```
 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  29/02/16  PAGE   1

#GUFUD    MODULE
```

```
ERR LOC  OBJECT CODE      ADDR STMT SOURCE STATEMENT        VER 15, MOD 00  29/02/16  PAGE   2

    0000                        1 #GUFUD START 0
                                2        PRINT ON,NODATA
                                3 *      @SYS  EXP-Y
                                5+       PRINT ON
```

```
                        7+****************************************************************************
                        8+*        CPU EQUATES                                                       *
                        9+****************************************************************************
                       10+*
                       11+***      REGISTER EQUATES
                       12+*
              0002     13+@REGL  EQU   2                          HARDWARE REGISTER LENGTH
              0001     14+@BR    EQU   1                          BASE REGISTER
              0002     15+@XR    EQU   2                          USABLE INDEX REGISTER
              0004     16+@PSR   EQU   4                          PROGRAM STATUS REGISTER
              0008     17+@ARR   EQU   8                          ADDRESS RECALL REGISTER
              0010     18+@IAR   EQU   16                         INSTRUCTION ADDRESS REGISTER
              0020     19+@P1IAR EQU   32                         PROGRAM LEVEL 1 IAR
              0040     20+@P2IAR EQU   64                         PROGRAM LEVEL 2 IAR
              00C0     21+@I1IAR EQU   X'C0'                      INTERRUPT LEVEL 1 IAR Q-CODE
                       22+*
                       23+***      EQUATES FOR BYTES OF AN INSTRUCTION
                       24+*
              0001     25+@Q     EQU   1                          Q-CODE BYTE
              0001     26+@VQ    EQU   1                          VARIABLE Q CODE FOR LENGTH
              0002     27+@D1    EQU   2                          1ST DISPLACEMENT
              0003     28+@OP1   EQU   3                          1ST ADDRESS
              0004     29+@DOP2  EQU   4                          2ND ADDR OF 5 BYTE INSTR.
              0004     30+@OPD2  EQU   4                          2ND DISP OF 5 BYTE INSTR.
              0003     31+@DD2   EQU   3                          2ND DISP OF 4 BYTE INSTR.
              0005     32+@OP2   EQU   5                          2ND ADDR OF 5 BYTE INSTR.
              0003     33+@INST3 EQU   3                          LENGTH OF 1 DISP INSTRUCTION
              0004     34+@INST4 EQU   4                          LENGTH OF 1 ADDR INSTRUCTION
              0005     35+@INST5 EQU   5                          LENGTH OF 1 DISP 1 ADDR INSTR.
              0006     36+@INST6 EQU   6                          LENGTH OF 2 ADDR INSTR.
                       37+*
                       38+***      CONDITION CODES FOR BRANCHES
                       39+*
              0087     40+@UCB   EQU   X'87'                      UNCONDITIONAL BRANCH
              0080     41+@NOP   EQU   X'80'                      NO BRANCH
              0084     42+@BH    EQU   X'84'                      BRANCH HIGH
              0082     43+@BL    EQU   X'82'                      BRANCH LOW
              0081     44+@BE    EQU   X'81'                      BRANCH EQUAL
              0004     45+@BNH   EQU   X'04'                      BRANCH NOT HIGH
              0002     46+@BNL   EQU   X'02'                      BRANCH NOT LOW
              0001     47+@BNE   EQU   X'01'                      BRANCH NOT EQUAL
              0088     48+@BOZ   EQU   X'88'                      BRANCH OVERFLOW ZONED
              00A0     49+@BOL   EQU   X'A0'                      BRANCH OVERFLOW LOGICAL
              0008     50+@BNOZ  EQU   X'08'                      BRANCH NO OVERFLOW ZONED
              0020     51+@BNOL  EQU   X'20'                      BRANCH NO OVERFLOW LOGICAL
              0010     52+@BT    EQU   X'10'                      BRANCH TRUE
              0090     53+@BF    EQU   X'90'                      BRANCH FALSE
              0084     54+@BP    EQU   X'84'                      BRANCH PLUS
              0082     55+@BM    EQU   X'82'                      BRANCH MINUS
              0081     56+@BZ    EQU   X'81'                      BRANCH ZERO
              0004     57+@BNP   EQU   X'04'                      BRANCH NOT PLUS
              0002     58+@BNM   EQU   X'02'                      BRANCH NOT MINUS
              0001     59+@BNZ   EQU   X'01'                      BRANCH NOT ZERO
                       60+*
                       61+***      MISCELLANEOUS CONSTANTS
                       62+*
```

```
                          0000   63+@ZERO  EQU   0                        ZERO
                          0001   64+@B1    EQU   1                        BINARY ONE
                          00F0   65+@DZERO EQU   X'F0'                    DECIMAL ZERO
                          0040   66+@BLANK EQU   C' '                     EBCDIC BLANK
                          006B   67+@COMMA EQU   C','                     EBCDIC COMMA
                          0061   68+@SLASH EQU   C'/'                     EBCDIC FORWARD SLASH
                          005B   69+@DOLAR EQU   C'$'                     EBCDIC DOLLAR SIGN
                          005C   70+@ASTER EQU   C'*'                     EBCDIC ASTERISK
                          007B   71+@NUMBR EQU   C'#'                     EBCDIC NUMBER #
                          007C   72+@ASIGN EQU   C'@'                     EBCDIC ASSIGN @
                          00C1   73+@CHARA EQU   C'A'                     EBCDIC CHAR A
                          00C6   74+@CHARF EQU   C'F'                     EBCDIC CHAR F
                          00D9   75+@CHARR EQU   C'R'                     EBCDIC CHAR R
                          00E9   76+@CHARZ EQU   C'Z'                     EBCDIC CHAR Z
                          001E   77+@EOS   EQU   X'1E'                    RETURN CARRIAGE
                          001C   78+@EOF   EQU   X'1C'                    END OF FILE CHARACTER
                          005A   79+@UPARW EQU   X'5A'                    UPARROW FROM KEYBOARD INPUT
                          004E   80+@CPLUS EQU   C'+'                     EBCDIC PLUS SIGN
                          0060   81+@MINUS EQU   C'-'                     EBCDIC MINUS SIGN
                          0001   82+@DCALK EQU   X'01'                    DCAL REQUESTED INDICATOR
                          0020   83+@PGCSZ EQU   32                       CORE SIZE IN PAGES
                          2000   84+@MINCR EQU   256*@PGCSZ               CORE SIZE IN BYTES
                          00F4   85+@LINSZ EQU   244                      LENGTH OF INPUT LINE BUFFER
                          0018   86+@DTRSZ EQU   24                       NO. OF DISK SECTORS PER TRACK
                          0030   87+@SECCY EQU   48                       SECTORS PER CYLINDER
                          0060   88+@CARDL EQU   96                       LENGTH OF 3700 INPUT CARD
                          0050   89+@BCRDL EQU   80                       LENGTH OF 5081 INPUT CARD
                          0005   90+@MAPEN EQU   5                        DISP TO END OF FE CORE MAP
                          0007   91+@SDFLN EQU   7                        LENGTH OF SDF
                          0006   92+@VOLID EQU   6                        LENGTH OF DISK ID FIELD
                          0007   93+@HDRLN EQU   7                        LENGTH OF PROGRAM HEADER
                          0011   94+@CLON  EQU   X'11'                    TURN ON COMMAND LITE Q-CODE
                          0010   95+@CLOFF EQU   X'10'                    TURN Off COMMAND LITE Q-CODE

                                 97+*********************************************************************
                                 98+*       DISK REGION EQUATES                                       *
                                 99+*********************************************************************
                                100+*
                          0100  101+@SCTSZ EQU   256                      LENGTH OF ONE SECTOR
                          0500  102+@WSFIT EQU   X'0500'                  SECTOR ADDR OF WS FIT SCTRS
                          0503  103+@WSTBL EQU   X'0503'                  SECTOR ADDR OF WORKING STORAGE
                          0005  104+@DWBCY EQU   5                        BASE CYL SYSTEM WORK FILE
                          0003  105+@DWTB1 EQU   3                        LOGICAL SCTR 1ST TEXT BLOCK
                          00C0  106+@DWSIZ EQU   192                      NO. OF WORK FILE DISK SECTORS
                          0004  107+@DSBCY EQU   4                        BASE CYL SYSTEM ROUTINES
                          0000  108+@DSCS1 EQU   0                        COMPILER SUBROUTINE 1ST SCTR
                          0007  109+@DVBCY EQU   7                        BASE CYL VIRTUAL MEMORY
                          0000  110+@VMFD1 EQU   0                        FILE DIRECTORY 1 PAGE
                          0001  111+@VMFD2 EQU   1                        FILE DIRECTORY 2 PAGE
                          0001  112+@VMTRL EQU   1                        TRACE REFERENCE LIST PAGE
                          0002  113+@VMRS3 EQU   2                        START OF VM RESIDENT SUBROUTINE
                          0056  114+@VENTA EQU   86                       FIRST PSEUDO CODE PAGE IN VM
                          00FE  115+@VMDDV EQU   254                      FUNC AND ARRAY TABLE - PAGE ONE
                          0009  116+@DCBCY EQU   9                        BASE CYL COMPILER VADDR TABLES
                          0040  117+@DCST1 EQU   64                       STMT ADDRESS TABLE 1ST SECTOR
                          0050  118+@DCBT1 EQU   80                       BRANCH ADDRESS TABLE 1ST SECTOR
```

```
      @SYSEQ - SYSTEM SOFTWARE EQUATES
```

```
                            120+******************************************************************************
                            121+*        DISK IOCR EQUATES                                                  *
                            122+******************************************************************************
                            123+*
                            124+***     DISK PARAMETER LIST (DPL) EQUATES
                            125+*
                   0000     126+@DCTRL EQU    0                             CONTROL PARAMETER
                   0001     127+@DCYL  EQU    1                             LOGICAL CYLINDER NUMBER
                   0002     128+@DSAD  EQU    2                             HEAD/SECTOR ADDRESS
                   0003     129+@DCNT  EQU    3                             SECTOR COUNT
                   0004     130+@DBFR1 EQU    4                             1ST BYTE OF DATA AREA
                   0005     131+@DBFR2 EQU    5                             DATA AREA ADDRESS
                   0002     132+@DSPIN EQU    X'02'                         SPINDLE BIT IN DISK ADDRESS
                   0006     133+@DPLNG EQU    6                             LENGTH OF DSL
                   0000     134+@DPOS  EQU    X'00'                         DPL - SEEK FUNCTION CODE
                   0001     135+@DGET  EQU    X'01'                         DPL - READ FUNCTION CODE
                   0002     136+@DPUT  EQU    X'02'                         DPL - WRITE FUNCTION CODE
                   0031     137+@DVRFY EQU    X'31'                         DPL - VERIFY FUNCTION CODE
                   00FF     138+@DWAIT EQU    X'FF'                         DPL - WAIT I/O COMPLETE FUNC COD
                   0003     139+@DSIVF EQU    X'03'                         SIO CTRL CODE FOR VERIFY
                            140+*
                   0002     141+@DADDR EQU    2                             LENGTH OF DISK ADDRESS
                   0002     142+@VADDR EQU    2                             LENGTH OF VIRTUAL ADDRESS
                   0002     143+@CADDR EQU    2                             LENGTH OF CORE ADDRESS

                            145+******************************************************************************
                            146+*        PRINT PARAMETER LIST (PPL) EQUATES                                  *
                            147+******************************************************************************
                            148+*
                   0004     149+@PPLNG EQU    4                             LENGTH OF PPL
                   0000     150+@PCTRL EQU    0                             CONTROL BYTE DISPLACEMENT
                   0001     151+@PRCNT EQU    1                             COUNT BYTE DISPLACEMENT
                   0003     152+@PDATA EQU    3                             DATA ADDR DISPLACEMENT
                   0040     153+@PRINT EQU    X'40'                         PRINT CONTROL
                   0080     154+@RETRN EQU    X'80'                         RETURN CARRIER CONTROL
                   00C0     155+@PRETR EQU    @PRINT+@RETRN                 PRINT AND RETURN CARRIER
                   0010     156+@TBLEF EQU    X'10'                         TAB LEFT CONTROL
                   0001     157+@INDEX EQU    X'01'                         INDEX FORMS CONTROL
                   0011     158+@TBLIX EQU    @TBLEF+@INDEX                 TAB LEFT AND INDEX CONTROL
                   00FF     159+@PWAIT EQU    X'FF'                         WITH AND CHECK ERROR CONTROL
                   004F     160+@RLDWN EQU    X'4F'                         ROLL DOWN CONTROL (CRT ONLY)
                   0000     161+@TBCNT EQU    0                             TAB LEFT COUNT
                   0080     162+@RTRNC EQU    X'80'                         CARRIER RETURN COUNT
                   0075     163+@EOFTC EQU    X'75'                         EOF RECORD TYPE CODE
                            164+*
                            165+***     STATEMENT/SEGMENT HEADER EQUATES
                            166+*
                   0000     167+@SDF0  EQU    0                             DISP TO NULL SEG INDICATOR
                   0001     168+@SDF1  EQU    1                             DISP TO LENGTH OF SEGMENT
                   0002     169+@SDF2  EQU    2                             DISP TO SEGMENTATION CODE
                   0003     170+@SDF3  EQU    3                             DISP TO END OF SDF
                   0005     171+@SBLN  EQU    5                             DISP TO STMT BINARY LINE NO.
                   0006     172+@STYPE EQU    6                             DISP TO STMT TYPE CODE
                   0007     173+@STEXT EQU    7                             DISP TO 1ST TEXT BYTE OF STMT
                   0080     174+@SNULL EQU    X'80'                         MASK FOR NULL SEG INDICATOR
                            175+*                                           *   1 = SEGMENT IS NULL
```

```
                              176+*                                    * 0 = SEGMENT IS NOT NULL
                              177+*
                              178+*                         FOLLOWING ARE THE MASKS FOR THE SEGMENTATION
                              179+*                         CODE. THE SEGMENTATION IS INDICATED BY VALUE
                              180+*                         IN @SDF2 AS FOLLOWS:
                    0000      181+@SONLY EQU    0                       ONLY SEG. IN RECORD
                    0001      182+@SIST  EQU    1                       1ST SEG. OF A MULTI-SEG RCD
                    0003      183+@SMIDL EQU    3                       MIDDLE SEG. OF A MULTI-SEG RCD
                    0002      184+@SLAST EQU    2                       LAST SEG. OF MULTI-SEG RCD
                    0002      185+@SBLNL EQU    2                       LENGTH OF STMT BINARY LINE NO.
                              186+*
                              187+****    FILE INDEX TABLE EQUATES SECTION
                              188+*
                              189+*                         ALL DISPLACEMENT ARE CALCULATED FROM THE
                              190+*                         * FIRST BYTE OF THE FIT TO THE RIGHTMOST BYTE
                              191+*                         * OF THE SPECIFIED FIELD UNLESS OTHERWISE
                              192+*                         * NOTED.
                              193+*
                    0002      194+@FDLNC EQU    2                       DISP TO FILE LINE COUNT
                    0002      195+@FLLNC EQU    2                       LNG OF FILE LINE COUNT FIELD
                    0000      196+@FDDBC EQU    0                       DISP TO FILE DATA BLOCK COUNT
                    0001      197+@FLDBC EQU    1                       LNG OF FILE DATA BLOCK COUNT
                    0009      198+@FLACE EQU    9                       DISP O ADDR OF CURR ENTRY
                    000B      199+@FDFNA EQU    11                      DISP TO ADDR OF 1ST NULL ENTRY
                    0002      200+@FLFNA EQU    2                       LNG OF ADDR OF 1ST NULL ENTRY
                    000C      201+@FDE1  EQU    12                      DISP TO 1ST BYTE OF 1ST ENTRY
                    0004      202+@FLENT EQU    4                       LNG OF A FIT ENTRY
                              203+*
                              204+*                         ENTRY FIELD DISPLACEMENTS ARE CALCULATED FROM
                              205+*                         * THE 1ST BYTE OF THE ENTRY.
                              206+*
                    0000      207+@FDSD  EQU    0                       DISP TO DB SECTOR DISP
                    0001      208+@FLSD  EQU    1                       LNG OF DB SECTOR DISP FIELD
                    0002      209+@FDHLN EQU    2                       DISP TO HIGH LINE NO. FIELD
                    0002      210+@FLHLN EQU    2                       LNG OF HIGH LINE NO. FIELD
                    0003      211+@FDNSC EQU    3                       DISP TO DB NULL SPACE CNT FIELD
                    0001      212+@FLNSC EQU    1                       LNG OF DB NULL SPACE CNT FIELD
                              213+*
                              214+*        END OF SYSTEM SOFTWARE EQUATES
                              215+        PRINT ON
                              216 *       @ERM  EXP-Y
                              218+        PRINT ON
```

```
       @ERMEQ - GENERAL ERROR MESSAGE EQUATES
```

```
                           220+*************************************************************************
                           221+*        ERROR MESSAGES EQUATES                                         *
                           222+*************************************************************************
                      0000 223+@@E100 EQU   0                            FIRST CHARACTER NOT
                           224+*                                         * ALPHABETIC
                      0001 225+@@E101 EQU   @@E100+1                     FIRST CHARACTER NOT
                           226+*                                         * <ALPHAMERIC CHARACTER>
                      0002 227+@@E102 EQU   @@E101+1                     PASSWORD OR FILENAME LONGER
                           228+*                                         * THEN 8 CHARACTERS
                      0003 229+@@E103 EQU   @@E102+1                     <DISK LABEL> LONGER THEN 6
                           230+*                                         * CHARACTERS
                      0004 231+@@E110 EQU   @@E103+1                     COMMA FOLLOWED BY NOTHING
                           232+*                                         *
                      0005 233+@@E112 EQU   @@E110+1                     <ARITHMETIC CONSTANT> CONTAINS
                           234+*                                         * 2 DECIMAL POINTS
                      0006 235+@@E113 EQU   @@E112+1                     DECIMAL POINT WITHOUT
                           236+*                                         * <ARITHMETIC CONSTANT>
                      0007 237+@@E114 EQU   @@E113+1                     INCOMPLETE <CHARACTER
                           238+*                                         * CONSTANT>
                      0008 239+@@E115 EQU   @@E114+1                     INVALID <SYSTEM CONSTANT>
                           240+*                                         *
                      0009 241+@@E116 EQU   @@E115+1                     VARIABLE IS NOT FOLLOWED BY A
                           242+*                                         * COMMA OR EQUAL SIGN
                      000A 243+@@E117 EQU   @@E116+1                     INVALID EXPONENT IN CONSTANT
                           244+*                                         *
                      000B 245+@@E120 EQU   @@E117+1                     NON-NUMERIC CHARACTER IN <LINE
                           246+*                                         * NUMBER> OR INEGER
                      000C 247+@@E122 EQU   @@E120+1                     MORE THAN 4 DIGITS IN <LINE
                           248+*                                         * NUMBER> OR INTEGER
                      000D 249+@@E123 EQU   @@E122+1                     UNBALANCED LINE NUMBER SERIES
                           250+*                                         *
                      000E 251+@@E124 EQU   @@E123+1                     LINE NUMBER IS NOT GREATER
                           252+*                                         * THAN PREVIOUS LINE NUMBER
                      000F 253+@@E129 EQU   @@E124+1                     PARAMETER FOUND WHERE NONE
                           254+*                                         * IS ALLOWED
                      0010 255+@@E130 EQU   @@E129+1                     REQUIRED PARAMETER MISSING
                           256+*                                         *
                      0011 257+@@E131 EQU   @@E130+1                     INVALID PARAMETER
                           258+*                                         *
                      0012 259+@@E133 EQU   @@E131+1                     TOO MANY <PARAMETERS>
                           260+*                                         *
                      0013 261+@@E134 EQU   @@E133+1                     DUPLICATE <PARAMETER>
                           262+*                                         *
                      0014 263+@@E135 EQU   @@E134+1                     INVALID USE OF ONE OR TWO
                           264+*                                         * STAR FILENAME
                      0015 265+@@E136 EQU   @@E135+1                     INVALID COMBINATION OF KEYWORDS
                           266+*                                         * <PARAMETERS>
                      0016 267+@@E137 EQU   @@E136+1                     NO <LINE-NUMBER-LIST>
                           268+*                                         * SPECIFIED
                      0017 269+@@E138 EQU   @@E137+1                     UNBALANCED QUOTES IN
                           270+*                                         * <CHARACTER CONSTANT>
                      0018 271+@@E139 EQU   @@E138+1                     INVALID <DELIMITER>
                           272+*                                         *
                      0019 273+@@E142 EQU   @@E139+1                     INCOMPLETE KEYWORD
                           274+*                                         * MISSING DASH
                      001A 275+@@E143 EQU   @@E142+1                     INCOMPLETE KEYWORD
```

```
                              276+*                                       * SECOND WORD UNRECOGNIZABLE
                    001B      277+@@E150 EQU    @@E143+1                   INVALID BASIC VARIABLE
                              278+*                                       *
                    001C      279+@@E151 EQU    @@E150+1                   VARIABLE SUBSCRIPT NOT
                              280+*                                       * AN INTEGER
                    001D      281+@@E160 EQU    @@E151+1                   MIXED DATA TYPE IN
                              282+*                                       * ASSIGNMENT
                    001E      283+@@E162 EQU    @@E160+1                   UNBALANCED <LABEL-PAIR>
                              284+*                                       *
                    001F      285+@@E163 EQU    @@E162+1                   DIFFERENT VARIABLE TYPES
                              286+*                                       * IN <LABEL-PAIR>
                    0020      287+@@E164 EQU    @@E163+1                   ODD TRACK NUMBER NOT
                              288+*                                       * ALLOWABLE
                    0021      289+@@E200 EQU    @@E164+1                   NO CURRENT <PASSWORD> OR
                              290+*                                       * DISK DEFINED
                    0022      291+@@E205 EQU    @@E200+1                   HELP TEXT NOT FOUND
                              292+*                                       *
                    0023      293+@@E210 EQU    @@E205+1                   <PASSWORD> NOT ON SPCIFIED
                              294+*                                       * DISK
                    0024      295+@@E211 EQU    @@E210+1                   SPECIFIED FILE NOT FOUND
                              296+*                                       *
                    0025      297+@@E212 EQU    @@E211+1                   DUPLICATE DISK LABELS
                              298+*                                       * ON SYSTEM
                    0026      299+@@E213 EQU    @@E212+1                   FILE NOT ON SYSTEM
                              300+*                                       *
                    0027      301+@@E215 EQU    @@E213+1                   SPECIFIED FILE PROTECTED
                              302+*                                       *
                    0028      303+@@E216 EQU    @@E215+1                   DISK LABEL NOT ON SPECIFIED
                              304+*                                       * LOCATION
                    0029      305+@@E217 EQU    @@E216+1                   SPECIFIED DISK NOT ON
                              306+*                                       * SYSTEM
                    002A      307+@@E220 EQU    @@E217+1                   NO <WORK FILE> DEFINED
                              308+*                                       * .
