  4872
GRABSE   004  15D0 4951    4759  4762
GRACCA   002  166C 4902
GRACFN   001  166B 4900
GRACPL   001  166B 4899
GRACSC   001  166E 4905    4781*
GRAEBS   001  00FF 4933    4780  4896
GRAEDB   001  0002 4919    4788  4891
GRAEDC   001  0001 4950
GRAEDL   001  0006 4938    4805  4823
GRAEDS   001  0005 4952    4886
GRAEDT   001  0007 4939    4795  4824  4826
GRAEET   001  0075 4941    4795  4826
GRAEFG   001  0004 4932    4817
GRAEFI   001  0000 4928    4764
GRAEFR   001  0001 4930    3046  4769  4815
GRAEFS   001  0002 4931    4771
GRAEFW   001  0003 4929
GRAELK   001  0000 4935    4786  4789  4889  4892
GRAELL   001  0002 4940    4823
GRAELN   001  0000 4936    4786  4889
GRAELP   001  0007 4946    4838
GRAELS   001  0004 4947    4851
GRAEMR   001  001B 4948    4858
GRAENC   001  0001 4949    4858  4863* 4869  4871
GRAERR   004  1684 4957    4784* 4800  4812  4816
GRAESC   001  0001 4934
GRAES0   001  0001 4942    4802  4811
GRAES1   001  0002 4943    4797  4798  4835  4836* 4837  4848  4849* 4850
GRAES2   001  0003 4944    4813  4832  4845
GRAETP   001  0002 4945    4813
GRAEW2   001  0006 4953
GRAEXA   001  0001 4937    4938  4939  4942  4943  4944
GRANCA   002  1676 4913    4778* 4785* 4886  4887*
GRANDA   002  1673 4909    4779* 4788* 4789* 4790* 4891* 4892* 4893*
GRANPB   002  167B 4918    4790  4893  4924  4925  4926
GRANPL   001  1671 4907    4895
GRANXC   002  167B 4926
GRAONE   002  167B 4924    4863
GRAPSG   002  1680 4922    4836
GRASAR   004  1573 4809    4763*
GRASBR   004  156F 4807    4761*
```

```
                              CROSS REFERENCE

SYMBOL   LEN VALUE DEFN    REFERENCES                      VER 15, MOD 00  04/07/22  PAGE 103

GRASEG   001  1683 4927   4837*  4850*  4872*
GRASIZ   001  167C 4920   4780*  4797*  4799   4835*  4848*  4896*
GRASSG   002  1682 4923   4849
GRASSZ   002  1679 4917   4785
GRASVC   003  15F4 4853   4843*
GRATND   005  160E 4862   4860*  4865   4867*
GRATXT   002  167E 4921   4825
GRA020   004  1504 4768   4804*
GRA100   003  1517 4777   4765
GRA140   003  1535 4786
GRA150   004  1542 4790   4787
GRA200   003  1549 4795   4772
GRA210   004  154F 4797   4773   4819
GRA220   003  1556 4799   4840   4842
GRA230   004  1565 4804   4796   4814   4818   4829
GRA240   004  156C 4806   4807
GRA245   004  1570 4808   4809
GRA250   003  1574 4810   4801   4803
GRA260   003  1577 4811   4791
GRA300   005  1595 4823   4770
GRA303   003  15B2 4831   4827
GRA305   004  15BE 4835   4833
GRA310   004  15D0 4840   4831*  4834*  4841   4847*  4873   4951
GRA313   004  15E4 4848   4846
GRA315   003  15F3 4852   4853
GRA316   004  15F6 4854   4874
GRA317   001  15FA 4855   4839
GRA320   005  160B 4861   4862   4868
GRA330   004  161E 4867   4864
GRA350   005  1625 4869   4857   4859   4870
GRA360   003  162A 4871   4866
GRA5SA   004  166A 4898   4878*
GRA500   003  1637 4878   4810   4844
GRA600   001  1640 4881
GRA620   004  165A 4893   4890
GRA640   004  165E 4894
GRA660   003  1664 4896
GRA680   004  1667 4897   4898
GRBFRA   002  1670 4906   4777   4885   4886*  4888
GRBFR1   001  1E00 4727   4906
GRLINE   001  1B97 5477   4823*  5478
GRSCTR   001  1674 4910   4781
GRSRDA   002  166D 4901   4779   4902
GRTEND   005  1628 4870   4825*  4854*  4860   4865*
GRTEXT   001  1B98 5478   3049   3052   4828*  4921   5479   5480
GRTYPE   001  1B96 5476   3112   3812   4824*  5477
GRWHAT   001  1677 4914   3046*  4764   4769   4771   4815   4817
KSYAAA   002  1947 5454   3097   3164
KSYACA   002  194B 5456   3105
KSYACV   002  1949 5455   3103
KSYALD   002  1945 5453   3081
KSYALP   001  0E67 3258   3054   3154   3176   3263
KSYASL   002  1943 5452   3072   3126
KSYAST   001  005C 3282   3140   3172
KSYBFR   001  1B98 5480   3122*  3123   3123*  3125   3209   3215   3230   3235*  3236   3236*  3237   3293
KSYBLA   001  0040 3285   3122   3209   3230   3235
```