                    002B      309+@@E221 EQU    @@E220+1                   <WORK FILE> IS PROGRAM
                              310+*                                       * GENERATED
                    002C      311+@@E222 EQU    @@E221+1                   WORK FILE IS PROTECTED
                              312+*                                       *
                    002D      313+@@E223 EQU    @@E222+1                   NO PROGRAM FILE IN
                              314+*                                       * <WORK FILE>
                    002E      315+@@E225 EQU    @@E223+1                   NO PROGRAM IN PAUSE STATE
                              316+*                                       *
                    002F      317+@@E226 EQU    @@E225+1                   <WORK FILE> IS EMPTY
                              318+*                                       * .
                    0030      319+@@E227 EQU    @@E226+1                   SPECIFIED FILE NOT
                              320+*                                       * A PROGRAM FILE
                    0031      321+@@E228 EQU    @@E227+1                   ONE-STAR OR TWO-STAR
                              322+*                                       * FILE PROTECTED
                    0032      323+@@E229 EQU    @@E228+1                   DESIRED CONDITION ALREADY
                              324+*                                       * PRESENT-FUNCTION IGNORED
                    0033      325+@@E230 EQU    @@E229+1                   FUNCTION REQUIRES WORK AREA
                              326+*                                       *
                    0034      327+@@E232 EQU    @@E230+1                   FUNCTION INVALID IN
                              328+*                                       * PAUSE STATE
                    0035      329+@@E234 EQU    @@E232+1                   ONLY MOUNT OR INITIALIZE
                              330+*                                       * COMMAND VALID
                    0036      331+@@E237 EQU    @@E234+1                   ORIGINAL MODE OF EXECUTION
```

```
                         332+*                                      * NOT 'TRACE'
              0037  333+@@E240 EQU    @@E237+1                DATA RECORDER NOT ON SYSTEM
                         334+*                                      *
              0038  335+@@E241 EQU    @@E240+1                CRT NOT ON SYSTEM
                         336+*                                      *
              0039  337+@@E242 EQU    @@E241+1                DRIVE 2 NOT ON SYSTEM
                         338+*                                      *
              003A  339+@@E248 EQU    @@E242+1                CRT SPECIFIED WHEN INPUT IS
                         340+*                                      * FROM CARDS OR PROCEDURE
              003B  341+@@E249 EQU    @@E248+1                CARD OUTPUT SPECIFIED WHEN
                         342+*                                      * INPUT IS FROM CARDS
              003C  343+@@E250 EQU    @@E249+1                VARIABLE NOT IN PROGRAM
                         344+*                                      *
              003D  345+@@E251 EQU    @@E250+1                <ARITHMETIC CONSTANT> NOT IN
                         346+*                                      * RANGE 1E-99 < X < 1E99
              003E  347+@@E252 EQU    @@E251+1                SUBSCRIPT EXCEEDS <ARRAY SIZE
                         348+*                                      * LIMIT>.
              003F  349+@@E253 EQU    @@E252+1                ARRAY NOT IN PROGRAM.
                         350+*                                      *
              0040  351+@@E254 EQU    @@E253+1                NO NON-ARRAY <VARIABLES> IN
                         352+*                                      * PROGRAMS
              0041  353+@@E255 EQU    @@E254+1                NO <VARIABLES> IN PROGRAM
                         354+*                                      *
              0042  355+@@E256 EQU    @@E255+1                INCONSISTENT NUMBER
                         356+*                                      * OF SUBSCRIPTS
              0043  357+@@E300 EQU    @@E256+1                REQUIRED <FILE LIBRARY AREA>
                         358+*                                      * SPACE NOT AVAILABLE
              0044  359+@@E301 EQU    @@E300+1                PREVIOUS FILENAME NOT
                         360+*                                      * ALLOCATED
              0045  361+@@E302 EQU    @@E301+1                NEW FILENAME ALREADY
                         362+*                                      * ALLOCATED
              0046  363+@@E303 EQU    @@E302+1                TWELVE FILES ALREADY ALLOCATED
                         364+*                                      * FOR WORK FILE PROGRAM
              0047  365+@@E304 EQU    @@E303+1                'NEW' FILE SPECIFIED ALREADY
                         366+*                                      * IS IN USER LIBRARY
              0048  367+@@E305 EQU    @@E304+1                'SPACE' PARAMETER EXECEEDS 256
                         368+*                                      * .
              0049  369+@@E308 EQU    @@E305+1                SPECIFIED <LINE NUMBER>
                         370+*                                      * DOES NOT EXIST
              004A  371+@@E310 EQU    @@E308+1                USER FILE POOLED
                         372+*                                      *
              004B  373+@@E315 EQU    @@E310+1                <PROGRAM-GENERATED DATA FILE>
                         374+*                                      * LARGER THEN WORK FILE
              004C  375+@@E316 EQU    @@E315+1                NO EXECUTED BASIC PROGRAM
                         376+*                                      * .
              004D  377+@@E320 EQU    @@E316+1                SCP NOT AVAILABLE ON SYSTEM
                         378+*                                      * DISK
              004E  379+@@E325 EQU    @@E320+1                LINE NUMBER LIST TOO LONG
                         380+*                                      *
              004F  381+@@E330 EQU    @@E325+1                HELP KEYWORD NOT RECOGNIZED
                         382+*                                      *
              0050  383+@@E335 EQU    @@E330+1                LINE NO. LIST SPECIFIED FOR
                         384+*                                      * <PROGRAM-GENERATED FILE>
              0051  385+@@E338 EQU    @@E335+1                INVALID COMBINATION OF
                         386+*                                      * <PARAMETERS>
              0052  387+@@E340 EQU    @@E338+1                NO ONE-STAR OR TWO STAR
```

```
                          388+*                                    * LIBRARIES ON SYSTEM
                     0053 389+@@E350 EQU    @@E340+1               83 <PASSWORDS> ALREADY DEFINED
                          390+*                                    * ON DISK
                     0054 391+@@E351 EQU    @@E350+1               NO <FILE LIBRARY AREA> ON
                          392+*                                    * SPECIFIED DISK
                     0055 393+@@E352 EQU    @@E351+1               FILE LIBRARY FRAGMENTED,
                          394+*                                    * USE PACK COMMAND
                     0056 395+@@E360 EQU    @@E352+1               MERGED FILE WOULD CONTAIN
                          396+*                                    * MORE THEN 990 LINES
                     0057 397+@@E361 EQU    @@E360+1               INCOMPATIBLE FILE TYPES
                          398+*                                    * FOR <MERGE>
                     0058 399+@@E362 EQU    @@E361+1               MERGED FILE WOULD EXCEED
                          400+*                                    * <WORK FILE> SIZE LIMIT
                     0059 401+@@E371 EQU    @@E362+1               <REMOVE> COMMAND NOT
                          402+*                                    * PREVIOUSLY ISSUED
                     005A 403+@@E380 EQU    @@E371+1               <PASSWORD> PREVIOUSLY DEFINED
                          404+*                                    *
                     005B 405+@@E390 EQU    @@E380+1               POOLED FILENAME ALREADY
                          406+*                                    * DEFINED
                     005C 407+@@E400 EQU    @@E390+1               CURRENT PASSWORD/DISK NOT THE
                          408+*                                    * SAME AS CREATING USER
                     005D 409+@@E410 EQU    @@E400+1               DISK LABEL NOT SAME AS LAST
                          410+*                                    * MOUNTED
                     005E 411+@@E415 EQU    @@E410+1               INVALID COMMAND KEY
                          412+*                                    *
                     005F 413+@@E417 EQU    @@E415+1               INVALID COMMAND SPECIFICATION
                          414+*                                    *
                     0060 415+@@E420 EQU    @@E417+1               USER FILENAME ALREADY DEFINED
                          416+*                                    *
                     0061 417+@@E430 EQU    @@E420+1               INVALID PARTIAL <RENUMBER>
                          418+*                                    * .
                     0062 419+@@E432 EQU    @@E430+1               MAX <LINE NUMBER> WOULD BE
                          420+*                                    * EXCEEDED IF RENUMBERED
                     0063 421+@@E433 EQU    @@E432+1               <RENUMBER> <INCREMENT> IS ZERO
                          422+*                                    *
                     0064 423+@@E450 EQU    @@E433+1               ANOTHER PROGRAM IS SUSPENSION
                          424+*                                    *
                     0065 425+@@E451 EQU    @@E450+1               SCRATCH FILE IN USE
                          426+*                                    *
                     0066 427+@@E460 EQU    @@E451+1               RIGHT MARGIN EXCEEDS
                          428+*                                    * PRINTER SIZE
                     0067 429+@@E461 EQU    @@E460+1               <WIDTH> LESS THAN 18
                          430+*                                    *
                     0068 431+@@E464 EQU    @@E461+1               NO SUSPENDED PROGRAM
                          432+*                                    *
                     0069 433+@@E465 EQU    @@E464+1               MISSING 'OPEN' DISK FILE
                          434+*                                    *
                     006A 435+@@E466 EQU    @@E465+1               SUSPENDED CONFIGURATION
                          436+*                                    * DIFFERS FROM CURRENT SYSTEM
                     006B 437+@@E467 EQU    @@E466+1               'OPEN' DISK FILE HAS BEEN
                          438+*                                    * MODIFIED
                     006C 439+@@E469 EQU    @@E467+1               DISK FOUND DEFECTIVE
                          440+*                                    *
                     006D 441+@@E470 EQU    @@E469+1               TRACK ALREADY ASSIGNED OR
                          442+*                                    * NOT AVAILABLE
                     006E 443+@@E471 EQU    @@E470+1               INVALID SECONDARY
```

```
                              444+*                                     * INITIALIZATION
                   006F       445+@@E473 EQU    @@E471+1                 DISK ALREADY CONTAINS A
                              446+*                                     * <FILE LIBRARY AREA>
                   0070       447+@@E474 EQU    @@E473+1                 SPACE NOT AVAILABLE FOR FILE
                              448+*                                     *
                   0071       449+@@E475 EQU    @@E474+1                 NO MORE ALTERNATE TRACKS
                              450+*                                     *
                   0072       451+@@E476 EQU    @@E475+1                 CRT, PROCESSING UNIT,
                              452+*                                     * COMMAND KEY CONFLICT
                   0073       453+@@E477 EQU    @@E476+1                 INVALID KEYBOARD TYPE
                              454+*                                     *
                   0074       455+@@E478 EQU    @@E477+1                 ACTIVE FILE(S) ON DISK
                              456+*                                     *
                   0075       457+@@E479 EQU    @@E478+1                 SPECIFIED FILE NOT ON DISK
                              458+*                                     *
                   0076       459+@@E480 EQU    @@E479+1                 FILES IN AREA TO BE DELETED
                              460+*                                     *
                   0077       461+@@E481 EQU    @@E480+1                 CYLINDER 0 DEFECTIVE
                              462+*                                     *
                   0078       463+@@E482 EQU    @@E481+1                 SPECIFIED <TRACK> EXCEEDS DISK
                              464+*                                     * CAPACITY
                   0079       465+@@E483 EQU    @@E482+1                 VTOC FULL
                              466+*                                     *
                   007A       467+@@E484 EQU    @@E483+1                 SPACE NOT AVAILABLE BEGINNING
                              468+*                                     * AT <TRACK> SPECIFIED
                   007B       469+@@E485 EQU    @@E484+1                 WORK AREA SPACE ALLOCATED FOR
                              470+*                                     * ANOTHER PURPOSE
                   007C       471+@@E486 EQU    @@E485+1                 <TRACK> NOT USABLE
                              472+*                                     *
                   007D       473+@@E487 EQU    @@E486+1                 NUMBER OF TRACKS REQUESTED
                              474+*                                     * EXCEEDS DISK CAPACITY
                   007E       475+@@E488 EQU    @@E487+1                 CONTRACTION PARAMETER EXCEED
                              476+*                                     * LIBRARY SIZE
                   007F       477+@@E489 EQU    @@E488+1                 RELEASE LEVEL ON HELP
                              478+*                                     * TEXT IS INCORRECT
                   0080       479+@@E490 EQU    @@E489+1                 NO SUSPECTED DEFECTIVE
                              480+*                                     * TRACKS
                   0081       481+@@E491 EQU    @@E490+1                 INVALID COMPONENT NAME
                              482+*                                     *
                   0082       483+@@E492 EQU    @@E491+1                 NO 'HDR' OR 'PTF' STATEMENT
                              484+*                                     *
                   0083       485+@@E493 EQU    @@E492+1                 INCORRECT CHECKSUM
                              486+*                                     *
                   0084       487+@@E494 EQU    @@E493+1                 NO 'PTF' FILE ON DISK
                              488+*                                     *
                   0085       489+@@E495 EQU    @@E494+1                 SYSTEM RELEASE LEVEL
                              490+*                                     * INCORRECT
                   0086       491+@@E496 EQU    @@E495+1                 THIS PTF NOT IN 'PTF'
                              492+*                                     * DISK FILE
                   0087       493+@@E497 EQU    @@E496+1                 NO WORKAREA ON 'CURRENT'
                              494+*                                     * SYSTEM DISK
                   0088       495+@@E498 EQU    @@E497+1                 TRACK NOT ASSIGNED
                              496+*                                     *
                   0089       497+@@E500 EQU    @@E498+1                 LINE LENGTH LIMIT EXCEED-1
                              498+*                                     * OR MORE LINES TRUNCATED
                   008A       499+@@E501 EQU    @@E500+1                 <WORK FILE> SIZE LIMIT
```

```
                              500+*                                     * EXCEEDED - FILE TRUNCATED
                         008B 501+@@E530 EQU     @@E501+1               <WORK FILE> SIZE LIMIT
                              502+*                                     * EXCEEDED
                         008C 503+@@E531 EQU     @@E530+1               <WORK FILE> SIZE LIMIT
                              504+*                                     * EXCEEDED
                         008D 505+@@E535 EQU     @@E531+1               WRONG/ NO <WORKAREA> ON R1 OR F1
                              506+*                                     *
                         008E 507+@@E540 EQU     @@E535+1               RIGHT MARGIN EXCEEDED
                              508+*                                     * LINE IGNORED
                         008F 509+@@E541 EQU     @@E540+1               'CURRENT' PASSOWRD/DISK LABEL
                              510+*                                     * CANCELLED
                         0090 511+@@E542 EQU     @@E541+1               DISK CYLINDER SIZE DOES NOT
                              512+*                                     * MATCH MACHINE CAPACITY
                         0091 513+@@E543 EQU     @@E542+1               R1 DISK NOT INITIALIZED
                              514+*                                     *
                         0092 515+@@E544 EQU     @@E543+1               F1 DISK NOT INITIALIZED
                              516+*                                     *
                         0093 517+@@E545 EQU     @@E544+1               R2 DISK NOT INITIALIZED
                              518+*                                     *
                         0094 519+@@E546 EQU     @@E545+1               F2 DISK NOT INITIALIZED
                              520+*                                     *
                         0095 521+@@E547 EQU     @@E546+1               MINIMUM CONFIGURATION
                              522+*                                     * RECORD ASSUMED
                         0096 523+@@E549 EQU     @@E547+1               PRINTER UNAVAILABLE DUE TO
                              524+*                                     * PREVIOUS PRINTER FAILURE
                         0097 525+@@E550 EQU     @@E549+1               TRAGIC DISK ERROR - BAD
                              526+*                                     * WORK FILE
                         0098 527+@@E551 EQU     @@E550+1               TRAGIC DISK ERROR - BAD
                              528+*                                     * SAVED FILE
                         0099 529+@@E552 EQU     @@E551+1               TRAGIC DISK ERROR - 'CURRENT'
                              530+*                                     * PASSWORD NOT FOUND
                         009A 531+@@E553 EQU     @@E552+1               TRAGIC DISK ERROR - POOLED
                              532+*                                     * FILE NOT IN DIRECTORY
                         009B 533+@@E554 EQU     @@E553+1               TRAGIC DISK ERROR - BAD
                              534+*                                     * FILENAME IN POOLED DIRECTORY
                         009C 535+@@E555 EQU     @@E554+1               TRAGIC DISK ERROR - 'OPEN'
                              536+*                                     * DISK FILE GONE
                         009D 537+@@E556 EQU     @@E555+1               TRAGIC DISK ERROR - PARAMETERS
                              538+*                                     * HAVE BEEN DESTROYED
                         009E 539+@@E558 EQU     @@E556+1               CURRENT SYSTEM PROGRAM FILE
                              540+*                                     * ON DISK SPECIFIED
                         009F 541+@@E570 EQU     @@E558+1               ONE OR MORE LINES TRUNCATED
                              542+*                                     * WHEN PUNCHED
                         00A0 543+@@E571 EQU     @@E570+1               ONE OR MORE DISABLED LINES
                              544+*                                     * PUNCHED
                         00A1 545+@@E572 EQU     @@E571+1               WRONG OR NO <WORKAREA> ON F1
                              546+*                                     *
                         00A2 547+@@E573 EQU     @@E572+1               WRONG OR NO <WORKAREA> ON R1
                              548+*                                     *
                         00A3 549+@@E574 EQU     @@E573+1               NEXT AUTOMATIC LINE NUMBER
                              550+*                                     * WILL EXCEED 9999
                         00A4 551+@@E578 EQU     @@E574+1               RESPONSE NOT ALLOWED WITH
                              552+*                                     * CARDS OR PROCEDURE INPUT
                         00A5 553+@@E585 EQU     @@E578+1               REQUESTED TRACK SPACE EXCEEDS
                              554+*                                     * DISK CONFIGURATION
                         00A6 555+@@E600 EQU     @@E585+1               DIM ARRAY NAME PREVIOUSLY
```

```
                          556+*                                * DEFINED
                00A7  557+@@E601 EQU    @@E600+1               REFERENCED MATRIX NOT
                          558+*                                * PREVIOUSLY DEFINED.
                00A8  559+@@E602 EQU    @@E601+1               MATRIX REFERENCED AS VECTOR
                          560+*                                *
                00A9  561+@@E603 EQU    @@E602+1               VECTOR REFERENCED AS MATRIX
                          562+*                                *
                00AA  563+@@E604 EQU    @@E603+1               DUPLICATE DEFINITION OF USER
                          564+*                                * FUNCTION
                00AB  565+@@E606 EQU    @@E604+1               <NEXT> STATEMENT OUT OF
                          566+*                                * SEQUENCE
                00AC  567+@@E607 EQU    @@E606+1               <FOR>/NEXT NESTED INCORRECTLY
                          568+*                                *
                00AD  569+@@E608 EQU    @@E607+1               MORE THAN 9 NESTED <FOR>/NEXT
                          570+*                                * LOOPS
                00AE  571+@@E609 EQU    @@E608+1               <FOR>/NEXT LOOP INCOMPLETE
                          572+*                                *
                00AF  573+@@E610 EQU    @@E609+1               COMPILED PROGRAM TOO LARGE
                          574+*                                *
                00B0  575+@@E611 EQU    @@E610+1               TOO MANY ARRAY ELEMENTS
                          576+*                                *
                00B1  577+@@E612 EQU    @@E611+1               TOO MANY LINE NUMBER
                          578+*                                * REFERENCES
                00B2  579+@@E613 EQU    @@E612+1               STORAGE SPACE REQUIRED FOR
                          580+*                                * FILES TOO LARGE
                00B3  581+@@E614 EQU    @@E613+1               FILE LINE PREVIOUSLY TRUNCATED
                          582+*                                *
                00B4  583+@@E700 EQU    @@E614+1               NON-EXISTENT LINE NUMBER
                          584+*                                * REFERENCED
                00B5  585+@@E701 EQU    @@E700+1               NON-EXISTENT USER FUNCTION
                          586+*                                * REFERENCED
                00B6  587+@@E710 EQU    @@E701+1               REQUIRED FILE NOT ALLOCATED
                          588+*                                *
                00B7  589+@@E712 EQU    @@E710+1               INCONSISTENT INPUT/OUTPUT FILE
                          590+*                                * USAGE
                00B8  591+@@E713 EQU    @@E712+1               ALLOCATED FILE NOT A DATA FILE
                          592+*                                *
                00B9  593+@@E714 EQU    @@E713+1               INSUFFICIENT DATA FOR <GET>
                          594+*                                *
                00BA  595+@@E715 EQU    @@E714+1               OUTPUT FILE EXCEEDED
                          596+*                                *
                00BB  597+@@E716 EQU    @@E715+1               NO SPACE FOR ALLOCATED SCRATCH
                          598+*                                * FILE
                00BC  599+@@E717 EQU    @@E716+1               ALLOCATED DEVICE NOT ON SYSTEM
                          600+*                                *
                00BD  601+@@E718 EQU    @@E717+1               INVALID DATA ITEM FROM CARD
                          602+*                                * FILE
                00BE  603+@@E720 EQU    @@E718+1               NO <DATA STATEMENT> SPECIFIED
                          604+*                                *
                00BF  605+@@E721 EQU    @@E720+1               INSUFFICIENT DATA FOR READ
                          606+*                                *
                00C0  607+@@E723 EQU    @@E721+1               INVALID <FOR> LOOP EXECUTION
                          608+*                                *
                00C1  609+@@E724 EQU    @@E723+1               NO PRINT IMAGE IN 0,01;E
                          610+*                                * STATEMENT,
                00C2  611+@@E725 EQU    @@E724+1               REFERENCED STATEMENT NOT AN
```

```
      @ERMEQ - GENERAL ERROR MESSAGE EQUATES
```

```
                        612+*                                        * IMAGE
             00C3   613+@@E726 EQU    @@E725+1                <RETURN> EXECUTED WITHOUT
                        614+*                                        * ACTIVE <WSW>
             00C4   615+@@E727 EQU    @@E726+1                INVALID VARIABLE ASSIGNED
                        616+*                                        *
             00C5   617+@@E728 EQU    @@E727+1                RECURSIVE FUNCTION REFERENCE
                        618+*                                        *
             00C6   619+@@E729 EQU    @@E728+1                STATEMENT BRANCHES TO ITSELF
                        620+*                                        *
             00C7   621+@@E730 EQU    @@E729+1                EXPRESSION TOO COMPLEX TO
                        622+*                                        * EXECUTE
             00C8   623+@@E732 EQU    @@E730+1                MORE THAN 10 ACTIVE USER
                        624+*                                        * FUNCTIONS
             00C9   625+@@E752 EQU    @@E732+1                ASSIGNED MATRIX NOT
                        626+*                                        * 2-DIMENSIONAL
             00CA   627+@@E753 EQU    @@E752+1                MATRIX MULTIPLIER NOT
                        628+*                                        * 2-DIMENSIONAL
             00CB   629+@@E754 EQU    @@E753+1                MATRIX FUNCTION ARGUMENT NOT
                        630+*                                        * 2-DIMENSIONAL
             00CC   631+@@E755 EQU    @@E754+1                ASSIGNED MATRIX DIMS NOT SAME
                        632+*                                        * AS EXPR
             00CD   633+@@E756 EQU    @@E755+1                MATRIX DIMENSIONS NOT REVERSED
                        634+*                                        *
             00CE   635+@@E757 EQU    @@E756+1                ASSIGNED MATRIX DIMS NOT SAYE
                        636+*                                        * AS INV ARG
             00CF   637+@@E758 EQU    @@E757+1                MATRIX EXPR DIMENSIONS NOT
                        638+*                                        * CONFORMABLE
             00D0   639+@@E759 EQU    @@E758+1                ATTEMPTED MATRIX
                        640+*                                        * MULTIPLICATION IN PLACE
             00D1   641+@@E760 EQU    @@E759+1                SUBSCRIPT OUT OF <ARRAY SIZE
                        642+*                                        * LIMIT>
             00D2   643+@@E761 EQU    @@E760+1                DIMENSIONED OUTSIDE MAX <ARRAY
                        644+*                                        * SIZE LIMIT>
             00D3   645+@@E762 EQU    @@E761+1                MATRIX EXPRESSION DIMENSIONS
                        646+*                                        * NOT IDENTICAL
             00D4   647+@@E763 EQU    @@E762+1                NEARLY SINGULAR MATRIX
                        648+*                                        *
             00D5   649+@@E764 EQU    @@E763+1                MATRIX TOO LARGE TO INVERT
                        650+*                                        *
             00D6   651+@@E765 EQU    @@E764+1                ATTEMPTED MATRIX INVERSION IN
                        652+*                                        * PLACE
             00D7   653+@@E766 EQU    @@E765+1                MATRIX NOT SQUARE
                        654+*                                        *
             00D8   655+@@E767 EQU    @@E766+1                ATTEMPTED MATRIX TRANSPOSITION
                        656+*                                        * IN PLACE
             00D9   657+@@E768 EQU    @@E767+1                SEC FUNCTION ARGUMENT > 1E6
                        658+*                                        *
             00DA   659+@@E769 EQU    @@E768+1                CSC FUNCTION ARGUMENT > 1E6
                        660+*                                        *
             00DB   661+@@E770 EQU    @@E769+1                SIN FUNCTION ARGUMENT > 1E6
                        662+*                                        *
             00DC   663+@@E771 EQU    @@E770+1                COS FUNCTION ARGUMENT > 1E6
                        664+*                                        *
             00DD   665+@@E772 EQU    @@E771+1                TAN FUNCTION ARGUMENT > 1E6
                        666+*                                        *
             00DE   667+@@E773 EQU    @@E772+1                COT FUNCTION ARGUMENT > 1E6
```

```
                                668+*                                          *
                     00DF  669+@@E774 EQU     @@E773+1                 EXPONENTIATION ERROR
                                670+*                                          *
                     00E0  671+@@E775 EQU     @@E774+1                 SOR FUNCTION ARGUMENT < ZERO
                                672+*                                          *
                     00E1  673+@@E776 EQU     @@E775+1                 EXP FUNCTION ARGUMENT > 227.96
                                674+*                                          *
                     00E2  675+@@E777 EQU     @@E776+1                 LOG FUNCTION ARGUMENT 0 OR
                                676+*                                          * NEGATIVE
                     00E3  677+@@E778 EQU     @@E777+1                 LSI FUNCTION ARGUMENT 0 OR
                                678+*                                          * NEGATIVE
                     00E4  679+@@E779 EQU     @@E778+1                 LTW FUNCTION ARGUMENT 0 OR
                                680+*                                          * NEGATIVE
                     00E5  681+@@E780 EQU     @@E779+1                 COT FUNCTION RESULT GOES TO
                                682+*                                          * INFINITY
                     00E6  683+@@E781 EQU     @@E780+1                 SEC FUNCTION RESULT GOES TO
                                684+*                                          * INFINITY
                     00E7  685+@@E782 EQU     @@E781+1                 CSC FUNCTION RESULT GOES TO
                                686+*                                          * INFINITY
                     00E8  687+@@E783 EQU     @@E782+1                 ASN FUNCTION ARG NOT IN RANGE
                                688+*                                          * -1 < X < 1
                     00E9  689+@@E784 EQU     @@E783+1                 ACS FUNCTION ARC NOT IN RANGE
                                690+*                                          * -1 < X < 1
                     00EA  691+@@E785 EQU     @@E784+1                 HSN FUNCTION--ARGUMENT > 225
                                692+*                                          *
                     00EB  693+@@E786 EQU     @@E785+1                 HCS FUNCTION--ARGUMENT > 225
                                694+*                                          *
                     00EC  695+@@E790 EQU     @@E786+1                 DIVISION BY ZERO
                                696+*                                          *
                     00ED  697+@@E791 EQU     @@E790+1                 OVERFLOW - VALUE NOT LESS THAN
                                698+*                                          * 1E99
                     00EE  699+@@E792 EQU     @@E791+1                 UNDERFLOW - VALUE LESS THAN
                                700+*                                          * 1E-99
                     00EF  701+@@E793 EQU     @@E792+1                 TAN FUNCTION ARGUMENT > 100
                                702+*                                          *
                     00F0  703+@@E794 EQU     @@E793+1                 COT FUNCTION ARGUMENT > 100
                                704+*                                          *
                     00F1  705+@@E795 EQU     @@E794+1                 SIN FUNCTION ARGUMENT > 100
                                706+*                                          *
                     00F2  707+@@E796 EQU     @@E795+1                 COS FUNCTION ARGUMENT > 100
                                708+*                                          *
                     00F3  709+@@E797 EQU     @@E796+1                 SEC FUNCTION ARGUMENT > 100
                                710+*                                          *
                     00F4  711+@@E798 EQU     @@E797+1                 CSC FUNCTION ARGUMENT > 100
                                712+*                                          *
                     00F5  713+@@E900 EQU     @@E798+1                 INVALID FUNCTION IN PROCEDURE
                                714+*                                          * STEP
                     00F6  715+@@E901 EQU     @@E900+1                 PROCEDURE ALREADY DEFINED
                                716+*                                          *
                     00F7  717+@@E902 EQU     @@E901+1                 PROCEDURE NOT DEFINED
                                718+*                                          *
                     00F8  719+@@E903 EQU     @@E902+1                 PROCEDURE > 512 CHARACTERS
                                720+*                                          *
                     00F9  721+@@E905 EQU     @@E903+1                 DESK CALCULATOR REQUIRES WITDH
                                722+*                                          * > 63
                     00FA  723+@@E906 EQU     @@E905+1                 INVALID CHARACTER IN PROCEDURE
```

```
                          724+*                                  * DEFINITION
            00FB 725+@@E910 EQU   @@E906+1                 INVALID OPERATION
                          726+*                                  *
            FFFF 727+@@E548 EQU   -1                       PRINTER FAILLQE, OUTPUT
                          728+*                                  *
            FFFF 729+@@E575 EQU   -1                       CHANGED LINE EXCEEDS WIDTH OF
                          730+*                                  *
            FFFF 731+@@E579 EQU   -1                       VTOC FILES EXIST, RE-IPL, USE
                          732+*                                  *
            FFFF 733+@@E580 EQU   -1                       DUPLICATE DISK LABELS -
                          734+*                                  *
            FFFF 735+@@E595 EQU   -1                       INVALID RESPONSE - TYPE ALPHA
                          736+*                                  *
            FFFF 737+@@E597 EQU   -1                       LLLLLL NOT ON UU
                          738+*                                  *
            FFFF 739+@@E598 EQU   -1                       DATA ON ABOVE TRACK
                          740+*                                  *
            FFFF 741+@@E800 EQU   -1                       INVALID INPUT DATA-NUMERIC
                          742+*                                  *
            FFFF 743+@@E801 EQU   -1                       INVALID INPUT DATA--CHARACTER
                          744+*                                  *
            FFFF 745+@@E802 EQU   -1                       TOO MANY INPUT DATA ELEMENTS
                          746+*                                  *
            FFFF 747+@@E803 EQU   -1                       NOT ENOUGH DATA ELEMENTS
                          748+*                                  *
            FFFF 749+@@E804 EQU   -1                       NOT ENOUGH ARRAY ROW ELEMENTS
                          750+*                                  *
            0000 751+@@E001 EQU   0                        MISSING <ARITHMETIC
                          752+*                                  * EXPRESSION>
            0001 753+@@E003 EQU   @@E001+1                 UNBALANCED <PARENTHESES>
                          754+*                                  *
            0002 755+@@E004 EQU   @@E003+1                 <ARITHMETIC CONSTANT> CONTAINS
                          756+*                                  * 2 DECIMAL POINTS
            0003 757+@@E005 EQU   @@E004+1                 DECIMAL POINT WITHOUT
                          758+*                                  * <ARITHMETIC CONSTANT>
            0004 759+@@E006 EQU   @@E005+1                 INCOMPLETE <ARITHMETIC
                          760+*                                  * EXPRESSION>
            0005 761+@@E007 EQU   @@E006+1                 INVALID CHARACTER FOLLOWING
                          762+*                                  * <OPERATOR>
            0006 763+@@E008 EQU   @@E007+1                 <CHARACTER VARIABLE> IN
                          764+*                                  * <ARITHMETIC EXPRESSION>
            0007 765+@@E009 EQU   @@E008+1                 INVALID EXPRESSION FIRST
                          766+*                                  * CHARACTER
            0008 767+@@E010 EQU   @@E009+1                 INVALID <SECONDARY KEYWORD>
                          768+*                                  *
            0009 769+@@E011 EQU   @@E010+1                 COMMA NOT FOLLOWING LINE
                          770+*                                  * NUMBER
            000A 771+@@E012 EQU   @@E011+1                 INVALID <DELIMITER>
                          772+*                                  *
            000B 773+@@E013 EQU   @@E012+1                 INCOMPLETE <CHARACTER
                          774+*                                  * CONSTANT>
            000C 775+@@E014 EQU   @@E013+1                 INVALID FILE SPECIFICATION
                          776+*                                  *
            000D 777+@@E015 EQU   @@E014+1                 VARIABLE NOT PRESENT IN INPUT
                          778+*                                  * LIST
            000E 779+@@E016 EQU   @@E015+1                 INVALID VARIABLE
```

```
                         780+*                                   *
              000F  781+@@E017 EQU    @@E016+1             INVALID EXPONENT IN CONSTANT
                         782+*                                   *
              0010  783+@@E018 EQU    @@E017+1             <OPERATOR> WITHOUT VALID
                         784+*                             * PRECEDING OPERAND
              0011  785+@@E019 EQU    @@E018+1             <OPERATOR> REQUIRED BTWN LAST
                         786+*                             * 2 CHARACTERS CHECKED
              0012  787+@@E020 EQU    @@E019+1             INVALID CONSTANT
                         788+*                                   *
              0013  789+@@E021 EQU    @@E020+1             <LINE NUMBER> TOO LONG
                         790+*                                   *
              0014  791+@@E023 EQU    @@E021+1             INVALID <SYSTEM CONSTANT>
                         792+*                                   *
              0015  793+@@E024 EQU    @@E023+1             INVALID OR MISSING <LINE
                         794+*                             * NUMBER>
              0016  795+@@E025 EQU    @@E024+1             INVALID <PRIMARY KEYWORD>
                         796+*                                   *
              0017  797+@@E026 EQU    @@E025+1             NO EQUAL SIGN AFTER
                         798+*                             * <ARITHMETIC VARIABLE>
              0018  799+@@E027 EQU    @@E026+1             INVALID SIMPLE <ARITHMETIC
                         800+*                             * VARIABLE>
              0019  801+@@E028 EQU    @@E027+1             INVALID <CONTROL VARIABLE>
                         802+*                             * CHARACTER
              001A  803+@@E029 EQU    @@E028+1             MISSING <RELATIONAL OPERATOR>
                         804+*                                   *
              001B  805+@@E030 EQU    @@E029+1             INVALID OR MISSING <CHARACTER
                         806+*                             * EXPRESSION>
              001C  807+@@E031 EQU    @@E030+1             INVALID <DEF> FUNCTION
                         808+*                             * DEFINITION
              001D  809+@@E032 EQU    @@E031+1             NO EQUAL SIGN AFTER VALID
                         810+*                             * FUNCTION DEFINITION
              001E  811+@@E035 EQU    @@E032+1             INVALID CHARACTER AFTER VALID
                         812+*                             * STATEMENT
              001F  813+@@E036 EQU    @@E035+1             VARIABLE IS NOT FOLLOWED BY A
                         814+*                             * COMMA OR EQUAL SIGN
              0020  815+@@E037 EQU    @@E036+1             CHARACTER AND ARITHMETIC
                         816+*                             * <VARIABLES> INTERmIxED
              0021  817+@@E038 EQU    @@E037+1             INVALID <CHARACTER VARIABLE>
                         818+*                                   *
              0022  819+@@E039 EQU    @@E038+1             INVALID <ARRAY NAME>
                         820+*                                   *
              0023  821+@@E040 EQU    @@E039+1             INVALID DIMENSION
                         822+*                                   *
              0024  823+@@E041 EQU    @@E040+1             INVALID <DELIMITER> AFTER
                         824+*                             * VALID ARRAY DEFINITION
              0025  825+@@E042 EQU    @@E041+1             INVALID MATRIX EXPRESSION ON
                         826+*                             * RIGHT OF EQUAL SIGN
              0026  827+@@E043 EQU    @@E042+1             INVALID <mATRIX> NAME,
                         828+*                                   *
              0027  829+@@E044 EQU    @@E043+1             MISSING MULTIPLICATION
                         830+*                             * <OPERATOR>
              0028  831+@@E045 EQU    @@E044+1             STATEMENT TERMINATED
                         832+*                             * PREMATURELY
              0029  833+@@E046 EQU    @@E045+1             <ARITHMETIC CONSTANT> NOT IN
                         834+*                             * RANGE 1E-99 < X < 1E99
              002A  835+@@E060 EQU    @@E046+1             EXPRESSION TOO COMPLEX
```

```
      @ERMEQ - GENERAL ERROR MESSAGE EQUATES
```

```
                              836+*                                 *
                         002B 837+@@E080 EQU   @@E060+1             DATA FILE LINE TOO LONG
                              838+*       END OF ERROR MESSAGES EQUATES