```
KSYCAC   001 1B79 5475    3018* 3019  3019* 5456  5476
KSYCHR   001 0E8F 3274    3135  3156* 3275
KSYCLR   001 0000 3310    3018
KSYCNT   001 0001 3315    3045* 3218*
KSYCVC   001 1B5C 5474    5455  5475
KSYDGT   001 0E8E 3273    3127* 3142  3145  3148* 3149
KSYDIG   002 0E85 3261    3081* 3086* 3090
KSYDIS   001 0080 3305    3112  3115  3137  3169
KSYDSH   001 0060 3309    3025
KSYDSI   001 0E8C 3269    3271
KSYDSP   002 0E8D 3270    3130  3152* 3155  3158  3165* 3177  3186* 3188  3216* 3217
KSYDUM   001 F199 3263    3264
KSYFBR   002 0E89 3264    3068
KSYLDC   001 1A1D 5472    5453  5473
KSYLNG   001 0195 3308    3312
KSYLN1   001 0001 3306    3061  3087  3109  3135  3145  3148  3152  3156  3178  3186  3218
KSYLN2   001 0002 3307
KSYLRY   001 0003 3281    3180  3203
KSYLVC   001 1A00 5471    3020  3020* 5452  5472
KSYMBL   001 05FF 3003
KSYMIN   002 0E60 3253
KSYNAC   001 1B3F 5473    5454  5474
KSYNXT   002 0E62 3254    3086  3147
KSYONE   001 0E90 3277    3087  3148  3152  3186
KSYPPL   001 0E95 3290    3045* 3218* 3234
KSYPRN   004 0E1F 3227    3146  3183  3219
KSYRAY   004 0E94 3278    3180
KSYREF   001 0001 3304    3107  3132  3166
KSYREG   001 0002 3303
KSYREM   001 0095 3312    3020  3020  3020*
KSYSCT   001 0100 3311    3019  3312
KSYTAB   002 0E87 3262    3126* 3129  3147*
KSYTB0   001 001C 3313    3018* 3019  3019* 3158  3188
KSYTHR   002 0E66 3256    3195  3204
KSYTMI   001 0E8A 3265    3267
KSYTMP   002 0E8B 3266    3109* 3110  3214* 3217* 3218
KSYXEF   001 00EF 3284    3127  3142
KSYXFO   001 00F0 3302    3083
KSYXF9   001 00F9 3283    3149
KSYZER   002 0E64 3255
KSY000   001 0000 3260    3025  3029  3036  3058  3061  3061  3083  3087* 3107* 3115* 3132  3137
                         3140* 3156  3165  3166  3169  3172* 3178  3194
KSY001   001 0001 3279    3065  3079  3135* 3178* 3185  3209
KSY002   001 0002 3296    3145* 3193
KSY003   001 0003 3297
KSY004   001 0004 3298    3180*
KSY005   001 0005 3299    3202
KSY006   001 0006 3300
KSY007   001 0007 3301    3229
KSY057   001 0039 3286    3230
KSY062   001 003E 3287    3122* 3123  3123  3123* 3235* 3236  3236  3236*
KSY063   001 003F 3280    3045
KSY100   004 0C3F 3042    3030
KSY150   004 0C4F 3047    3059
KSY200   004 0C5B 3052
KSY250   004 0C5F 3054    3117
```

```
KSY300   004  0C6E 3061   3066
KSY350   004  0C7C 3068   3063
KSY400   004  0C8E 3075   3070
KSY450   003  0CA6 3083   3088
KSY500   004  0CBB 3090   3084
KSY525   004  0CBF 3091   3078*
KSY550   004  0CC6 3094   3076
KSY600   004  0CD4 3100   3095
KSY650   004  0CE2 3105   3101
KSY700   003  0CE6 3107   3073  3092  3098  3104
KSY750   004  0CFD 3117   3113
KSY800   004  0D01 3122   3050
KSY805   006  0D0F 3126   3159
KSY810   004  0D19 3129   3150
KSY820   004  0D35 3142   3138
KSY830   004  0D41 3146   3143
KSY840   006  0D45 3147   3133
KSY845   004  0D78 3165   3198  3207
KSY847   003  0D7C 3166   3189
KSY850   003  0D88 3172
KSY855   004  0D8B 3174   3170
KSY858   005  0D9B 3180   3193* 3194* 3195* 3202* 3203* 3204*
KSY860   004  0DA0 3182   3174*
KSY865   003  0DA8 3185   3167
KSY870   003  0DB9 3191   3197*
KSY875   003  0DD2 3200   3191  3206*
KSY880   004  0DEB 3209   3200
KSY885   003  0E12 3221   3210  3228*
KSY887   003  0E2E 3231   3212*
KSY899   004  0E45 3238   3227* 3231
KSY900   004  0E49 3240   3026
KSY925   004  0E50 3243   3221
KSY950   004  0E57 3246   3037
KSY999   004  0E5B 3248   3034  3241  3244
SCACNT   002  0ED9 3439   3429* 3430*
SCACOF   001  0087 3411
SCACOM   001  0001 3410   3547
SCAINC   001  0001 3409   3418  3424
SCAMMA   003  0EB6 3433   3547*
SCANIT   001  0E99 3413   3028  3080  3570
SCASVE   002  0ED7 3438   3415* 3430
SCASV1   001  0ED6 3437
SCA100   003  0EA8 3418   3420
SCA200   003  0EAB 3419   3417
SCA250   003  0EB5 3422   3433
SCA300   003  0EB8 3424   3426
SCA400   004  0EC8 3429   3422
SCA500   004  0ED2 3432   3414* 3428
SCKCCR   003  0F70 3628   3551
SCKCL0   006  0FC7 3670
SCKCL1   004  0FCD 3671   3670* 3672*
SCKCMP   007  0F77 3629   3554
SCKDEV   001  0F7E 3635   3042  3663
SCKEND   001  0FDF 3677
SCKERR   001  0469 3314   3664
SCKOUT   001  0EDA 3544   3032
```

```
                               CROSS REFERENCE

SYMBOL  LEN VALUE DEFN   REFERENCES                        VER 15, MOD 00  04/07/22  PAGE 106

SCK001  001  0003 3623   3551   3551   3565   3628
SCK002  001  0007 3624   3554   3554   3568   3629
SCK003  002  0F79 3630   3559
SCK004  002  0F7B 3631   3600
SCK005  002  0F7D 3632   3614
SCK100  004  0EFD 3564   3552
SCK150  003  0F07 3568   3555
SCK200  004  0F0A 3570   3566
SCK300  003  0F1B 3577   3564*  3572   3618*
SCK350  004  0F33 3593   3577
SCK400  004  0F45 3600   3589
SCK410  004  0F4C 3605   3575
SCK420  004  0F53 3608   3583   3597
SCK430  004  0F5A 3611   3586   3594
SCK440  004  0F5E 3613   3546*  3606   3609
SCK450  004  0F66 3618   3560   3601
SCK460  004  0F6A 3619   3545*
SCK475  004  0FA2 3651   3640
SCK500  004  0FB7 3661   3652
SCK550  004  0FBB 3663   3649   3659
SCK600  004  0FC3 3668   3655
SCK650  004  0FDB 3676   3636*  3643   3646
SVABRT  001  146B 4587   3820
SVABSW  001  13F6 4492   3847   3862*  3876   3881*  3903   3909*  3981   3983*  4002   4007*  4023   4029*
                         4047   4052*  4084*
SVACAC  001  0002 4488   4195
SVACON  003  1464 4569   3956
SVACVC  001  0004 4486   3100   4189
SVADET  003  1411 4527   4262
SVADIS  001  0080 4480   3813
SVADP0  001  0000 4467   3824   3834   3849   3851   3853   3855   3857   3924   3926   3930   3932   3934
                         4103   4113   4117   4122   4127   4132   4134   4136   4138   4140   4181   4221
                         4239   4256   4384   4387   4394   4410   4412   4414   4416   4418   4420   4430
                         4448
SVADSW  001  13FB 4504   4364*  4422   4424   4430*
SVAFIL  001  1406 4516   4244
SVAFNC  002  1416 4532   4226
SVAFTD  001  0041 4479   4236
SVAIDN  003  1467 4570   3958
SVAIFT  001  141A 4537   4237
SVAINV  003  145E 4563   3938
SVAIO1  002  1405 4515   3836   3945   4156   4176   4267   4309   4349   4386   4389   4401   4405   4444
                         4453   4454
SVAKGO  002  140E 4523   4215
SVAKLN  001  0002 4474   4207   4209   4213   4215
SVAKST  002  140A 4521   4209
SVAKTH  002  140C 4522   4213
SVAKTO  002  1408 4520   4207
SVAKWL  001  13F8 4498   3822*  3884*  4349*  4499
SVALDC  001  0010 4485   3075   4158
SVALNG  001  1401 4508   3109   3916*  3948*  4068*  4108*  4156*  4176*  4270*  4281*  4454*
SVALSA  003  13FF 4506   3938   3940   3956   3958   3960   4575   4576   4577
SVALS1  003  13FD 4575   3930*  4103*
SVALS2  003  13FE 4576   3932*  4113*  4207   4209   4213   4215   4226
SVALS3  003  13FF 4577   3934*  4221*  4228   4239   4250   4262
SVALVC  001  0001 4484   3069   4149   4279
```