                              839+       PRINT ON
                              840 *      @SPF  EXP-Y
                              842+       PRINT ON
```

```
     @SPFEQ - SYSTEM PROGRAM FILE EQUATES
```

```
                      844+*********************************************************************
                      845+*       SYSTEM PROGRAM FILE (SPF) EQUATES                            *
                      846+*********************************************************************
                      847+*
          0000        848+#$#0TR EQU   X'0000'                   DISK ADDR OF ##0TRK
          0700        849+#$$#0T EQU   X'0700'                   CORE LOAD ADDRESS OF ##0TRK
          0018        850+#$@#0T EQU   24                        SECTOR COUNT OF ##0TRK
                      851+*
          0080        852+#$#1TR EQU   X'0080'                   DISK ADDR OF ##1TRK
          0000        853+#$$#1T EQU   X'0000'                   CORE LOAD ADDRESS OF ##1TRK
          0018        854+#$@#1T EQU   24                        SECTOR COUNT OF ##1TRK
                      855+*
          0000        856+#$#DRT EQU   X'0000'                   DISK ADDR OF ##DRTY
          0000        857+#$$#DR EQU   X'0000'                   CORE LOAD ADDRESS OF ##DRTY
          0008        858+#$@#DR EQU   08                        SECTOR COUNT OF ##DRTY
                      859+*
          0020        860+#$INST EQU   X'0020'                   DISK ADDR OF #INSTD
          0600        861+#$$INS EQU   X'0600'                   CORE LOAD ADDRESS OF #INSTD
          0010        862+#$@INS EQU   16                        SECTOR COUNT OF #INSTD
                      863+*
          0080        864+#$BCOM EQU   X'0080'                   DISK ADDR OF #BCOMP
          0600        865+#$$BCO EQU   X'0600'                   CORE LOAD ADDRESS OF #BCOMP
          0018        866+#$@BCO EQU   24                        SECTOR COUNT OF #BCOMP
                      867+*
          0100        868+#$LOAD EQU   X'0100'                   DISK ADDR OF #LOADR
          0600        869+#$$LOA EQU   X'0600'                   CORE LOAD ADDRESS OF #LOADR
          0013        870+#$@LOA EQU   19                        SECTOR COUNT OF #LOADR
                      871+*
          014C        872+#$DPRI EQU   X'014C'                   DISK ADDR OF #DPRIN
          0700        873+#$$DPR EQU   X'0700'                   CORE LOAD ADDRESS OF #DPRIN
          0005        874+#$@DPR EQU   05                        SECTOR COUNT OF #DPRIN
                      875+*
          0180        876+#$KGOS EQU   X'0180'                   DISK ADDR OF #KGOSL
          0C00        877+#$$KGO EQU   X'0C00'                   CORE LOAD ADDRESS OF #KGOSL
          0002        878+#$@KGO EQU   02                        SECTOR COUNT OF #KGOSL
                      879+*
          0188        880+#$KEDI EQU   X'0188'                   DISK ADDR OF #KEDIT
          0C00        881+#$$KED EQU   X'0C00'                   CORE LOAD ADDRESS OF #KEDIT
          000E        882+#$@KED EQU   14                        SECTOR COUNT OF #KEDIT
                      883+*
          01C4        884+#$KENA EQU   X'01C4'                   DISK ADDR OF #KENAB
          0C00        885+#$$KEN EQU   X'0C00'                   CORE LOAD ADDRESS OF #KENAB
          0006        886+#$@KEN EQU   06                        SECTOR COUNT OF #KENAB
                      887+*
          0200        888+#$DREA EQU   X'0200'                   DISK ADDR OF #DREAD
          0889        889+#$$DRE EQU   X'0889'                   CORE LOAD ADDRESS OF #DREAD
          0001        890+#$@DRE EQU   01                        SECTOR COUNT OF #DREAD
                      891+*
          0204        892+#$KMOU EQU   X'0204'                   DISK ADDR OF #KMOUN
          0C00        893+#$$KMO EQU   X'0C00'                   CORE LOAD ADDRESS OF #KMOUN
          0004        894+#$@KMO EQU   04                        SECTOR COUNT OF #KMOUN
                      895+*
          0214        896+#$KRMO EQU   X'0214'                   DISK ADDR OF #KRMOV
          0C00        897+#$$KRM EQU   X'0C00'                   CORE LOAD ADDRESS OF #KRMOV
          0003        898+#$@KRM EQU   03                        SECTOR COUNT OF #KRMOV
                      899+*
```

```
                          0220  900+#$KPAS EQU   X'0220'                 DISK ADDR OF #KPASW
                          0C00  901+#$$KPA EQU   X'0C00'                 CORE LOAD ADDRESS OF #KPASW
                          0005  902+#$@KPA EQU   05                      SECTOR COUNT OF #KPASW
                                903+*
                          0234  904+#$KEXT EQU   X'0234'                 DISK ADDR OF #KEXTR
                          0C00  905+#$$KEX EQU   X'0C00'                 CORE LOAD ADDRESS OF #KEXTR
                          0003  906+#$@KEX EQU   03                      SECTOR COUNT OF #KEXTR
                                907+*
                          0240  908+#$DSPL EQU   X'0240'                 DISK ADDR OF #DSPLY
                          2800  909+#$$DSP EQU   X'2800'                 CORE LOAD ADDRESS OF #DSPLY
                          0004  910+#$@DSP EQU   04                      SECTOR COUNT OF #DSPLY
                                911+*
                          0250  912+#$TSYK EQU   X'0250'                 DISK ADDR OF #TSYKT
                          1000  913+#$$TSY EQU   X'1000'                 CORE LOAD ADDRESS OF #TSYKT
                          0003  914+#$@TSY EQU   03                      SECTOR COUNT OF #TSYKT
                                915+*
                          0280  916+#$KRNU EQU   X'0280'                 DISK ADDR OF #KRNUM
                          0700  917+#$$KRN EQU   X'0700'                 CORE LOAD ADDRESS OF #KRNUM
                          0003  918+#$@KRN EQU   03                      SECTOR COUNT OF #KRNUM
                                919+*
                          028C  920+#$KROV EQU   X'028C'                 DISK ADDR OF #KROVL
                          0D00  921+#$$KRO EQU   X'0D00'                 CORE LOAD ADDRESS OF #KROVL
                          000A  922+#$@KRO EQU   10                      SECTOR COUNT OF #KROVL
                                923+*
                          0290  924+#$KOVM EQU   X'0290'                 DISK ADDR OF #KOVME
                          0E00  925+#$$KOV EQU   X'0E00'                 CORE LOAD ADDRESS OF #KOVME
                          0009  926+#$@KOV EQU   09                      SECTOR COUNT OF #KOVME
                                927+*
                          02B4  928+#$KWRI EQU   X'02B4'                 DISK ADDR OF #KWRIT
                          0C00  929+#$$KWR EQU   X'0C00'                 CORE LOAD ADDRESS OF #KWRIT
                          0002  930+#$@KWR EQU   02                      SECTOR COUNT OF #KWRIT
                                931+*
                          02BC  932+#$KREA EQU   X'02BC'                 DISK ADDR OF #KREAD
                          0C00  933+#$$KRE EQU   X'0C00'                 CORE LOAD ADDRESS OF #KREAD
                          0002  934+#$@KRE EQU   02                      SECTOR COUNT OF #KREAD
                                935+*
                          02C4  936+#$KWID EQU   X'02C4'                 DISK ADDR OF #KWIDT
                          0C00  937+#$$KWI EQU   X'0C00'                 CORE LOAD ADDRESS OF #KWIDT
                          0002  938+#$@KWI EQU   02                      SECTOR COUNT OF #KWIDT
                                939+*
                          02CC  940+#$KRUN EQU   X'02CC'                 DISK ADDR OF #KRUNI
                          0C00  941+#$$KRU EQU   X'0C00'                 CORE LOAD ADDRESS OF #KRUNI
                          0003  942+#$@KRU EQU   03                      SECTOR COUNT OF #KRUNI
                                943+*
                          0300  944+#$KDNT EQU   X'0300'                 DISK ADDR OF #KDNTE
                          0C00  945+#$$KDN EQU   X'0C00'                 CORE LOAD ADDRESS OF #KDNTE
                          0010  946+#$@KDN EQU   16                      SECTOR COUNT OF #KDNTE
                                947+*
                          030C  948+#$KMER EQU   X'030C'                 DISK ADDR OF #KMERG
                          0D00  949+#$$KME EQU   X'0D00'                 CORE LOAD ADDRESS OF #KMERG
                          0003  950+#$@KME EQU   03                      SECTOR COUNT OF #KMERG
                                951+*
                          0350  952+#$TDCK EQU   X'0350'                 DISK ADDR OF #TDCKT
                          1000  953+#$$TDC EQU   X'1000'                 CORE LOAD ADDRESS OF #TDCKT
                          0003  954+#$@TDC EQU   03                      SECTOR COUNT OF #TDCKT
                                955+*
```

```
                          035C  956+#$KDEL EQU   X'035C'                 DISK ADDR OF #KDELE
                          0C00  957+#$$KDE EQU   X'0C00'                 CORE LOAD ADDRESS OF #KDELE
                          0010  958+#$@KDE EQU   16                      SECTOR COUNT OF #KDELE
                                959+*
                          03BC  960+#$KCTL EQU   X'03BC'                 DISK ADDR OF #KCTL0
                          0C00  961+#$$KCT EQU   X'0C00'                 CORE LOAD ADDRESS OF #KCTL0
                          0009  962+#$@KCT EQU   09                      SECTOR COUNT OF #KCTL0
                                963+*
                          0400  964+#$KLIS EQU   X'0400'                 DISK ADDR OF #KLIST
                          0C00  965+#$$KLI EQU   X'0C00'                 CORE LOAD ADDRESS OF #KLIST
                          0011  966+#$@KLI EQU   17                      SECTOR COUNT OF #KLIST
                                967+*
                          0444  968+#$KLOG EQU   X'0444'                 DISK ADDR OF #KLOGO
                          0C00  969+#$$KLO EQU   X'0C00'                 CORE LOAD ADDRESS OF #KLOGO
                          0008  970+#$@KLO EQU   08                      SECTOR COUNT OF #KLOGO
                                971+*
                          0484  972+#$SPSY EQU   X'0484'                 DISK ADDR OF #SPSYN
                          0C00  973+#$$SPS EQU   X'0C00'                 CORE LOAD ADDRESS OF #SPSYN
                          0001  974+#$@SPS EQU   01                      SECTOR COUNT OF #SPSYN
                                975+*
                          0488  976+#$KSAV EQU   X'0488'                 DISK ADDR OF #KSAVE
                          0C00  977+#$$KSA EQU   X'0C00'                 CORE LOAD ADDRESS OF #KSAVE
                          0011  978+#$@KSA EQU   17                      SECTOR COUNT OF #KSAVE
                                979+*
                          04CC  980+#$SPAC EQU   X'04CC'                 DISK ADDR OF #SPACK
                          0C00  981+#$$SPA EQU   X'0C00'                 CORE LOAD ADDRESS OF #SPACK
                          0004  982+#$@SPA EQU   04                      SECTOR COUNT OF #SPACK
                                983+*
                          04DC  984+#$SPOV EQU   X'04DC'                 DISK ADDR OF #SPOVL
                          0806  985+#$$SPO EQU   X'0806'                 CORE LOAD ADDRESS OF #SPOVL
                          0003  986+#$@SPO EQU   03                      SECTOR COUNT OF #SPOVL
                                987+*
                          0508  988+#$KPOO EQU   X'0508'                 DISK ADDR OF #KPOOL
                          0C00  989+#$$KPO EQU   X'0C00'                 CORE LOAD ADDRESS OF #KPOOL
                          000D  990+#$@KPO EQU   13                      SECTOR COUNT OF #KPOOL
                                991+*
                          053C  992+#$KCHA EQU   X'053C'                 DISK ADDR OF #KCHAN
                          0C00  993+#$$KCH EQU   X'0C00'                 CORE LOAD ADDRESS OF #KCHAN
                          000C  994+#$@KCH EQU   12                      SECTOR COUNT OF #KCHAN
                                995+*
                          058C  996+#$KSVL EQU   X'058C'                 DISK ADDR OF #KSVLA
                          0980  997+#$$KSV EQU   X'0980'                 CORE LOAD ADDRESS OF #KSVLA
                          0002  998+#$@KSV EQU   02                      SECTOR COUNT OF #KSVLA
                                999+*
                          0594 1000+#$KSSP EQU   X'0594'                 DISK ADDR OF #KSSPN
                          0C00 1001+#$$KSS EQU   X'0C00'                 CORE LOAD ADDRESS OF #KSSPN
                          000B 1002+#$@KSS EQU   11                      SECTOR COUNT OF #KSSPN
                               1003+*
                          05C0 1004+#$KNAM EQU   X'05C0'                 DISK ADDR OF #KNAME
                          0C00 1005+#$$KNA EQU   X'0C00'                 CORE LOAD ADDRESS OF #KNAME
                          0008 1006+#$@KNA EQU   08                      SECTOR COUNT OF #KNAME
                               1007+*
                          0600 1008+#$KSYM EQU   X'0600'                 DISK ADDR OF #KSYMB
                          0C00 1009+#$$KSY EQU   X'0C00'                 CORE LOAD ADDRESS OF #KSYMB
                          000F 1010+#$@KSY EQU   15                      SECTOR COUNT OF #KSYMB
                               1011+*
```

```
                        063C 1012+#$KPRT EQU   X'063C'                 DISK ADDR OF #KPRTC
                        0C00 1013+#$$KPR EQU   X'0C00'                 CORE LOAD ADDRESS OF #KPRTC
                        0009 1014+#$@KPR EQU   09                      SECTOR COUNT OF #KPRTC
                             1015+*
                        0680 1016+#$KSET EQU   X'0680'                 DISK ADDR OF #KSETI
                        0E00 1017+#$$KSE EQU   X'0E00'                 CORE LOAD ADDRESS OF #KSETI
                        0004 1018+#$@KSE EQU   04                      SECTOR COUNT OF #KSETI
                             1019+*
                        0690 1020+#$GRAP EQU   X'0690'                 DISK ADDR OF #GRAPR
                        0889 1021+#$$GRA EQU   X'0889'                 CORE LOAD ADDRESS OF #GRAPR
                        0003 1022+#$@GRA EQU   03                      SECTOR COUNT OF #GRAPR
                             1023+*
                        06A4 1024+#$KALL EQU   X'06A4'                 DISK ADDR OF #KALLO
                        0C00 1025+#$$KAL EQU   X'0C00'                 CORE LOAD ADDRESS OF #KALLO
                        000F 1026+#$@KAL EQU   15                      SECTOR COUNT OF #KALLO
                             1027+*
                        0700 1028+#$KRLA EQU   X'0700'                 DISK ADDR OF #KRLAB
                        0700 1029+#$$KRL EQU   X'0700'                 CORE LOAD ADDRESS OF #KRLAB
                        0004 1030+#$@KRL EQU   04                      SECTOR COUNT OF #KRLAB
                             1031+*
                        0710 1032+#$KRVL EQU   X'0710'                 DISK ADDR OF #KRVLA
                        0800 1033+#$$KRV EQU   X'0800'                 CORE LOAD ADDRESS OF #KRVLA
                        000D 1034+#$@KRV EQU   13                      SECTOR COUNT OF #KRVLA
                             1035+*
                        0744 1036+#$KDIS EQU   X'0744'                 DISK ADDR OF #KDISP
                        0D00 1037+#$$KDI EQU   X'0D00'                 CORE LOAD ADDRESS OF #KDISP
                        0005 1038+#$@KDI EQU   05                      SECTOR COUNT OF #KDISP
                             1039+*
                        0780 1040+#$KDOV EQU   X'0780'                 DISK ADDR OF #KDOVR
                        0E00 1041+#$$KDO EQU   X'0E00'                 CORE LOAD ADDRESS OF #KDOVR
                        000C 1042+#$@KDO EQU   12                      SECTOR COUNT OF #KDOVR
                             1043+*
                        07B4 1044+#$VCRT EQU   X'07B4'                 DISK ADDR OF #VCRTI
                        2000 1045+#$$VCR EQU   X'2000'                 CORE LOAD ADDRESS OF #VCRTI
                        0008 1046+#$@VCR EQU   08                      SECTOR COUNT OF #VCRTI
                             1047+*
                        07D4 1048+#$EXMS EQU   X'07D4'                 DISK ADDR OF #EXMSG
                        0C00 1049+#$$EXM EQU   X'0C00'                 CORE LOAD ADDRESS OF #EXMSG
                        0003 1050+#$@EXM EQU   03                      SECTOR COUNT OF #EXMSG
                             1051+*
                        0800 1052+#$#COR EQU   X'0800'                 DISK ADDR OF ##CORE
                        0000 1053+#$$#CO EQU   X'0000'                 CORE LOAD ADDRESS OF ##CORE
                        003A 1054+#$@#CO EQU   58                      SECTOR COUNT OF ##CORE
                             1055+*
                        0928 1056+#$#ERM EQU   X'0928'                 DISK ADDR OF ##ERMS
                        0000 1057+#$$#ER EQU   X'0000'                 CORE LOAD ADDRESS OF ##ERMS
                        0032 1058+#$@#ER EQU   50                      SECTOR COUNT OF ##ERMS
                             1059+*
                        0A30 1060+#$KHEL EQU   X'0A30'                 DISK ADDR OF #KHELP
                        0C00 1061+#$$KHE EQU   X'0C00'                 CORE LOAD ADDRESS OF #KHELP
                        000C 1062+#$@KHE EQU   12                      SECTOR COUNT OF #KHELP
                             1063+*
                        0A80 1064+#$MIPP EQU   X'0A80'                 DISK ADDR OF #MIPPE
                        0C00 1065+#$$MIP EQU   X'0C00'                 CORE LOAD ADDRESS OF #MIPPE
                        000D 1066+#$@MIP EQU   13                      SECTOR COUNT OF #MIPPE
                             1067+*
```

```
                             0AC8 1068+#$KSOV EQU    X'0AC8'                   DISK ADDR OF #KSOVR
                             0C20 1069+#$$KSO EQU    X'0C20'                   CORE LOAD ADDRESS OF #KSOVR
                             000D 1070+#$@KSO EQU    13                        SECTOR COUNT OF #KSOVR
                                  1071+*
                             0B00 1072+#$VXIT EQU    X'0B00'                   DISK ADDR OF #VXITI
                             0600 1073+#$$VXI EQU    X'0600'                   CORE LOAD ADDRESS OF #VXITI
                             0002 1074+#$@VXI EQU    02                        SECTOR COUNT OF #VXITI
                                  1075+*
                             0B08 1076+#$#VUF EQU    X'0B08'                   DISK ADDR OF ##VUFA
                             0600 1077+#$$#VU EQU    X'0600'                   CORE LOAD ADDRESS OF ##VUFA
                             0002 1078+#$@#VU EQU    02                        SECTOR COUNT OF ##VUFA
                                  1079+*
                             0B80 1080+#$VLOA EQU    X'0B80'                   DISK ADDR OF #VLOAD
                             0600 1081+#$$VLO EQU    X'0600'                   CORE LOAD ADDRESS OF #VLOAD
                             0002 1082+#$@VLO EQU    02                        SECTOR COUNT OF #VLOAD
                                  1083+*
                             0B88 1084+#$VODK EQU    X'0B88'                   DISK ADDR OF #VODKA
                             0600 1085+#$$VOD EQU    X'0600'                   CORE LOAD ADDRESS OF #VODKA
                             0016 1086+#$@VOD EQU    22                        SECTOR COUNT OF #VODKA
                                  1087+*
                             0BAC 1088+#$TVKB EQU    X'0BAC'                   DISK ADDR OF #TVKBT
                             0FC0 1089+#$$TVK EQU    X'0FC0'                   CORE LOAD ADDRESS OF #TVKBT
                             0001 1090+#$@TVK EQU    01                        SECTOR COUNT OF #TVKBT
                                  1091+*
                             0C00 1092+#$VVMR EQU    X'0C00'                   DISK ADDR OF #VVMRS
                             0000 1093+#$$VVM EQU    X'0000'                   CORE LOAD ADDRESS OF #VVMRS
                             0030 1094+#$@VVM EQU    48                        SECTOR COUNT OF #VVMRS
                                  1095+*
                             0D00 1096+#$FMST EQU    X'0D00'                   DISK ADDR OF #FMSTD
                             0200 1097+#$$FMS EQU    X'0200'                   CORE LOAD ADDRESS OF #FMSTD
                             0052 1098+#$@FMS EQU    82                        SECTOR COUNT OF #FMSTD
                                  1099+*
                             0EA8 1100+#$UEXL EQU    X'0EA8'                   DISK ADDR OF #UEXLI
                             0C00 1101+#$$UEX EQU    X'0C00'                   CORE LOAD ADDRESS OF #UEXLI
                             000E 1102+#$@UEX EQU    14                        SECTOR COUNT OF #UEXLI
                                  1103+*
                             0F00 1104+#$UALL EQU    X'0F00'                   DISK ADDR OF #UALLO
                             0C00 1105+#$$UAL EQU    X'0C00'                   CORE LOAD ADDRESS OF #UALLO
                             0011 1106+#$@UAL EQU    17                        SECTOR COUNT OF #UALLO
                                  1107+*
                             0F80 1108+#$KCND EQU    X'0F80'                   DISK ADDR OF #KCNDI
                             0C00 1109+#$$KCN EQU    X'0C00'                   CORE LOAD ADDRESS OF #KCNDI
                             0010 1110+#$@KCN EQU    16                        SECTOR COUNT OF #KCNDI
                                  1111+*
                             1000 1112+#$#CSA EQU    X'1000'                   DISK ADDR OF #CSAV
                             0000 1113+#$$#CS EQU    X'0000'                   CORE LOAD ADDRESS OF #CSAV
                             003A 1114+#$@#CS EQU    58                        SECTOR COUNT OF #CSAV
                                  1115+*
                             1128 1116+#$#SSA EQU    X'1128'                   DISK ADDR OF #SSAV
                             0000 1117+#$$#SS EQU    X'0000'                   CORE LOAD ADDRESS OF #SSAV
                             0001 1118+#$@#SS EQU    01                        SECTOR COUNT OF #SSAV
                                  1119+*
                             1180 1120+#$#SAV EQU    X'1180'                   DISK ADDR OF ##SAVM
                             0000 1121+#$$#SA EQU    X'0000'                   CORE LOAD ADDRESS OF ##SAVM
                             0108 1122+#$@#SA EQU    264                       SECTOR COUNT OF ##SAVM
                                  1123+*
```

```
                          1700 1124+#$FIST EQU    X'1700'                    DISK ADDR OF #FISTD
                          0E00 1125+#$$FIS EQU    X'0E00'                    CORE LOAD ADDRESS OF #FISTD
                          0009 1126+#$@FIS EQU    09                         SECTOR COUNT OF #FISTD
                               1127+*
                          1724 1128+#$FILN EQU    X'1724'                    DISK ADDR OF #FILNG
                          0E00 1129+#$$FIL EQU    X'0E00'                    CORE LOAD ADDRESS OF #FILNG
                          0009 1130+#$@FIL EQU    09                         SECTOR COUNT OF #FILNG
                               1131+*
                          1780 1132+#$#RSP EQU    X'1780'                    DISK ADDR OF ##RSPG
                          0000 1133+#$$#RS EQU    X'0000'                    CORE LOAD ADDRESS OF ##RSPG
                          0030 1134+#$@#RS EQU    48                         SECTOR COUNT OF ##RSPG
                               1135+*
                          1780 1136+#$BOLV EQU    X'1780'                    DISK ADDR OF #BOVLY
                          0800 1137+#$$BOV EQU    X'0800'                    CORE LOAD ADDRESS OF #BOVLY
                          0018 1138+#$@BOV EQU    24                         SECTOR COUNT OF #BOLVY
                               1139+*
                          1800 1140+#$SFSY EQU    X'1800'                    DISK ADDR OF #SFSYN
                          0C00 1141+#$$SFS EQU    X'0C00'                    CORE LOAD ADDRESS OF #SFSYN
                          0011 1142+#$@SFS EQU    17                         SECTOR COUNT OF #SFSYN
                               1143+*
                          1844 1144+#$SFOV EQU    X'1844'                    DISK ADDR OF #SFOVR
                          1500 1145+#$$SFO EQU    X'1500'                    CORE LOAD ADDRESS OF #SFOVR
                          0003 1146+#$@SFO EQU    03                         SECTOR COUNT OF #SFOVR
                               1147+*
                          1850 1148+#$STRO EQU    X'1850'                    DISK ADDR OF #STROV
                          1600 1149+#$$STR EQU    X'1600'                    CORE LOAD ADDRESS OF #STROV
                          0002 1150+#$@STR EQU    02                         SECTOR COUNT OF #STROV
                               1151+*
                          1880 1152+#$#FSP EQU    X'1880'                    DISK ADDR OF ##FSPG
                          0000 1153+#$$#FS EQU    X'0000'                    CORE LOAD ADDRESS OF ##FSPG
                          0030 1154+#$@#FS EQU    48                         SECTOR COUNT OF ##FSPG
                               1155+*
                          1880 1156+#$GUFU EQU    X'1880'                    DISK ADDR OF #GUFUD
                          0C00 1157+#$$GUF EQU    X'0C00'                    CORE LOAD ADDRESS OF #GUFUD
                          0010 1158+#$@GUF EQU    16                         SECTOR COUNT OF #GUFUD
                               1159+*
                          18C0 1160+#$ERRP EQU    X'18C0'                    DISK ADDR OF #ERRPG
                          0C00 1161+#$$ERR EQU    X'0C00'                    CORE LOAD ADDRESS OF #ERRPG
                          0003 1162+#$@ERR EQU    03                         SECTOR COUNT OF #ERRPG
                               1163+*
                          18D4 1164+#$#BLN EQU    X'18D4'                    DISK ADDR OF ##BLNB
                          0000 1165+#$$#BL EQU    X'0000'                    CORE LOAD ADDRESS OF ##BLNB
                          0001 1166+#$@#BL EQU    01                         SECTOR COUNT OF ##BLNB
                               1167+*
                          1900 1168+#$ECMA EQU    X'1900'                    DISK ADDR OF #ECMAN
                          0C00 1169+#$$ECM EQU    X'0C00'                    CORE LOAD ADDRESS OF #ECMAN
                          0006 1170+#$@ECM EQU    06                         SECTOR COUNT OF #ECMAN
                               1171+*
                          1918 1172+#$SFLO EQU    X'1918'                    DISK ADDR OF #SFLOA
                          0F00 1173+#$$SFL EQU    X'0F00'                    CORE LOAD ADDRESS OF #SFLOA
                          0005 1174+#$@SFL EQU    05                         SECTOR COUNT OF #SFLOA
                               1175+*
                          192C 1176+#$SDSY EQU    X'192C'                    DISK ADDR OF #SDSYN
                          0C00 1177+#$$SDS EQU    X'0C00'                    CORE LOAD ADDRESS OF #SDSYN
                          0004 1178+#$@SDS EQU    04                         SECTOR COUNT OF #SDSYN
                               1179+*
```

```
                       193C 1180+#$SFFI EQU   X'193C'                  DISK ADDR OF #SFFIN
                       0E00 1181+#$$SFF EQU   X'0E00'                  CORE LOAD ADDRESS OF #SFFIN
                       0008 1182+#$@SFF EQU   08                       SECTOR COUNT OF #SFFIN
                            1183+*
                       1980 1184+#$UPAC EQU   X'1980'                  DISK ADDR OF #UPACK
                       0C00 1185+#$$UPA EQU   X'0C00'                  CORE LOAD ADDRESS OF #UPACK
                       0004 1186+#$@UPA EQU   04                       SECTOR COUNT OF #UPACK
                            1187+*
                       1990 1188+#$EFKE EQU   X'1990'                  DISK ADDR OF #EFKEY
                       0C00 1189+#$$EFK EQU   X'0C00'                  CORE LOAD ADDRESS OF #EFKEY
                       0002 1190+#$@EFK EQU   02                       SECTOR COUNT OF #EFKEY
                            1191+*
                       19B8 1192+#$UCNF EQU   X'19B8'                  DISK ADDR OF #UCNFI
                       0C00 1193+#$$UCN EQU   X'0C00'                  CORE LOAD ADDRESS OF #UCNFI
                       0009 1194+#$@UCN EQU   09                       SECTOR COUNT OF #UCNFI
                            1195+*
                       19DC 1196+#$UCPL EQU   X'19DC'                  DISK ADDR OF #UCPLI
                       0700 1197+#$$UCP EQU   X'0700'                  CORE LOAD ADDRESS OF #UCPLI
                       000F 1198+#$@UCP EQU   15                       SECTOR COUNT OF #UCPLI
                            1199+*
                       1A38 1200+#$UATR EQU   X'1A38'                  DISK ADDR OF #UATRC
                       0900 1201+#$$UAT EQU   X'0900'                  CORE LOAD ADDRESS OF #UATRC
                       000C 1202+#$@UAT EQU   12                       SECTOR COUNT OF #UATRC
                            1203+*
                       1A88 1204+#$UINI EQU   X'1A88'                  DISK ADDR OF #UINIT
                       0C00 1205+#$$UIN EQU   X'0C00'                  CORE LOAD ADDRESS OF #UINIT
                       000F 1206+#$@UIN EQU   15                       SECTOR COUNT OF #UINIT
                            1207+*
                       1AD8 1208+#$UCDI EQU   X'1AD8'                  DISK ADDR OF #UCDIS
                       0900 1209+#$$UCD EQU   X'0900'                  CORE LOAD ADDRESS OF #UCDIS
                       000B 1210+#$@UCD EQU   11                       SECTOR COUNT OF #UCDIS
                            1211+*
                       1B24 1212+#$UDEL EQU   X'1B24'                  DISK ADDR OF #UDELV
                       0C00 1213+#$$UDE EQU   X'0C00'                  CORE LOAD ADDRESS OF #UDELV
                       000E 1214+#$@UDE EQU   14                       SECTOR COUNT OF #UDELV
                            1215+*
                       1B5C 1216+#$UDIS EQU   X'1B5C'                  DISK ADDR OF #UDISV
                       0C00 1217+#$$UDI EQU   X'0C00'                  CORE LOAD ADDRESS OF #UDISV
                       0008 1218+#$@UDI EQU   08                       SECTOR COUNT OF #UDISV
                            1219+*
                       1B9C 1220+#$ZTRA EQU   X'1B9C'                  DISK ADDR OF #ZTRAC
                       1000 1221+#$$ZTR EQU   X'1000'                  CORE LOAD ADDRESS OF #ZTRAC
                       0001 1222+#$@ZTR EQU   01                       SECTOR COUNT OF #ZTRAC
                            1223+*
                       1BA4 1224+#$ZDUM EQU   X'1BA4'                  DISK ADDR OF #ZDUMP
                       1100 1225+#$$ZDU EQU   X'1100'                  CORE LOAD ADDRESS OF #ZDUMP
                       0008 1226+#$@ZDU EQU   08                       SECTOR COUNT OF #ZDUMP
                            1227+*
                       1BC4 1228+#$ZLOA EQU   X'1BC4'                  DISK ADDR OF #ZLOAD
                       1100 1229+#$$ZLO EQU   X'1100'                  CORE LOAD ADDRESS OF #ZLOAD
                       000C 1230+#$@ZLO EQU   12                       SECTOR COUNT OF #ZLOAD
                            1231+*
                       1C14 1232+#$ZUTM EQU   X'1C14'                  DISK ADDR OF #ZUTMO
                       0C00 1233+#$$ZUT EQU   X'0C00'                  CORE LOAD ADDRESS OF #ZUTMO
                       0014 1234+#$@ZUT EQU   20                       SECTOR COUNT OF #ZUTMO
                            1235+*
```

@SPFEQ - SYSTEM PROGRAM FILE EQUATES

```
                        1C84 1236+#$INLN EQU   X'1C84'                 DISK ADDR OF #INLNG
                        0600 1237+#$$INL EQU   X'0600'                 CORE LOAD ADDRESS OF #INLNG
                        0010 1238+#$@INL EQU   16                      SECTOR COUNT OF #INLNG
                        1239+*
                        1CC4 1240+#$KCAL EQU   X'1CC4'                 DISK ADDR OF #KCALL
                        0C00 1241+#$$KCA EQU   X'0C00'                 CORE LOAD ADDRESS OF #KCALL
                        000C 1242+#$@KCA EQU   12                      SECTOR COUNT OF #KCALL
                        1243+*
                        1D24 1244+#$KRSU EQU   X'1D24'                 DISK ADDR OF #KRSUM
                        0C00 1245+#$$KRS EQU   X'0C00'                 CORE LOAD ADDRESS OF #KRSUM
                        000A 1246+#$@KRS EQU   10                      SECTOR COUNT OF #KRSUM
                        1247+*
                        1D5C 1248+#$UPTF EQU   X'1D5C'                 DISK ADDR OF #UPTFI
                        0C00 1249+#$$UPT EQU   X'0C00'                 CORE LOAD ADDRESS OF #UPTFI
                        0012 1250+#$@UPT EQU   18                      SECTOR COUNT OF #UPTFI
                        1251+*
                        1D24 1252+#$UPOV EQU   X'1D24'                 DISK ADDR OF #UPOVL
                        0C00 1253+#$$UPO EQU   X'0C00'                 CORE LOAD ADDRESS OF #UPOVL
                        0005 1254+#$@UPO EQU   05                      SECTOR COUNT OF #UPOVL
                        1255+*
                        1E00 1256+#$FMLN EQU   X'1E00'                 DISK ADDR OF #FMLNG
                        0200 1257+#$$FML EQU   X'0200'                 CORE LOAD ADDRESS OF #FMLNG
                        0052 1258+#$@FML EQU   82                      SECTOR COUNT OF #FMLNG
                        1259+*
                        2000 1260+#$#CNF EQU   X'2000'                 DISK ADDR OF ##CNFI
                        0000 1261+#$$#CN EQU   X'0000'                 CORE LOAD ADDRESS OF ##CNFI
                        0001 1262+#$@#CN EQU   01                      SECTOR COUNT OF ##CNFI
                        1263+*
                        2004 1264+#$KLLA EQU   X'2004'                 DISK ADDR OF #KLLAY
                        0920 1265+#$$KLL EQU   X'0920'                 CORE LOAD ADDRESS OF #KLLAY
                        0001 1266+#$@KLL EQU   01                      SECTOR COUNT OF #KKLAY
                        1267+*
                        2008 1268+#$ZLBM EQU   X'2008'                 DISK ADDR OF #ZLBMA
                        1100 1269+#$$ZLB EQU   X'1100'                 CORE LOAD ADDRESS OF #ZLBMA
                        0002 1270+#$@ZLB EQU   02                      SECTOR COUNT OF #ZLBMA
                        1271+*
                        2010 1272+#$ZL1M EQU   X'2010'                 DISK ADDR OF #ZL1MA
                        0F00 1273+#$$ZL1 EQU   X'0F00'                 CORE LOAD ADDRESS OF #ZL1MA
                        0007 1274+#$@ZL1 EQU   07                      SECTOR COUNT OF #ZL1MA
                        1275+*
                        2030 1276+#$ZL2M EQU   X'2030'                 DISK ADDR OF #ZL2MA
                        0F00 1277+#$$ZL2 EQU   X'0F00'                 CORE LOAD ADDRESS OF #ZL2MA
                        000D 1278+#$@ZL2 EQU   13                      SECTOR COUNT OF #ZL2MA
                        1279+*
                        2088 1280+#$ZL3M EQU   X'2088'                 DISK ADDR OF #ZL3MA
                        0C00 1281+#$$ZL3 EQU   X'0C00'                 CORE LOAD ADDRESS OF #ZL3MA
                        000A 1282+#$@ZL3 EQU   10                      SECTOR COUNT OF #ZL3MA
                        1283+*
                        20B0 1284+#$ZLVR EQU   X'20B0'                 DISK ADDR OF #ZLVRL
                        0F00 1285+#$$ZLV EQU   X'0F00'                 CORE LOAD ADDRESS OF #ZLVRL
                        0006 1286+#$@ZLV EQU   06                      SECTOR COUNT OF #ZLVRL
                        1287+*
                        2100 1288+#$KKEY EQU   X'2100'                 DISK ADDR OF #KKEYS
                        0C00 1289+#$$KKE EQU   X'0C00'                 CORE LOAD ADDRESS OF #KKEYS
                        0006 1290+#$@KKE EQU   06                      SECTOR COUNT OF #KKEYS
                        1291+*
```

```
      @SPFEQ - SYSTEM PROGRAM FILE EQUATES

ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT          VER 15, MOD 00  29/02/16  PAGE  27
                           2118 1292+#$#CKT EQU   X'2118'                      DISK ADDR OF #CKTB
                           0000 1293+#$$#CK EQU   X'0000'                      CORE LOAD ADDRESS OF #CKTB
                           0004 1294+#$@#CK EQU   04                           SECTOR COUNT OF #CKTB
                                1295+*
                           212C 1296+#$#INV EQU   X'212C'                      DISK ADDR OF ##INVD
                           0000 1297+#$$#IN EQU   X'0000'                      CORE LOAD ADDRESS OF ##INVD
                           003A 1298+#$@#IN EQU   58                           SECTOR COUNT OF ##INVD
                                1299+*
                           2300 1300+#$#PWR EQU   X'2300'                      DISK ADDR OF ##PWRK
                           0000 1301+#$$#PW EQU   X'0000'                      CORE LOAD ADDRESS OF ##PWRK
                           00C0 1302+#$@#PW EQU   192                          SECTOR COUNT OF ##PWRK
                                1303+*       END OF SYSTEM PROGRAM FILE EQUATES
                                1304+        PRINT ON
                                1305 *       @FXD  EXP-Y
                                1307+        PRINT ON
```

```
      @FXDEQ - FIXED ADDRESSES FOR SYSTEM NUCLEUS
```

```
                            1309+*******************************************************************
                            1310+*       GLOBAL INDICATORS STORED IN THE SYSTEM NUCLEUS, ENTRY POINTS  *
                            1311+*       FOR SYSNUC INTERFACE ROUINES.                                *
                            1312+*******************************************************************
  0000                      1313+       ORG   X'0000'                         *
            0000 1314+$$ZERO EQU   *                          ENTRY POINT TO LOAD DUMP PGM
            0004 1315+$FEARR EQU   $$ZERO+4                   VALUE OF ADDR IN ARR ON FE AID
                 1316+*
            0025 1317+$DISKN EQU   $$ZERO+37                  ADDR OF ENTRY TO DISK IOCS
            00DE 1318+$KE090 EQU   $$ZERO+X'00DE'             ADDR OF DKDISK ERR-PEND EXIT
            01D5 1319+$KE130 EQU   $$ZERO+X'01D5'             ADDR OF DKDISK HARD ERROR EXIT

  0345                      1321+       ORG   X'0345'                         *
            0345 1322+$ERLOG EQU   *                          ADDR OF ENTRY TO LOG I/O ERRORS
            0363 1323+$ER050 EQU   $$ZERO+X'0363'             START OF DISK OPS IN NERLOG


                            1325+*******************************************************************
                            1326+*       COMMUNICATION AREA REFERENCING NUCLEUS                     *
                            1327+*******************************************************************
                            1328+*
  03C0                      1329+       ORG   X'03C0'                         *
            03C0 1330+$NUCBS EQU   *                          START OF COMMUNICATION AREA
            03C0 1331+$RMRGN EQU   $NUCBS                     ADDR OF BYTE CONTAINING THE
                 1332+*                                       * SOFTWARE RIGHT MARGIN VALUE
            03C1 1333+$LMRGN EQU   $RMRGN+1                   ADDR OF BYTE CONTAINING THE
                 1334+*                                       * SOFTWARE LEFT MARGIN VALUE
            03C2 1335+$PRPOS EQU   $LMRGN+1                   ADDR OF BYTE CONTAINING CURRENT
                 1336+*                                       * POSITION OF MATRIX PRINTER
                 1337+*                                       * HEAD
            03C3 1338+$KEYCD EQU   $PRPOS+1                   ADDR OF BYTE CONTAINING KEYBOARD
                 1339+*                                       * INDICATORS. A LIST OF THE
                 1340+*                                       * INDICATORS AND MASKS FOLLOW
            0001 1341+$CARDI EQU   X'01'                      INPUT SOURCE INDR MASK
                 1342+*                                       * 0 - KEYBOARD INPUT
                 1343+*                                       * 1 - CARD OR PROC INPUT
            0002 1344+$IOYES EQU   X'02'                      I/O ROUTINES IN CORE INDR MASK
                 1345+*                                       * 0 - I/O ROUTINES NOT IN CORE
                 1346+*                                       * 1 - I/O ROUTINES IN CORE
            0004 1347+$NOLST EQU   X'04'                      NO LIST INDR MASK
                 1348+*                                       * 0 - LISTING REQUIRED
                 1349+*                                       * 1 - NO LISTING RESIRED
            0008 1350+$GUFIR EQU   X'08'                      GUFUDI ABORT INDR
                 1351+*                                       * 1 - GUFUDI INTERRUPT, NOT ABOR
                 1352+*                                       * 0 - GUFUDI ABORTED
                 1353+*                                       * FOR THE ABOVE INDICATOR TO BE
                 1354+*                                       * VALID, $INTRP MUST BE PRESENT
            0010 1355+$KYBSY EQU   X'10'                      KEYBOARD BUSY INDR
                 1356+*                                       * 0 - LINE FINISHED
                 1357+*                                       * 1 - LINE NOT YET COMPLETE
            0020 1358+$INRPT EQU   X'20'                      INTERRUPT INDR
                 1359+*                                       * 0 - PROGRAM NOT ABORTED
                 1360+*                                       * 1 - PROGRAM ABOPRTED
            0040 1361+$DTNMB EQU   X'40'                      * 1 - AUTOMATIC LINE NUMBERS
                 1362+*                                       * GENERATED FOR CARD INPUT
            0080 1363+$TRUNK EQU   X'80'                      TRUNCATED LINE INDR
                 1364+*                                       * 1 - LAST LINE TRUNCATED
```

                   1365+*   * 0 - LAST LINE COMPLETED

```
                    1367+********************************************************************
                    1368+*       REGISTER SAVE AREAS. THESE AREAS ARE AVAILABLE FOR         *
                    1369+*       TEMPORARELY USE BY ANY PROGRAM                              *
                    1370+********************************************************************

          03C5 1372+$BRSAV EQU    $KEYCD+2                  ADDR OF 2 BYTE BASE REG SAVE
          03C7 1373+$XRSAV EQU    $BRSAV+2                  ADDR OF 2 BYTE XR SAVE AREA

          03CB 1375+$TABLN EQU    $XRSAV+4                  CURRENT AUTOMATIC LINE NUMBER
                    1376+*                                  * TO BE INSERTED IF TAB KEY
                    1377+*                                  * PRESSED. (ADDR OF LINE NO.)
          03CD 1378+$CAERR EQU    $TABLN+2                  ADDR OF ERROR CODE SAVED FOR
                    1379+*                                  * INTERFACE WITH ERRPGM
          03CF 1380+$INLNO EQU    $CAERR+2                  ADDR OF EXECUTION TIME LINE
                    1381+*                                  * NUMBER FOR INTERPRETER
          03CE 1382+$ERRPG EQU    $INLNO-1                  ADDR OF INDICATOR BYTE IF
                    1383+*                                  * SPECIAL FUNCTION REQUESTED
                    1384+*                                  * OF ERROR PROGRAM
          0030 1385+$ERSTK EQU    X'30'                     TO BE MOVED TO $ERRPG IF A STACK
                    1386+*                                  * OF ERROR CODES IS TO BE PROCES
          0035 1387+$ERSFL EQU    X'35'                     SYNTAX CHECKERS $ERRPG SETTING
          0040 1388+$ERFIL EQU    X'40'                     TO BE MOVED TO $ERRPG IF FILE
                    1389+*                                  * LINE ERROR OCCURS
          0050 1390+$ER1N2 EQU    X'50'                     TO BE MOVED TO $ERRPG IF LEVEL
                    1391+*                                  * 1 AND 2 MESSAGES REQUIRED
          0080 1392+$ERKEY EQU    X'80'                     STANDARD ERROR SETTING USED BY
                    1393+*                                  * COMMAND ANALYZER ONLY
          03CF 1394+$ERRCT EQU    $INLNO                    ADDR OF COUNT BYTE FOR STACK
                    1395+*                                  * OF ERROR MESSAGES
```

```
                    1397+*********************************************************************
                    1398+*       SYSTEM STATUS EQUATES                                        *
                    1399+*********************************************************************
                    1400+*
               03D0 1401+$XIND1 EQU    $INLNO+1                    ADDR OF PRIMARY EXEC MODE INDRS
                    1402+*                                         * ENTRIES FOLLOW
               0001 1403+$RUNIT EQU    X'01'                       1 - EXECUTE IN RUN MODE
               0002 1404+$STEPT EQU    X'02'                       1 - EXECUTE IN STEP MODE
               0004 1405+$TRACE EQU    X'04'                       1 - EXECUTE IN TRACE MODE
                    1406+*                                         THE THREE MODE INDICATORS ARE
                    1407+*                                         MUTUALLY EXCLUSIVE. IF $TRACE
                    1408+*                                         IS ON, AT LEAST 1 OF THE TRACE
                    1409+*                                         TYPE CODE MUST ALSO BE ON.