```
SYMBOL  LEN VALUE DEFN    REFERENCES                          VER 15, MOD 00  04/07/22  PAGE 107

SVAL15  001  000F 4477    4094
SVAMAG  001  1402 4509    3807* 3815* 3817
SVANAC  001  0008 4487    3094  3918  3950  4071  4165  4272
SVAOFF  001  0000 4469    3847  3876  3903  3910  3981  4002  4008  4023  4047  4053  4084
SVAONN  001  0001 4470    3862  3881  3909  3983  4007  4029  4052
SVAOTD  001  0000 4468    3822
SVAPCT  001  13FC 4505    3910* 4008* 4053* 4060  4094* 4147  4309* 4401* 4405*
SVARAB  001  0FDF 3802    3056
SVARND  003  1414 4528    4250
SVASSS  001  13F7 4495    4211* 4292  4294* 4496
SVASTC  002  13FA 4501    3812* 3813* 3815  3819* 3821
SVASTR  003  1419 4533    4228
SVATRN  003  1461 4564    3940
SVAVL1  001  0001 4472    3916  3948  4068  4108  4270  4281
SVAVL2  001  0002 4475
SVAVL3  001  0003 4476
SVAVTC  001  1400 4507    3069  3075  3094  3100  3918* 3950* 4071* 4149* 4158* 4165* 4189* 4195*
                         4272* 4279*
SVAZER  003  146A 4571    3960
SVAZRO  001  1403 4510    3808* 3817* 3818  3818* 3819
SVA020  004  0FF7 3813
SVA030  004  100E 3820    3816
SVA050  003  1021 3834    3837  4590  4593  4638  4641  4662  4671  4695  4698  4701  4704  4707
SVA070  004  1030 3846    4605  4611  4617  4716  4719
SVA075  003  103A 3849    3860
SVA080  003  1046 3853    3850
SVA085  004  105F 3861    3848  3852  3854  3856  3858
SVA090  004  106A 3872    4635
SVA095  003  106E 3876
SVA1TD  001  0001 4471
SVA100  003  1074 3881
SVA110  004  1086 3889    3877
SVA120  004  108E 3899    4674
SVA125  003  1092 3903
SVA130  004  1098 3908
SVA140  004  10A2 3914
SVA144  003  10AD 3917
SVA148  004  10B3 3919    3915*
SVA150  003  10BB 3924    3904
SVA151  003  10C7 3928    3925
SVA152  004  10CA 3929    3927
SVA154  004  10E0 3938
SVA155  004  10E7 3940
SVA156  003  10EE 3945    3939
SVA158  004  1102 3951    3947*
SVA160  004  1109 3956    3941
SVA163  003  111E 3962    3957  3959
SVA165  004  1124 3967    3929* 3961
SVA168  004  1128 3968    3963
SVA170  004  112C 3977    4644  4647  4650  4653
SVA175  003  1130 3981
SVA190  004  113D 3988    3982
SVA192  004  1144 3998    4680  4683  4689  4692
SVA194  003  1148 4002
SVA196  003  114E 4007
SVA2TD  001  0002 4473    3823
```