               0008 1410+$TFLOW EQU    X'08'                       1 - TRACE FLOW
               0010 1411+$TRALL EQU    X'10'                       1 - TRACE ALL
               0020 1412+$TRVAR EQU    X'20'                       1 - TRACE SELECTED VARIABLES
               0040 1413+$XPREC EQU    X'40'                       EXECUTION PRECISION INDR
                    1414+*                                         * 0 - SHORT PRECISION
                    1415+*                                         * 1 - LONG PRECISION
               0080 1416+$VMDEF EQU    X'80'                       VM USAGE INDR
                    1417+*                                         * 1 - VIRTUAL MEMORY NOT EMPTY
                    1418+*                                         * 0 - VIRTUAL MEMORY EMPTY

               03D1 1420+$XIND2 EQU    $XIND1+1                    ADDR OF EXECUTION INDICATORS
                    1421+*                                         * MASK AND INDRS FOLLOW
               0001 1422+$EXCMD EQU    X'01'                       EXECUTION INDR
                    1423+*                                         * 1 - IN EXECUTION
               0002 1424+$PAUSE EQU    X'02'                       * 1 - PROGRAM IN PAUSE STATE
               0004 1425+$PSTEP EQU    X'04'                       * 1 - PAUSE CAUSED BY STEP MODE
               0008 1426+$PSTMT EQU    X'08'                       * 1 - PAUSE CAUSED BY PAUSE STMT
               0010 1427+$ABORT EQU    X'10'                       * 1 - ABORT EXECUTION

               03D2 1429+$IOIND EQU    $XIND2+1                    I/O STATUS INDICATORS
                    1430+*                                         * MASKS AND EXPLANATION FOLLOW
               0001 1431+$MPDWN EQU    X'01'                       MP STATE
                    1432+*                                         * 0 - MATRIX PRINTER OPERATIONAL
                    1433+*                                         * 1 - MATRIX PRINTER DOWN
               0002 1434+$CRTAV EQU    X'02'                       CRT AVAILABILITY
                    1435+*                                         * 0 - NO CRT ON SYSTEM
                    1436+*                                         * 1 - CRT ON THE SYSTEM
               0004 1437+$CRTNO EQU    X'04'                       SYSPRNT ON CRT
                    1438+*                                         * 0 - CRT NOT AVAIL FOR SYSPRNT
                    1439+*                                         * 1 - CRT MAY BE USED FOR SYSPRN
               0008 1440+$CMDKY EQU    X'08'                       KEYBOARD MODE
                    1441+*                                         * 0 - NORMAL KEYBOARD INPUT
                    1442+*                                         * 1 - COMMAND KEYS USE ONLY
               0010 1443+$PGMST EQU    X'10'                       PGM START KEY
                    1444+*                                         * 0 - MAY BE USED FOR AUTO LINE
                    1445+*                                         * 1 - NOT USED FOR AUTO LINE #
               0020 1446+$HRDER EQU    X'20'                       HARD ERROR INDICATOR
                    1447+*                                         * 0 - SOFT ERROR
                    1448+*                                         * 1 - HARD ERROR
               0040 1449+$DTRDR EQU    X'40'                       DATA RECORDER
                    1450+*                                         * 0 - DATA RECORDER NOT ON SYSTE
                    1451+*                                         * 1 - DATA RECORDER IS ON SYSTEM
               0080 1452+$LNPTR EQU    X'80'                       MP OPTION
```

@FXDEQ - FIXED ADDRESSES FOR SYSTEM NUCLEUS

```
                        1453+*                                  * 1 - 50 LPM OPTION AVAILABLE

                 03D3 1455+$CRTIN EQU    $IOIND+1               CRT COMMAND INDICATORS
                        1456+*                                  * MASKS AND EXPLANATION FOLLOW
                 0001 1457+$CRTUP EQU    X'01'                  1 - CRT IN ROLL UP MODE
                 0002 1458+$CRTDN EQU    X'02'                  1 - CRT IN ROLL DOWN MODE
                 0004 1459+$CRTPU EQU    X'04'                  1 - POP UP CONDITION REQUESTED
                 0008 1460+$CRTSP EQU    X'08'                  1 - ROLL STOP REQUESTED

                 03D4 1462+$INDR1 EQU    $CRTIN+1               WORK FILE STATUS INDICATORS
                        1463+*                                  * MASKS AND EXPLANATION FOLLOW
                 0001 1464+$PROCI EQU    X'01'                  PROCEDURE FILE INDR
                        1465+*                                  * 0 - NOT A PROCEDURE
                        1466+*                                  * 1 - A PROCEDURE
                 0002 1467+$PRESN EQU    X'02'                  WORK FILE PRECISION INDR
                        1468+*                                  * 0 - SHORT PRECISION USED
                        1469+*                                  * 1 - LONG PRECISION BEING USED
                 0004 1470+$WSIND EQU    X'04'                  WORKING STORAGE INDR MASK
                        1471+*                                  * 0 - WORKING STOR ON DISK IS EM
                        1472+*                                  * 1 - WORKING STORAGE IS NOT EMP
                 0008 1473+$WFLOK EQU    X'08'                  WORK FILE LOCK INDR
                        1474+*                                  * 0 - FILE NOT PROTECTED
                        1475+*                                  * 1 - FILE PROTECTED
                 0010 1476+$FITIN EQU    X'10'                  FIT SECTORS INDR MASK
                        1477+*                                  * 0 - FIT SECTORS NOT PRESENT
                        1478+*                                  * 1 - FIT SECTORS IN CORE
                 0020 1479+$PGMDT EQU    X'20'                  PGM DATA FILE INDR
                        1480+*                                  * 1 - PROGRAM GENERATED
                        1481+*                                  * DATA FILE IN WORK FILE
                 0040 1482+$KEYDT EQU    X'40'                  KEYBOARD OR CARD FILE INDR
                        1483+*                                  * 1 - KYBRD OR CARD GENERATED
                        1484+*                                  * DATA FILE IN WORK FILE
                 0080 1485+$BASIC EQU    X'80'                  BASIC PROGRAM INDR
                        1486+*                                  * 1 - BASIC PGM IN WORK FILE

                 03D5 1488+$INDR2 EQU    $INDR1+1               ADDR OF SYSTEM 1-BIT INDRS
                        1489+*                                  * MASKS AND EXPLANATION FOLLOW
                 0002 1490+$CMODE EQU    X'02'                  CONVERSATIONAL MODE INDR MASK
                        1491+*                                  * 0 - UTILITY MODE
                        1492+*                                  * 1 - CONVERSATIONAL MODE
                 0004 1493+$ERPND EQU    X'04'                  ERROR LOG PENDING INDR
                        1494+*                                  * 0 - NO LOGGING REQUIRED
                        1495+*                                  * 1 - ERROR LOGGING PENDING
                 0008 1496+$DKERR EQU    X'08'                  DISK ERROR INDR
                        1497+*                                  * 0 - ERROR WAS NOT DISK
                        1498+*                                  * 1 - ERROR WAS DISK, 2 ENTRIES
                        1499+*                                  * REQUIRED IN HISTORY LOG
                 0010 1500+$FCIND EQU    X'10'                  CRUSH INDR MASK
                        1501+*                                  * 1 - SINGLE LINE NO DELETION
                        1502+*                                  * THROUGH THE CMD ANALYZER REQUI
                        1503+*                                  * IF $FUIND, $FCIND AND $FDIND A
                        1504+*                                  * ALL ZERO, CRUCHING OP REQUIRED
                 0020 1505+$FUIND EQU    X'20'                  LINE PASSED INDR MASK
                        1506+*                                  * 1 - LINE PASSED
                 0040 1507+$FDIND EQU    X'40'                  LINE NUMBER LIST
                        1508+*                                  * 1 - LINE NO LIST IS DELETED
```

```
                         0080 1509+$READY EQU   X'80'                    PRINT READY INDR
                              1510+*                                     * 0 - READY WILL BE PRINTED
                              1511+*                                     * 1 - READY WON'T BE PRINTED

                         03D6 1513+$INDR3 EQU   $INDR2+1                 ADDR OF SYSTEM 1-BIT INDRS
                              1514+*                                     * MASKS AND EXPLANATION FOLLOW
                         0001 1515+$DBLOK EQU   X'01'                    SAVE PROTECTED WORK FILE MASK
                              1516+*                                     * 1 - FILE MAY BE SAVED TO $$LIB
                         0002 1517+$LIST  EQU   X'02'                    KLISTN INDR
                              1518+*                                     * 0 - IGNORE ROLL DOWN KEY
                              1519+*                                     * 1 - EXCEPT ROLL DOWN KEY
                         0004 1520+$ERHRD EQU   X'04'                    ERRPGM HARD ERROR INDR
                              1521+*                                     * 1 - ERRPGM WILL EXECUTE HARD
                              1522+*                                     *     HALT AFTER PRINTING MSG
                         0008 1523+$NOENB EQU   X'08'                    KEYBOARD ENABLE INDR
                              1524+*                                     * 0 - KEYBOARD NOT ENABLED -
                              1525+*                                     *     GUFUDI WILL ENABLE
                              1526+*                                     * 1 - KEYBOARD HAS ALREADY
                              1527+*                                     *     BEEN ENABLED
                         0010 1528+$CLBFR EQU   X'10'                    CLEAR INPUT LINE BUFFER INDR
                              1529+*                                     * 0 - DON'T CLEAR LINE BUFFER
                              1530+*                                     * 1 - CLEAR THE INPUT LINE BUFF
                         0020 1531+$MOUNT EQU   X'20'                    MOUNT KEYBOARD INDR MASK
                              1532+*                                     * 1 - ONLY MOUNT COMMAND VALID
                         0040 1533+$NWRKR EQU   X'40'                    REMOVABLE DISK WORK AREA INDR
                              1534+*                                     * 0 - CORRECT WORK AREA ON R1
                              1535+*                                     * 1 - NO WORK AREA ON R1
                         0080 1536+$NWRKF EQU   X'80'                    FIXED DISK WORK AREA INDR
                              1537+*                                     * 0 - CORRECT WORK AREA ON F1
                              1538+*                                     * 1 - NO WORK AREA ON F1

                         03D7 1540+$DKSIZ EQU   $INDR3+1                 ADDR OF DISK SIZE INDR
                              1541+*                                     * MASKS AND EXPLANATION FOLLOW
                         0001 1542+$DK100 EQU   X'01'                    1 - SYSTEM HAS 100 CYLS
                         0002 1543+$DK200 EQU   X'02'                    1 - SYSTEM HAS 200 CYLS
                         0004 1544+$DK400 EQU   X'04'                    1 - SYSTEM HAS 400 CYLS
                         0008 1545+$DK600 EQU   X'08'                    1 - SYSTEM HAS 600 CYLS
                         0010 1546+$DK800 EQU   X'10'                    1 - SYSTEM HAS 800 CYLS

                         03D8 1548+$XIND3 EQU   $DKSIZ+1                 PAST $XIND1
                              1549+*                                     * SEE $XIND1 FOR INDR MASKS

                         03DA 1551+$FILIB EQU   $XIND3+2                 ADDR OF CURRENT FILE LIB DADDR
                         03DC 1552+$USRDR EQU   $FILIB+2                 ADDR OF REL DISP TO 1ST USER BK
                         03DD 1553+$CONFG EQU   $USRDR+1                 CONFIGURATION INDRS
                         0001 1554+$22IMP EQU   X'01'                    0 - 13 INCH MATRIX PRINTER
                              1555+*                                     1 - 22 INCH MATRIX PRINTER
                         0002 1556+$16K   EQU   X'02'                    1 - CPU HAS 12 KBYTE
                         0004 1557+$12K   EQU   X'04'                    1 - CPU HAS 16 KBYTE
                              1558+*                                     * IF BOTH OFF: CPU HAS 8 KBYTE
                         0008 1559+$16CKY EQU   X'08'                    0 - KEYBOARD HAS  8 CMD KEYS
                              1560+*                                     1 - KEYBOARD HAS 16 CMD KEYS
                         0080 1561+$BIGCD EQU   X'80'                    1 - CPU HAS 129 DATA RECORDER

                         03DF 1563+$LEVEL EQU   $CONFG+2                 ADDR OF SYSTEM LEVEL NUMBER
```

@FXDEQ - FIXED ADDRESSES FOR SYSTEM NUCLEUS

```
                        03E0 1565+$DBGUF EQU    $LEVEL+1                  ADDR OF GUFUDI DEBUG INDR
                        0080 1566+$CRUSH EQU    X'80'                     0 - CRUSH THE FILE
                        0040 1567+$REORD EQU    X'40'                     0 - REORDER THE FILE
                        0020 1568+$IRKEY EQU    X'20'                     1 - ENABLE KEYBOARD INPUT
                        0010 1569+$IOPGS EQU    X'10'                     D1 PAGES INDR: 0 - ONE
                        0008 1570+$CALLI EQU    X'08'                     PROCEDURE CALL INDR
                             1571+*                                       * 0 - NOT A CALL
                             1572+*                                       * 1 - A CALL

                        03E1 1574+$KEYBD EQU    $DBGUF+1                  KEYBOARD TYPE INDR
                             1575+*                                       * THIS VALUE WILL BE A BINARY
                             1576+*                                       * VALUE FROM 1 TO 12 INDICATING
                             1577+*                                       * WHICH DATA TABLE IS IN USE

                        03E2 1579+$CRPOS EQU    $KEYBD+1                  ADDR OF CURRENT CURSOR POSITION
                        03E3 1580+$BUFPT EQU    $CRPOS+1                  LINE PRINTER BUFFER POINTER  1-3
                        03E4 1581+$LPRP3 EQU    $BUFPT+1                  LINE PRINTER FLAGS           1-3
                        03E5 1582+$LPROS EQU    $LPRP3+1                  TRUE LINE PRINTER PRINT POS. 1-3

                        03E6 1584+$NEXTB EQU    $LPROS+1                  REL DADDR PROCEDURE CALL     1-4
                        03E7 1585+$NEXTL EQU    $NEXTB+1                  DISPLACEMENT WITHIN DB       1-4
                        03E8 1586+$DFDET EQU    $NEXTL+1                  GRAPRO INTERNAL INDR         1-4
                        03EA 1587+$LPRIO EQU    $DFDET+2                  LINE PRINTER BUF INC. + PDAR 1-4

                        03F5 1589+$PTCH1 EQU    $DKSIZ+30                 LAST BYTE OF NUCLUES AREA
                             1590+*************************************************************************
                             1591+*         TABLES AND SYSTEM WORK AREAS                                 *
                             1592+*************************************************************************
                        03F6 1593+$VOLID EQU    $PTCH1+1                  ADDR OF LEFT BYTE VOLID TABLE
                        03F6 1594+$VOLR1 EQU    $VOLID                    ADDR LEFT BYTE VOLID FOR R1
                        03FE 1595+$VOLF1 EQU    $VOLR1+8                  ADDR LEFT BYTE VOLID FOR F1
                        0406 1596+$VOLR2 EQU    $VOLF1+8                  ADDR LEFT BYTE VOLID FOR R2
                        040E 1597+$VOLF2 EQU    $VOLR2+8                  ADDR LEFT BYTE VOLID FOR F2
                        0419 1598+$PKERT EQU    $VOLID+35                 ADDR OF 1ST ENTRY IN PACK ERROR
                             1599+*                                       * RATE TABLE
                        042D 1600+$PASWD EQU    $PKERT+20                 ADDR OF CURRENT PASSWORD
                        042E 1601+$HISTE EQU    $PASWD+1                  LEFT BYTE OF HISTORY LOG ENTRY
                        0435 1602+$HIST1 EQU    $HISTE+7                  ADDR OF 1ST ENTRY OF HIST LOG
                        043A 1603+$DATE  EQU    $HIST1+5                  ADDR OF CURRENT DATE
                        043B 1604+$EXFTR EQU    $DATE+1                   ADDR OF CORE EXPANSION FACTOR
                             1605+*                                       * THIS VALUE WILL BE ADDED TO
                             1606+*                                       * BUFFER ADDRESS (SET FOR 8K)
                             1607+*                                       * TO RE-POSITION THEM FOR
                             1608+*                                       * LARGER MACHINES
                        0443 1609+$WFNME EQU    $EXFTR+8                  ADDR OF WORK FILE NAME
                        0040 1610+$WFDEF EQU    X'40'                     WORK FILE DEFINED INDR
                             1611+*                                       * THIS MASK IS USED ON $WFNME
                             1612+*                                       * 0 - WORK FILE UNDEFINED
                             1613+*                                       * 1 - WORK FILE DEFINED
                        0449 1614+$DPLSV EQU    $WFNME+6                  ADDR OF 6 BYTE DPL SAVE AREA
                             1615+*                                       * FOR KEYBOARD PROGRAMS
                        044B 1616+$PRDEV EQU    $DPLSV+2                  ADDR OF 2 BYTE FIELD POINTING
                             1617+*                                       * TO THE SYSTEM PRINTER IOCR
                        044D 1618+$CRTAD EQU    $PRDEV+2                  ADDR OF ENTRY TO RELOCATE CRT
                        0454 1619+$PLST1 EQU    $CRTAD+7                  ADDR OF THREE 7-BYTES ENTRY I/O
                        045B 1620+$PLST2 EQU    $PLST1+7                  * PARM LISTS MOST RECENTLY USED
```

```
                        0462 1621+$PLST3 EQU   $PLST2+7                   * THE 1ST ENTRY IS MOST RECENT
                        0464 1622+$C0001 EQU   $PLST3+2                   ADDR OF 2 BYTE CONSTANT 1
```

```
                    1624+************************************************************************
                    1625+*        ENTRY POINTS TO INTERFACE ROUTINES AND THEIR WORK AREAS       *
                    1626+************************************************************************
          0465 1628+$SPRNT EQU    $C0001+1                      ADDR OF ENTRY TO THE SYSTEM
               1629+*                                           * PRINTER IOCR
          0469 1630+$CAERK EQU    $SPRNT+4                      ADDR OF ENTRY TO ERR ROUTINE
               1631+*                                           * INTERFACE. ERROR CODE MUST
               1632+*                                           * BE STORED PREVIOUS TO ENTRY
          046F 1633+$ERDPL EQU    $CAERK+6                      ADDR OF LEFT BYTE OF ERRPGM
               1634+*                                           * LOAD DPL
          0472 1635+$ERMAD EQU    $ERDPL+3                      ADDR OF DK ADDR, CNT OF ERRPGM
          0476 1636+$CIMSK EQU    $ERMAD+4                      ADDR OF THE INQUIRY REQUEST INDR
               1637+*                                           * X'87' IR NOT DISABLED
               1638+*                                           * X'80' IR MASKED
          0480 1639+$CIEXT EQU    $CIMSK+10                     ADDR OF IR EXIT INSTRUCTION
          0483 1640+$CIENT EQU    $CIEXT+3                      ADDR OF ENTRY FOR IR
          048D 1641+$UNMSK EQU    $CIENT+10                     ADDR OF ENTRY TO UNMASK IR
               1642+*                                           * IF NO SUSPENDED IR, CALLING
               1643+*                                           * PROGRAM RETURNED TO
          0496 1644+$CISUS EQU    $UNMSK+9                      ADDR OF INDR FOR SUSPENDED IR
               1645+*                                           * IF X'80' AN IR OCCURRED WHILE
               1646+*                                           * IR WAS MASKED
               1647+*                                           * IF X'87' NO IR TOOK PLACE
               1648+*                                           * WHILE IR WAS MASKED
          049D 1649+$CAIPL EQU    $CISUS+7                      ADDR OF ENTRY TO ABORT CURRENT
               1650+*                                           * OP AND RE-ENABLE KEYBOARD AND
          04A1 1651+$CARPL EQU    $CAIPL+4                      ADDR OF ENTRY TO ABORT CURRENT
               1652+*                                           * OP AND ENABLE IR
          04B4 1653+$CABLD EQU    $CARPL+X'13'                  ADDR OF ENTRY TO ABORT CURRENT O
          04BA 1654+$PAUSD EQU    $CABLD+6                      ADDR OF ENTRY OF ROUTINE TO
               1655+*                                           * SWAP CORE
          04D6 1656+$RSTR  EQU    $PAUSD+X'1C'                  ADDR OF ENTRY TO ENTRY CORE
               1657+*                                           * FROM DISK
          04F2 1658+$PSDXR EQU    $RSTR+X'1C'                   ADDR OF SAVED XR IN NPAUSE
          04FA 1659+$PSDBR EQU    $PSDXR+8                      ADDR OF SAVED BR IN NPAUSE
          04FE 1660+$SRTRN EQU    $RSTR+X'28'                   ADDR OF RETURN ADDR FROM $PAUSD
          050D 1661+$SFAID EQU    $SRTRN+15                     ADDR OF RETURN IF FE AID REQUEST
               1662+*                                           * IF THE ABOVE TWO ADDRESSES ARE
               1663+*                                           * EQUAL, RETURN TO $RSTR WILL BE
               1664+*                                           * BE FROM THE FE AID PROGRAM
          050E 1665+$CSDPL EQU    $RSTR+X'38'                   ADDR OF LEFT BYTE OF SAVE/RSTR D
          0511 1666+$SWPCR EQU    $CSDPL+3                      ADDR OF DKADDR, COUNT FOR CORE
               1667+*                                           * SAVE AREA
          0517 1668+$EXADR EQU    $SWPCR+6                      ADDRR OF DK ADDR, COUNT OF EXEC
               1669+*                                           * TIME MESSAGE PROGRAM
          051A 1670+$LOADR EQU    $EXADR+3                      ADDR OF ENTRY TO BLAST LOAD
               1671+*                                           * PROGRAM NOT RESIDING ON CYL 4
               1672+*                                           * RETURN IS TO CALLING PROGRAM
          051E 1673+$RLOAD EQU    $LOADR+4                      ADDR OF ENTRY TO BLAST LOAD
               1674+*                                           * PROGRAM NOT RESIDING ON CYL 4
          0522 1675+$BLOAD EQU    $RLOAD+4                      ADDR OF ENTRY TO BLAST LOAD
               1676+*                                           * PROGRAM RESIDING ON CYL 4
          054A 1677+$LOADB EQU    $BLOAD+X'28'                  ADDR OF SPECIAL ENTRY TO
               1678+*                                           * NBLOAD FOR SFLOAD/SFFIND
               1679+*                                           * AND FZPINV
```

```
                       054E 1680+$TROVR EQU    $BLOAD+X'2C'              ADDR OF FE TRACE INDR
                            1681+*                                      * @NOP - NO TRACE PERFORMED
                            1682+*                                      * @UCB - TRACE PERFORMED
                       0550 1683+$BLRTN EQU    $TROVR+2                  ADDR OF RETURN POINT FROM ZTRACE
                       0569 1684+$BLNOE EQU    $BLRTN+X'19'              ADDR OF NO EXECUTE INDR-NBLOAD
                            1685+*                                      * @NOP - CALLING PGM RETURNED TO
                            1686+*                                      * @UCB - LOADED PROGRAM EXECUTED
                            1687+*                                      * ENTRY TO $LOADR SETS THE ABOVE
                            1688+*                                      * INDR TO @NOP.  IF THE CALLING
                            1689+*                                      * SETS THE INDR TO @NOP BEFORE
                            1690+*                                      * CALLING $BLOAD, RETURN WILL BE
                            1691+*                                      * MADE UPON COMPLETION OF THE
                            1692+*                                      * ABSOLUE LOAD
                       0571 1693+$LDRTN EQU    $BLOAD+X'4F'              ADDR OF THE RETURN ADDR IN NBLOA
                       0579 1694+$BLDPL EQU    $BLOAD+X'57'              ADDR OF LEFT BYTE OF $BLOAD'S
                            1695+*                                      * DPL (DPL OF LAST PGM LOADED)
                       057F 1696+$WAITF EQU    $BLDPL+6                  ADDR OF LEFT BYTE OF DISK
                            1697+*                                      * WAIT AND CHECK ERRORS DPL
                       0583 1698+$GUFIO EQU    $WAITF+4                  ADDR OF DK ADDR, COUNT OF GUFUDI
                       0587 1699+$BSADR EQU    $GUFIO+4                  ADDR OF DADDR RELOCATION FACTOR
                            1700+*                                      * FOR PGMS NOT RESIDING ON CYL 6
                       0588 1701+$FEMAP EQU    $BSADR+1                  ADDR OF START OF CORE MAP
                       05A2 1702+$ZTRAD EQU    $FEMAP+X'1A'              ADDR OF ZTRACE DADDR

   05FF                     1704+       ORG    X'05FF'
                       05FF 1705+$IPLDV EQU    *                        ADDR OF IPL INDR
                            1706+*                                      * X'00' - IPL WAS FROM R1
                            1707+*                                      * X'01' - IPL WAS FROM F1
                       0600 1708+$ENDNU EQU    $IPLDV+1                  ADDR OF THE FIRST BYTE
                            1709+*                                      * FOLLOWING SYSNUC
                            1710+*      END OF FIXED ADDRESSES SYSTEM NUCLEUS EQUATES
                            1711+       PRINT ON
                            1712 *      @CAN  EXP-Y
                            1714+       PRINT ON
```

```
      @CANEQ - COMMON CORE LOCATIONS OUTSIDE NUCLEUS
```

```
                       1716+********************************************************************
                       1717+*       INPUT LINE HEADER                                          *
                       1718+********************************************************************
              0600 1719+$$ILHD EQU    $ENDNU                     FIRST BYTE OF INPUT LINE HEADER
                       1720+*
              0601 1721+$$ILEN EQU    $$ILHD+1                   SECOND BYTE OF SDF LENGTH FIELD
                       1722+*
              0602 1723+$$UPAR EQU    $$ILEN+1                   UP ARROW LOCATION IN LAST LINE
                       1724+*
              0603 1725+$$CKEY EQU    $$UPAR+1                   CMD KEY FUNCTION CODE
                       1726+*                                    * EXECUTABLE CMD KEYS
              0605 1727+$$BNLN EQU    $$ILEN+4                   SECOND BYTE OF BINARY LINE NO.
                       1728+*
              0606 1729+$$TPCD EQU    $$BNLN+1                   TYPE CODE FIELD

                       1731+********************************************************************
                       1732+*       INPUT LINE TEXT                                            *
                       1733+********************************************************************
              0607 1734+$$INLN EQU    $$TPCD+1                   FIRST BYTE CHAR OF INPUT LINE
                       1735+*
              0666 1736+$$CDND EQU    $$INLN+@CARDL-1            LAST CHAR OF CARD INPUT
                       1737+*
              06FA 1738+$$INND EQU    $$INLN+@LINSZ-1            LAST CHAR OF INPUT LINE BUFFER

                       1740+********************************************************************
                       1741+*       KEYBOARD ROUTINE LOCATIONS AND MASKS                       *
                       1742+********************************************************************
              0890 1743+$$PRES EQU    $ENDNU+X'0290'             ENABLE KEYBOARD ENTRY TO DEPRES
                       1744+*
              09E1 1745+$$KBDT EQU    $$PRES+X'0151'             DATA BYTE FROM KEYBOARD
              0081 1746+$$$STD EQU    B'10000001'                CLI MASK FOR START KEY DATA
              0091 1747+$$$EPL EQU    B'10010001'                CLI MASK FOR ENTER PLUS KEY
                       1748+*
              09E2 1749+$$KBSN EQU    $$KBDT+1                   TYPE BYTE FROM KEYBOARD
              0040 1750+$$$DAT EQU    B'01000000'                TBM MASK FOR DATA KEY
              0020 1751+$$$CMD EQU    B'00100000'                TBM MASK FOR COMMAND KEY
              0010 1752+$$$FUN EQU    B'00010000'                TBM MASK FOR FUNCTION KEY
                       1753+*
              09EB 1754+$$LPOS EQU    $$KBSN+9                   PRINT HEAD POSITION ADDR
              0AFE 1755+$$EOSA EQU    $$PRES+X'026E'             LOCATION OF EOS ADDR
              0B44 1756+$$COFF EQU    $$PRES+X'02B4'             ENTRY TO TURN OFF CMD LIGHTS
              0B3D 1757+$$CKFF EQU    $$PRES+X'02AD'             ENTRY TO TURN OFF CMD LIGHTS 1-1
              0BBF 1758+$$DATB EQU    $$PRES+X'032F'             ADDR OF DATA TABLE TYPE INDR IN
                       1759+*                                    * DEPRES (VALUE: 1-9)

                       1761+********************************************************************
                       1762+*       MATRIX PRINTER ROUTINE ENTRY POINT                         *
                       1763+********************************************************************
              0707 1764+$$PRNT EQU    $ENDNU+X'0100'+@HDRLN      DPRINT ENTRY
              0782 1765+$$PRTN EQU    $$PRNT+X'007B'             ADDR OF CARRIER RETURN TEST IN
                       1766+*                                    * DPRINT.  MASKS FOLLOE
                       1767+*                                    * @NOP - NO TEST MADE
                       1768+*                                    * @BNL - TEST WILL BE MADE
              07CE 1769+$$PSIO EQU    $$PRNT+X'00C7'             ADDR OF SIO CTRL IN DPRINT
              07E9 1770+$$PCNT EQU    $$PRNT+X'00E2'             ADDR OF PPL CNT IN DPRINT
```

```
          @CANEQ - COMMON CORE LOCATIONS OUTSIDE NUCLEUS
```

```
                    1772+*********************************************************************
                    1773+*        CARD READER LOCATIONS                                      *
                    1774+*********************************************************************
          0890 1775+$$CDRD EQU    $$PRES                     ENTRY POINT TO READ CARDS
                    1776+*
          08C0 1777+$$CDBS EQU    $$CDRD+X'0030'             ENTRY POINT TO WAIT FOR READ

                    1779+*********************************************************************
                    1780+*        CRT OUTPUT ROUTINE LOCATIONS                               *
                    1781+*********************************************************************
          2000 1782+$$PYMP EQU    $$ZERO+X'2000'             ENTRY POINT TO CRT PLUS PRINT
                    1783+*
          2004 1784+$$PLYN EQU    $$PYMP+4                   ENTRY POINT TO CRT ONLY
                    1785+*
          209C 1786+$$CSNS EQU    $$PYMP+X'009C'             LOCATION OF SENSE BYTE IN
                    1787+*                                   * DSPLYN
          2143 1788+$$PRFL EQU    $$PYMP+X'0143'             ENTRY POINT FOR PRINTER FAILURE
                    1789+*
          2200 1790+$$PYCD EQU    $$PYMP+X'0200'             ENTRY POINT FOR COMMAND KEYS
                    1791+*                                   * OR CLEAR CRT FUNCTION

                    1793+*********************************************************************
                    1794+*        MISCELLANEOUS LOCATIONS                                    *
                    1795+*********************************************************************
          1C00 1796+$$ERSK EQU    X'1C00'                    START ADDR OF ERROR CODE STACK
          00A0 1797+$$$NLN EQU    X'00A0'                    HIGH ORDER BYTE OF LINE NUMBER
                    1798+*                                   * IN STACK IF NO. NOT DESIRED
          1C00 1799+$$SLIB EQU    X'1C00'                    SECONDAIRY LINE INPUT BUFFER
          06FF 1800+$$XIND EQU    $ENDNU+X'00FF'             EXEC INDR PASS AREA
          0080 1801+$$$ERN EQU    B'10000000'                RUN FUNC SAVED FILE INDR MASK
          1E00 1802+$$WSPB EQU    X'1E00'                    LOCATION OF BAGETC BUFFER
          06FF 1803+$$FLIB EQU    $$XIND                     FILE LIB ADDR PASS AREA
          1D00 1804+$$FITS EQU    X'1D00'                    LOCATION OF FIT

                    1806+*********************************************************************
                    1807+*        KEYWORD COMMAND LOAD ADDRESSES                             *
                    1808+*********************************************************************
          0600 1809+$$KLD1 EQU    $ENDNU                     PROGRAMS THAT LOAD BEHIND
                    1810+*                                   * SYSNUC
          0700 1811+$$KLD2 EQU    $ENDNU+X'0100'             PROGRAMS THAT LOAD BEHIND
                    1812+*                                   * THE INPUT LINE BUFFER
          0C00 1813+$$KLD3 EQU    $ENDNU+X'0600'             STANDARD LOAD ADDRESS BEHIND
                    1814+*                                   * I/O ROUTINES
                    1815+*        END OF COMMON CORE LOCATIONS EQUATES
                    1816+       PRINT ON
                    1817 *       @WKA  EXP-Y
                    1819+       PRINT ON
```

```
                   1821+****************************************************************************
                   1822+*        THIS EQUATE MODULE PROVIDES THE FIXED PHYSICAL DISK         *
                   1823+*        ADDRESSES OF PGM'S AND WA'S IN THE SYSTEM WORK AREA.         *
                   1824+****************************************************************************
                   1825+*
                   1826+***     SELECTED SYSTEM PROGRAMS AND BAD LINE
                   1827+*
         0400 1828+#@WAR1 EQU   X'0400'                      DADDR OF SELECTED PGM AREA
         0401 1829+#@WAF1 EQU   X'0401'                      DADDR OF SELECTED PGM AREA
         0400 1830+#@BOVL EQU   X'0400'                      PHYSICAL DADDR OF #BOVLY
         0018 1831+#@@BOV EQU   24                           SECTOR COUNT OF #BOVLY
         0480 1832+#@SFSY EQU   X'0480'                      PHYSICAL DADDR OF #SFSYN
         0011 1833+#@@SFS EQU   17                           SECTOR COUNT OF #SFSYN
         0401 1834+#@GUFU EQU   X'0401'                      PHYSICAL DADDR OF #GUFUD
         0010 1835+#@@GUF EQU   16                           SECTOR COUNT OF #GUFUD
         04AD 1836+#@SDSY EQU   X'04AD'                      PHYSICAL DADDR OF #SDSYN
         0004 1837+#@@SDS EQU   4                            SECTOR COUNT OF #SDSYN
         0441 1838+#@ERRP EQU   X'0441'                      PHYSICAL DADDR OF #ERRPG
         0003 1839+#@@ERR EQU   3                            SECTOR COUNT OF #ERRPG
         044D 1840+#@LDSV EQU   X'044D'                      PHYS DADDR OF #LOADR SAVE AREA
         0002 1841+#@@LDS EQU   2                            SECTOR COUNT OF #LOADR SA
         0455 1842+#@#BAD EQU   X'0455'                      PHYSICAL DADDR OF THE BAD LINE
         0001 1843+#@@#BA EQU   1                            SECTOR COUNT OF ##BADL
         0481 1844+#@ECMA EQU   X'0481'                      PHYSICAL DADDR OF #ECMAN
         0006 1845+#@@ECM EQU   6                            SECTOR COUNT OF #ECMAN
         0499 1846+#@SFLO EQU   X'0499'                      PHYSICAL DADDR OF SFLOAD
         0005 1847+#@@SFL EQU   5                            SECTOR COUNT OF SFLOAD
         04BD 1848+#@SFFI EQU   X'04BD'                      PHYSICAL DADDR OF SFFIND
         0008 1849+#@@SFF EQU   8                            SECTOR COUNT OF SFFIND
         0459 1850+#@#IO1 EQU   X'0459'                      PHYSICAL DADDR OF 1ST I/O SECTOR
         045D 1851+#@#IO2 EQU   X'045D'                      PHYSICAL DADDR OF 2ST I/O SECTOR
         0002 1852+#@@#SC EQU   2                            SECTOR COUNT OF I/O SECTOR
         0008 1853+#@@#08 EQU   8                            NO. ENTRIES IN 1ST I/O SECTOR
         0004 1854+#@@#04 EQU   4                            NO. ENTRIES IN 2ND I/O SECTOR
         0001 1855+#@@#IO EQU   1                            SECTOR COUNT OF I/O SECTOR
         04C4 1856+#@SFOV EQU   X'04C4'                      PHYSICAL DADDR OF #SFOVR
         0005 1857+#@@SFO EQU   5                            SECTOR COUNT OF #SFOVR
                   1858+*
                   1859+***     WORK FILE ADDRESSES
                   1860+*
         0500 1861+#@#WFT EQU   X'0500'                      PHYSICAL DADDR 1ST SCTR OF FIT
         0003 1862+#@@#WF EQU   3                            SCTR COUNT OF FIT
         050C 1863+#@#WDB EQU   X'050C'                      PHYSICAL DADDR OF 1ST DATA BLOCK
         00BD 1864+#@@#WD EQU   189                          SCTR COUNT OF DATA BLOCKS
                   1865+*
                   1866+***     VIRTUAL MEMORY ADDRESSES
                   1867+*
         0700 1868+#@#VFP EQU   X'0700'                      PHYSICAL DADDR FIRST PAGE OF VM
         0708 1869+#@VTRL EQU   X'0708'                      DADDR OF SAVED 'TRACE' VAR.LIST
         0001 1870+#@@VTR EQU   1                            SCTR COUNT SAVED 'TRACE' VAR.LIS
         093D 1871+#@#VLP EQU   X'093D'                      PHYSICAL DADDR LAST PAGE OF VM
         0100 1872+#@@#VM EQU   256                          SCTR COUNT OF VIRTUAL MEMORY
                   1873+*
                   1874+***     TEMPORARELY WORK AREA ADDRESSES
                   1875+*
         0941 1876+#@#TFS EQU   X'0941'                      PHYSICAL DADDR 1ST SCTR TEMP WK
```

@WKAEQ - SYSTEM WORK AREA ADDRESSES

```
                    0020 1877+#@@#TW EQU    32                        SCTR COUNT OF TEMP WORKAREA
                    0941 1878+#@#TAT EQU    X'0941'                   PHYSICAL DADDR STMT ADDR TABLE
                    0010 1879+#@@#TA EQU    16                        SCTR COUNT OF STMT ADDR TABLE
                    0941 1880+#@#TSY EQU    X'0941'                   PHYSICAL DADDR SYMBOL TBL SAVE A
                    0005 1881+#@@#TS EQU    5                         SCTR COUNT OF SYMBOL TBL SAVE AR
                    09A1 1882+#@#TBA EQU    X'09A1'                   PHYSICAL DADDR BRANCH ADDR TABLE
                    0010 1883+#@@#TB EQU    16                        SCTR COUNT OF BRANCH ADDR TABLE
                    09A1 1884+#@VSFI EQU    X'09A1'                   PHYSICAL DADDR VSFINT
                    0010 1885+#@@VSF EQU    16                        SCTR COUNT OF VSFINT
                    000F 1886+#@@VSL EQU    15                        SCTR COUNT OF VSFLOA
                         1887+*      END OF WORK AREA EQUATES
                         1888+      PRINT ON
```

```
                          1891 ****************************************************************************
                          1892 * 5703-XM1  COPYRIGHT IBM CORP. 1970                              *
                          1893 *           REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083   *
                          1894 *                                                                 *
                          1895 ****************************************************************************
                          1896 *STATUS                                                           *
                          1897 *    VERSION 1 MODIFICATION 0                                      *
                          1898 *                                                                 *
                          1899 *FUNCTION                                                         *
                          1900 *    #GUFUD UPDATES THE WORK FILE IN THE SYSTEM WORK AREA AND      *
                          1901 *    MAINTAINS THE FILE IN LINE-NUMBER ORDER.                     *
                          1902 *                                                                 *
                          1903 *ENTRY POINTS                                                     *
                          1904 *     GUFCSH - WORK FILE CRUSH AND REORDER.                       *
                          1905 *     GURDIN - COMMON DISK READ SUBROUTINE.                       *
                          1906 *     DL4ICS - WORK FILE IOCS.                                    *
                          1907 *     GUFPAK - PACK CORE BUFFERS SUBROUTINE.                      *
                          1908 *     GUFENT - INITIALIZATION.                                    *
                          1909 *     GCPACK - PACK BASIC PROGRAM STATEMENT SUBROUTINE.           *
                          1910 *     GUFUPD - WORK FILE UPDATE.                                  *
                          1911 *                                                                 *
                          1912 *INPUT                                                            *
                          1913 *   N/A                                                           *
                          1914 *                                                                 *
                          1915 *OUTPUT                                                           *
                          1916 *   N/A                                                           *
                          1917 *                                                                 *
                          1918 *EXTERNAL REFENECES                                               *
                          1919 *   $DISKN - ENTRY TO SYSTEM DISK ROUTINE                         *
                          1920 *                                                                 *
                          1921 *EXITS, NORMAL                                                    *
                          1922 *   NORMAL RETURN IS TO THE 1ST INSTRUCTION FOLLOWING THE TWO BYTE *
                          1923 *   ADDRESS POINTING TO THE DPL.                                  *
                          1924 *                                                                 *
                          1925 *EXITS, ERROR                                                     *
                          1926 *   N/A                                                           *
                          1927 *                                                                 *
                          1928 *TABLES/WORK AREAS                                                *
                          1929 *   N/A                                                           *
                          1930 *                                                                 *
                          1931 *ATTRIBUTES                                                       *
                          1932 *   RELOCATABLE                                                   *
                          1933 *   REUSABLE                                                      *
                          1934 *                                                                 *
                          1935 *CHARACTER CODE DEPENDENCY                                        *
                          1936 *   THE OPERATION OF THIS MODULE DOES NOT DEPEND UPON A PARTICULAR *
                          1937 *   INTERNAL REPRESENTATION OF THE EXTERNAL CHARACTER SET.        *
                          1938 *                                                                 *
                          1939 *NOTES                                                            *
                          1940 *   ERROR PROCEDURES                                              *
                          1941 *      N/A                                                        *
                          1942 *   REGISTER USAGE                                                *
                          1943 *      @BR IS SAVED AND RESTORED ON EXIT, @XR IS NOT USED.  @ARR IS *
                          1944 *      USED TO PROVIDE THE ADDRESS OF THE PARAMETER.  THE @ARR IS  *
                          1945 *      INCREMENTED BT TWO AND SAVED AS THE RETURN ADDRESS.        *
                          1946 *   SAVED/RESTORED AREAS                                          *
```

```
                              1947 *      N/A                                                      *
                              1948 *    MODIFICATION CONSIDERATIONS                                *
                              1949 *      N/A                                                      *
                              1950 *    REQUIRED MODULES                                           *
                              1951 *      @SYSEQ - SYSTEM SOFTWARE EQUATES                         *
                              1952 *      @FXDEQ - SYSTEM NUCLEUS EQUATES                          *
                              1953 *      @ERMEQ - GENERAL ERROR MESSAGE EQUATES                   *
                              1954 *      @SPFEQ - SYSTEM PROGRAM FILE EQUATES                     *
                              1955 *      @CANEQ - COMMON CORE LOCATIONS OUTSIDE NUCLEUS           *
                              1956 *      @WKAEQ - WORK AREA EQUATES                               *
                              1957 *    OTHER                                                      *
                              1958 *      FIT - FILE INDEX TABLE                                   *
                              1959 *      CIT - CORE INDEX TABLE                                   *
                              1960 *      SDF - SEGMENT DESCRIPTION FIELD                          *
                              1961 *      EOS - END OF SEGMENT                                     *
                              1962 *****************************************************************
```

```
      #GUFUD GUFLD1 - FILE UPDATE/CRUSHER
```

```
                         1964 *
                         1965 *                      ENTRY 2 - GUFUDI ONLY LOADED
                         1966 *
                         1967 *         HDR   #GUFUD,0                   SYSTEM PROGRAM HEADER
                         1968 *************************************************************************
                         1969 *   PROGRAM HEADER FOR DISK LOAD
                         1970 *************************************************************************
                         1971 *#$GUFU EQU   X'1880'                     DISK ADDR OF #GUFUD
                         1972 *#$$GUF EQU   X'0C00'                     CORE LOAD ADDRESS OF #GUFUD
                         1973 *#$@GUF EQU   016                         SECTOR COUNT OF #GUFUD
   0C00                  1974         ORG   #$$GUF                      CORE LOAD ADDRESS
                    0C00 1975 $$$$$$ EQU   *                            FIRST LOCATION IN PROGRAM
   0C00 7BC7E4C6E4C4  0C05 1976         DC    CL6'#GUFUD'                PROGRAM NAME
   0C06 4A            0C06 1977         DC    IL1'074'                   PROGRAM NUMBER OF #GUFUD
                         1978 *  UFUD EQU    *                           ENTRY POINT TO PROGRAM
                         1979 *** END OF EXPENSION ***
                         1980 *
                    0C07 1981 GUFUDI EQU    *                           ENTRY TO FILE UPDATE/CRUSHER
   0C07 C0 87 0D90   1982         B     GUF711                     GO TO ENTRY ROUTINE
                         1983 *      MTEXT @@M130-@PRINT,@@M131-@PRETR,@@M132-@PRETR
                         1984 *************************************************************************
                         1985 *      PPL'S AND TEXT FOR MESSAGE
                         1986 *************************************************************************
   0C0B 40            0C0B 1987 @@M130 DC    AL1(@PRINT)                PRINT CONTROL FUNCTION
   0C0C 17            0C0C 1988         DC    IL1'23'                    LENGTH OF MESSAGE
   0C0D 0C17          0C0E 1989         DC    AL(@CADDR)(@@T130)         ADDRESS OF MESSAGE
                         1990 *
   0C0F C0            0C0F 1991 @@M131 DC    AL1(@PRETR)                PRINT CONTROL FUNCTION
   0C10 05            0C10 1992         DC    IL1'05'                    LENGTH OF MESSAGE
   0C11 0C2E          0C12 1993         DC    AL(@CADDR)(@@T131)         ADDRESS OF MESSAGE
                         1994 *
   0C13 C0            0C13 1995 @@M132 DC    AL1(@PRETR)                PRINT CONTROL FUNCTION
   0C14 32            0C14 1996         DC    IL1'50'                    LENGTH OF MESSAGE
   0C15 0C33          0C16 1997         DC    AL(@CADDR)(@@T132)         ADDRESS OF MESSAGE
                         1998 *
                    0C17 1999 @@T130 EQU    *                           LEFT BYTE OF MESSAGE
   0C17 C6E4D5C3E3C9D6D5 0C2D 2000         DC    CL23'FUNCTION INTERRUPTED - '
                    0C2E 2001 @@T131 EQU    *                           LEFT BYTE OF MESSAGE
   0C2E D9C5C1C4E8    0C32 2002         DC    CL05'READY'
                    0C33 2003 @@T132 EQU    *                           LEFT BYTE OF MESSAGE
   0C33 C5D9D9D6D940F5F7 0C64 2004         DC    CL50'ERROR 574 NEXT AUTOMATIC LINE NUMBER WILL OVERFLOW'


                         2006 *
                         2007 *        PATCH AREA FOR MESSAGES
                         2008 *
   0C65                0C73 2009 $$$001 DS    CL15                       MSG EXPANSION PATCH AREA
```

```
       #GUFUD GCPACK - BASIC STATEMENT CHARACTER PACKING ROUTINE
```

```
                          2011 *
                          2012 *        GCPBFR MUST BE EQUATED TO THE FIRST BYTE OF THE SDF
                          2013 *        PRECEEDING THE BASIC STATEMENT IN THE USED DEFINED AREA
                          2014 *
                     0001 2015         DROP  1                        NO BASE REGISTER USED IN RTN
                     0C74 2016 GCPACK EQU   *                         ENTRY TO GCPACK ROUTINE
                          2017 *
                          2018 ***    SAVE REGISTERS AND SET UP POINTERS
                          2019 *
   0C74 34 08 0CE1        2020         ST    GCP140+@OP1,@ARR         SAVE RET ADDR IN RESTORE INSTR
   0C78 34 02 0CDD        2021         ST    GCP130+@OP1,@XR          SAVE @XR IN RESTORE INSTR
   0C7C 34 01 0CD9        2022         ST    GCP120+@OP1,@BR          SAVE @RB IN RESTORE INSTR
   0C80 C2 01 1C08        2023         LA    GCPBFR+@STEXT+@B1,@BR    SET POINTER FOR PACKED PORTION
   0C84 C2 02 1C07        2024         LA    GCPBFR+@STEXT,@XR        SET POINTER FOR UNPACKED PART
                          2025 *
                          2026 ***    TEST FOR EOS AND REPEAT CHARACTERS
                          2027 *
   0C88 BD 1E 00          2028 GCP020 CLI   @ZERO(,@XR),@EOS         TEST FOR CARR RETURN CHAR
   0C8B F2 81 3E          2029         JE    GCP110                   YES, GO CALC STMT LENGTH
   0C8E AD 00 00 01       2030         CLC   @ZERO(1,@XR),@B1(,@XR)   COMPARE FIRST TWO CHAR'S
   0C92 F2 01 29          2031         JNE   GCP090                   NOT EQUAL, GO MOVE 1ST TO PACKD
   0C95 AD 00 01 02       2032         CLC   @B1(1,@XR),GCPTWO(,@XR)  COMPARE 2ND 3RD CHAR'S
   0C99 F2 01 22          2033         JNE   GCP090                   NOT EQUAL, GO MOVE 1ST TO PACKD
                          2034 *
                          2035 ***    DETERMINE LENGTH OF REPEAT COUNT
                          2036 *
   0C9C 7C 02 00          2037         MVI   @ZERO(,@BR),GCPTWO       SET UP INITIAL REPEAT COUNT
   0C9F E2 02 01          2038 GCP050 LA    @B1(,@XR),@XR            SET UNPACKED POINTER UP 1 CHAR
   0CA2 AD 00 01 02       2039         CLC   @B1(1,@XR),GCPTWO(,@XR)  TEST FOR ADDITIONAL REPEATS
   0CA6 F2 01 19          2040         JNE   GCP100                   NO, GO INCR POINTERS
                          2041 *
                          2042 ***    TEST FOR MAX REPEAT COUNT AND RETURN TO PACKING MORE CHARACTERS
                          2043 *
   0CA9 7D 1B 00          2044         CLI   @ZERO(,@BR),GCPMAX       IS REPEAT COUNT AT MAX ?