```
                              CROSS REFERENCE

SYMBOL   LEN VALUE DEFN   REFERENCES                        VER 15, MOD 00  04/07/22  PAGE 108

SVA200   004 115B 4019    4596  4599  4602  4608  4614  4620  4623  4626  4629  4632  4656  4659
                          4665  4668  4710  4713  4722
SVA206   003 115F 4023
SVA210   004 1165 4028
SVA220   004 116C 4033    4024
SVA250   004 1173 4043    4677  4686
SVA255   003 1177 4047
SVA260   003 117D 4052
SVA270   004 118A 4059    3968  4003  4010  4048
SVA300   003 118E 4060
SVA305   004 1194 4062    4055
SVA310   004 119B 4067    4061
SVA312   004 11A5 4070    4067*
SVA315   004 11AC 4072    3804* 3835  3863  3890  3920  3952  3989  4034  4063  4085
SVA320   004 11B0 4073    3803*
SVA330   003 11B4 4084    4395
SVA395   003 11BB 4094    3861  3889  3988  4033
SVA400   004 11BE 4098    4062
SVA410   004 11C2 4102    4258  4300  4312
SVA415   003 11CE 4108
SVA420   003 11D1 4112
SVA430   003 11D8 4117
SVA440   003 11DE 4122
SVA450   003 11E4 4127
SVA460   003 11EA 4132
SVA465   003 11F6 4136    4133
SVA470   003 1208 4147
SVA480   004 1214 4156    4118
SVA490   003 1221 4165    4123  4148
SVA500   004 1227 4176    4128
SVA505   003 122E 4181
SVA510   003 1235 4189
SVA520   003 123B 4195    4182
SVA525   003 123E 4196    4150  4159  4166  4190
SVA530   004 1241 4207    4135  4137  4139  4141
SVA535   004 1256 4213    4210
SVA540   003 1264 4220
SVA545   004 1267 4221    4296
SVA550   004 126B 4226
SVA560   004 1279 4234
SVA565   003 1285 4237    4245
SVA570   003 1288 4238    4236* 4244*
SVA580   004 129B 4249    4234*
SVA590   003 12A6 4255
SVA600   004 12B3 4262    4251
SVA610   003 12BA 4267
SVA620   006 12D1 4278    4263
SVA625   004 12DA 4280    4278*
SVA630   003 12E7 4291    4212  4214  4216
SVA632   003 12FC 4297    4293
SVA635   003 12FF 4298    4208
SVA640   003 1306 4308    4227  4229  4322
SVA645   004 1309 4309    4257  4313
SVA650   004 131B 4321    4235* 4240
SVA660   004 1323 4330    4104* 4196  4269* 4273  4278  4283
SVA670   004 1327 4331    4098*
```

```
                              CROSS REFERENCE

SYMBOL  LEN VALUE DEFN    REFERENCES                          VER 15, MOD 00  04/07/22  PAGE 109

SVA700  004  132B 4340    3882  3885  3908  3984  4009  4028  4054
SVA710  004  132F 4344    4350
SVA720  004  1336 4349
SVA750  004  133E 4354    4340*
SVA900  004  1342 4363    3883  3914  3946  4059  4102  4268  4344
SVA902  003  1349 4368    4390  4432
SVA904  003  134F 4373
SVA906  003  1361 4379    4377
SVA908  003  1364 4380    4375
SVA910  003  1367 4384    4369
SVA915  003  136D 4386    4388
SVA920  003  137E 4394    4385
SVA925  003  1385 4399
SVA928  003  1392 4403    4400
SVA930  003  139F 4410    4404
SVA940  004  13CF 4426    4363* 4411  4413  4415  4421
SVA950  004  13D3 4430    4402  4406  4417  4419  4423  4425
SVA960  003  13DE 4443    3805  3806  3846  3872  3899  3917  3928  3931  3933  3949  3977  3998
                         4019  4043  4069  4112  4157  4177  4220  4255  4271  4282  4291  4297
                         4298  4308  4310  4373  4378  4379  4380  4431
SVA966  003  13E4 4448    4455
SVA970  004  13E7 4449    4443*
SVA975  003  13EB 4453

TOTAL STATEMENTS IN ERROR IN THIS ASSEMBLY =      0

OL105 I   THE CODE LENGTH OF #KSYMB IS  6656 DECIMAL.
OL103 I   TOTAL NUMBER OF LIBRARY SECTORS REQUIRED IS   21
          NAME-#KSYMB,PACK-R1R1R1,UNIT-R1,RETAIN-P,LIBRARY-R,CATEGORY-000
```

```
        OVERLAY LINKAGE EDITOR CORE USAGE MAP      04/07/22


  START      CATEGORY   NAME AND      CODE LENGTH
 ADDRESS                  ENTRY     HEXADECIMAL DECIMAL

  0C00          0      #KSYMB          1A00     6656


OL100 I  THE TOTAL CORE USED BY #KSYMB IS  6656 DECIMAL.
OL101 I  THE START CONTROL ADDRESS OF THIS MODULE IS 0C00.
OL104 I  TOTAL NUMBER OF LIBRARY SECTORS REQUIRED IS    27
         NAME-#KSYMB,PACK-R1R1R1,UNIT-R1,RETAIN-P,LIBRARY-O
```