   0CAC F2 81 09          2045         JE    GCP080                   YES, GO INCR POINTERS
   0CAF 4E 00 00 0CE2     2046         ALC   @ZERO(1,@BR),GCPONE      NO, ADD ONE TO REPEAT COUNTER
   0CB4 C0 87 0C9F        2047         B     GCP050                   GO TEST FOR MORE REPEAT CHAR'S
   0CB8 D2 01 01          2048 GCP080 LA    @B1(,@BR),@BR            SET POINTER OF PACKED AREA UP 1
   0CBB E2 02 01          2049         LA    @B1(,@XR),@XR            SET POINTER OF INPUT AREA UP 1
   0CBE 6C 00 00 01       2050 GCP090 MVC   @ZERO(1,@BR),@B1(,@XR)   MOVE CHAR TO PACKED STMT AREA
   0CC2 D2 01 01          2051 GCP100 LA    @B1(,@BR),@BR            INCREMENT PACKED AREA POINTER
   0CC5 E2 02 01          2052         LA    @B1(,@XR),@XR            INCREMENT INPUT AREA POINTER
   0CC8 C0 87 0C88        2053         B     GCP020                   GO BACK TO CHECK NEXT CHARACTER
                          2054 *
                          2055 ***    CALCULATE STATEMENT LENGTH AND RETURN TO CALLING PROGRAM
                          2056 *
   0CCC 34 01 1C01        2057 GCP110 ST    GCPBFR+@SDF1,@BR         SAVE PTR TO CALCULATE LENGTH
   0CD0 0F 01 1C01 0CE4   2058         SLC   GCPBFR+@SDF1,GCPSTL(@CADDR)  SUBTRACT STARTING LOCATION
   0CD6 C2 01 0000        2059 GCP120 LA    *-*,@BR                  RELOAD BASE REGISTER
   0CDA C2 02 0000        2060 GCP130 LA    *-*,@XR                  RELOAD INDEX REGISTER
   0CDE C0 87 0000        2061 GCP140 B     *-*                      RETURN
                          2063 *
                          2064 ***    DEFINE CONSTANTS AREA
                          2065 *
   0CE2 01           0CE2 2066 GCPONE DC    XL1'01'                  INCR REPEAT COUNTER FACTOR
```

```
    0CE3 1C00            0CE4 2067 GCPSTL DC    AL2(GCPBFR)               START OF STATEMENT CADDR
                              2068 *
                              2069 ***    EQUATES
                         0002 2070 GCPTWO EQU   2                         INITLZ REPEAT COUNT VALUE
                         001B 2071 GCPMAX EQU   27                        MAX REPITITION COUNT ALLOWED
                              2072 *      END OF GCPACK
```

```
        #GUFUD GCPACK - BASIC STATEMENT CHARACTER PACKING ROUTINE


ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT          VER 15, MOD 00  29/02/16  PAGE  48

                           2074 *        PATCH ,1
                           2075 ************************************************************************
                           2076 *        PATCH AREA 1
                           2077 ************************************************************************
                           2078 *
                           2079 ***     CALCULATE AREA LEFT IN THIS SECTOR
                           2080 *
                      0CE5 2081 $$$$L1 EQU   *                        START OF PATCH AREA 1
    0D00                   2082        ORG   *,256,0                   SET LOC CNTR TO NEXT SECTOR
                      0D00 2083 $$$$T1 EQU   *                        DEFINE ADDR OF SCTR BOUNDARY
    0CE5                   2084        ORG   $$$$L1                    SET LOC CNTR TO START OF
                           2085 *                                     * PATCH AREA
    0CE5             0CFF  2086 $$$$$1 DS    CL($$$$T1-$$$$L1)         PATCH AREA
                           2087 ************************************************************************
```

```
      #GUFUD GCPACK - BASIC STATEMENT CHARACTER PACKING ROUTINE


ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT          VER 15, MOD 00  29/02/16  PAGE  49

    0D0C                        2089         ORG   *+@HDRLN+5            SET BOUNDARY
                                2090  *
                                2091  ***      NEW LINE IS AN ADDITION TO THE FILE
                                2092  *
    0D0C 0D 01 1D02 1A5B        2093 GUF822 CLC   GUF939(2),GUF519       LINE COUNT AT MAXIMUM ?
    0D12 F2 04 08               2094         JNH   GUF825               NO, BYPASS ERROR MESSAGE
                                2095  *
                                2096  ***      FILE IS LOGICALLY FULL ERROR
                                2097  *
    0D15 3C 8C 03CD             2098         MVI   $CAERR,@@E531          SET ERROR CODE
    0D19 C0 87 1841             2099         B     GUF330               GO SET LINE NUMBER
    0D1D 0D 01 1D09 1A70        2100 GUF825 CLC   GUF942(@CADDR),GUF666  LAST DB IN THE WORK FILE ?
    0D23 F2 82 11               2101         JL    GUF831               NO, GO INCR LINE COUNT
    0D26 0D 00 1C01 1A3B        2102         CLC   GUF924,GUF486         YES, WILL NEW STMT FIT ?
    0D2C F2 04 08               2103         JNH   GUF831               YES, GO INCR LINE COUNT
                                2104  *
                                2105  ***      FILE IS PHYSICALLY FULL ERROR
                                2106  *
    0D2F 3C 8B 03CD             2107         MVI   $CAERR,@@E530          SET ERROR CODE
    0D33 C0 87 1841             2108 GUF828 B     GUF330               GO SET LINE NUMBER
    0D37 0E 01 1D02 1A55        2109 GUF831 ALC   GUF939(2),GUF510       INCR FILE LINE COUNT
    0D3D 3C 80 1839             2110         MVI   GUF327+@Q,@NOP         SET SW TO DECR COUNT IF NO FIT
                                2111  *
                                2112  ***      A NEW REPLACEMENT IS HANDLED AS AN ADDITION TO THE FILE WITH
                                2113  ***      THE OLD LINE BEING WRITTEN OVER.
                                2114  *
    0D41 0C 01 1A3A 1C05        2115 GUF834 MVC   GUF483(GUF507),GUF927  REPLACE HIGHEST LINE NUMBER
                           0001 2116         DROP  1                    DROP BASE ADDRESSING
    0D47 0C 01 1A5D 1A57        2117         MVC   GUF525(@CADDR),GUF513  CALCULATE NUMBER OF BYTES
    0D4D 0F 01 1A5D 1A35        2118         SLC   GUF525(@CADDR),GUF468  * LEFT IN CBI
    0D53 0C 01 1A5F 1A5D        2119         MVC   GUF528(@CADDR),GUF525  CALCULATE NUMBER OF BYTES OF DATA
    0D59 0F 00 1A5F 1A3B        2120         SLC   GUF528(1),GUF486       * LEFT IN CBI
    0D5F C0 81 1086             2121         BZ    GUF855               IF ZERO, GO SET INDR
    0D63 34 02 1052             2122         ST    GUF846+@OP2,@XR        CALCULATE NUMBER OF DATA TO BE
    0D67 0E 01 1052 1A5F        2123         ALC   GUF846+@OP2(@CADDR),GUF528  * MOVED FROM CBI
    0D6D 0E 01 1052 1A37        2124         ALC   GUF846+@OP2(@CADDR),GUF471  *
    0D73 C0 87 1007             2125         B     GUF837               GO TO PART 2 TO CONTINUE INSERT
```

```
        #GUFUD GCPACK - BASIC STATEMENT CHARACTER PACKING ROUTINE


ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT            VER 15, MOD 00  29/02/16  PAGE  50

                                2127 *        PATCH 25,2
                                2128 ***********************************************************************
                                2129 *        PATCH AREA 2
                                2130 ***********************************************************************
      0D77                  0D8F 2131 $$$$$2 DS    CL25                           PATCH AREA FOR PROGRAM
                            0D90 2132 GUFENT EQU   *                              ENTRY SECTION
                                2133 *
                                2134 ***    ENTRY ROUTINE
                                2135 *
      0D90 3C 80 0476            2136 GUF711 MVI   $CIMSK,@NOP                    MASK INQUIRY REQUESTS
      0D94 39 80 03D5            2137        TBF   $INDR2,$READY                  SHOULD 'READY' BE PRINTED ?
      0D98 F2 10 14              2138        JT    GUF713                         YES, GO RETURN CARR AND PRINT
      0D9B 3B 80 03D5            2139        SBF   $INDR2,$READY                  RESET READY INDR
      0D9F 38 01 03C3            2140        TBN   $KEYCD,$CARDI                  IN CARD INPUT MODE ?
      0DA3 F2 90 82              2141        JF    GUF732                         NO, BYPASS PRINTER WAIT
      0DA6 C0 87 0465            2142        B     $SPRNT                         WAIT FOR PRINT COMPLETE
      0DAA 057F             0DAB 2143        DC    AL2($WAITF)                    *
      0DAC F2 87 79              2144        J     GUF732                         GO START SEEK
      0DAF 38 20 03E0            2145 GUF713 TBN   $DBGUF,$IRKEY                  SHOULD KEYBD INPUT BE ENABLED ?
      0DB3 F2 90 08              2146        JF    GUF714                         NO, GO TEST I/O RTN'S IN CORE
      0DB6 3B 07 03C3            2147        SBF   $KEYCD,$CARDI+$IOYES+$NOLST  SET INPUT TO KEYBOARD
      0DBA 3B 20 03E0            2148        SBF   $DBGUF,$IRKEY                  SET SW OFF TO ENABLE KEYBD INPUT
      0DBE 38 02 03C3            2149 GUF714 TBN   $KEYCD,$IOYES                  ARE I/O RTN'S IN CORE ?
      0DC2 F2 10 2E              2150        JT    GUF717                         YES, BYPASS DISK OP
      0DC5 C0 87 051A            2151        B     $LOADR                         READ IN I/O ROUTINES
      0DC9 0F55             0DCA 2152        DC    AL2(GUF816)                    *
      0DCB 3A 02 03C3            2153        SBN   $KEYCD,$IOYES                  SET I/O RTN'S IN CORE INDR
      0DCF 38 08 03E0            2154        TBN   $DBGUF,$CALLI                  CALL PROCEDURE ?
      0DD3 F2 90 10              2155        JF    GUF715                         NO
      0DD6 38 01 03C3            2156        TBN   $KEYCD,$CARDI                  CARD BIT SET ON ?
      0DDA F2 90 09              2157        JF    GUF715                         BYPASS OVERLAY IF NOT
      0DDD C0 87 051A            2158        B     $LOADR                         LOAD GRAPRO
      0DE1 0F47             0DE2 2159        DC    AL2(GUF790)                    DPL OF PARAMETER
      0DE3 F2 87 0D              2160        J     GUF717                         BYPASS OVERLAY READ
      0DE6 38 01 03C3            2161 GUF715 TBN   $KEYCD,$CARDI                  INPUT SOURCE = CARDS ?
      0DEA F2 90 06              2162        JF    GUF717                         NO, BYPASS ACCESS OF CARD RTN
      0DED C0 87 051A            2163        B     $LOADR                         READ IN #DREAD
      0DF1 0F41             0DF2 2164        DC    AL(@CADDR)(GUF789)             * CADDR OF #DREAD DPL
      0DF3 C0 87 0465            2165 GUF717 B     $SPRNT                         GO TO SYSTEM D/P DEVICE TO
      0DF7 0F3F             0DF8 2166        DC    AL2(GUF786)                    * RETURN THE CARRIAGE
      0DF9 3B 70 03D5            2167        SBF   $INDR2,$FDIND+$FUIND+$FCIND  SET FOR CRUSH ONLY OPERATION
      0DFD 38 20 03C3            2168 GUF720 TBN   $KEYCD,$INRPT                  WAS PRIOR FUNC ABORTED ?
      0E01 F2 90 18              2169        JF    GUF729                         NO, GO TO PRINT READY
      0E04 38 08 03C3            2170        TBN   $KEYCD,$GUFIR                  WAS GUFUDI CRUSHING ?
      0E08 F2 10 09              2171        JT    GUF723                         YES, BYPASS FUNC INTERRUPTED MS
      0E0B C0 87 0465            2172        B     $SPRNT                         PRINT FUNC INTERRUPTED MESSAGE
      0E0F 0C0B             0E10 2173        DC    AL(@CADDR)(@@M130)             *
      0E11 F2 87 04              2174        J     GUF726                         GO SET OFF INTERRUPTED INDR
      0E14 3C 00 0602            2175 GUF723 MVI   $$UPAR,@ZERO                   SET CMD KEY 4 COUNT TO 0
      0E18 3B 03 03C3            2176 GUF726 SBF   $KEYCD,$INRPT                  SET OFF INTERRUPTED INDR
      0E1C C0 87 0465            2177 GUF729 B     $SPRNT                         PRINT 'READY' MESSAGE
      0E20 0C0F             0E21 2178        DC    AL(@CADDR)(@@M131)             *
                                2179 *        SPRNT $WAITF                         WAIT FOR PRINT COMPLETE
      0E22 C0 87 0465            2180        B     $SPRNT                         PRINT ON SYSTEM PRINTER
      0E26 057F             0E27 2181        DC    AL2($WAITF)                    PPL ADDRESS
                                2182 *** END OF EXPANSION ***
```

```
                                2183 *
                                2184 ***     ENTRY TWO ROUTINE - GUFUDI INITIALZATION.  THIS ROUTINE IS
                                2185 ***     EXECUTED FOLLOWING ENTRY ONE ROUTINE OR INITIALLY IF GUFUDI
                                2186 ***     ONLY IS LOADED.
                                2187 *
    0E28 38 02 03D5             2188 GUF732 TBN    $INDR2,$CMODE           IN CONVERSATIONAL MODE
    0E2C F2 90 1F               2189        JF     GUF741                  NO, GO SET NO CRUSH INDR
    0E2F 38 10 03D4             2190 GUF735 TBN    $INDR1,$FITIN           FIT SECTORS IN CORE ?
    0E33 F2 10 0A               2191        JT     GUF738                  YES, BYPASS FIT READ
                                2192 *     DISK  GUF810                     READ FILE INDEX TABLE
    0E36 C0 87 0025             2193        B      $DISKN                  PERFORM PHYSICAL DISK OP
    0E3A 0F4D           0E3B    2194        DC     AL2(GUF810)             DPL ADDRESS
                                2195 *** END OF EXPANSION ***
    0E3C 3A 10 03D4             2196        SBN    $INDR1,$FITIN           SET FIT IN CORE INDR
    0E40 38 20 03D4             2197 GUF738 TBN    $INDR1,$PGMDT           PROGRAM GENERATED DATA FILE ?
    0E44 F2 10 07               2198        JT     GUF741                  YES, GO SET NO CRUSH INDR
    0E47 38 40 0443             2199        TBN    $WFNME,$WFDEF           IS WORK FILE DEFINED
    0E4B F2 10 07               2200        JT     GUF744                  YES, GO CHK TYPE OF UPDATE
    0E4E 3C 80 1110             2201 GUF741 MVI    GUF003,@NOP             SET INSTR TO BYPASS CRUSHER
    0E52 F2 87 75               2202        J      GUF762                  GO TO ENABLE INPUT
    0E55 38 40 03D5             2203 GUF744 TBN    $INDR2,$FDIND           DELETE LIST PASSED
    0E59 F2 10 D3               2204        JT     GUF777                  YES, GO WAIT FOR FIT TO READ IN
    0E5C 38 10 03D5             2205        TBN    $INDR2,$FCIND           SINGLE DELETE FROM CA
    0E60 F2 10 17               2206        JT     GUF747                  YES, GO MOVE LINE NUMBER
    0E63 38 20 03D5             2207        TBN    $INDR2,$FUIND           LINE TO BE UPDATED
    0E67 F2 90 60               2208        JF     GUF762                  NO, GO ENABLE INPUT
    0E6A 38 40 03D4             2209        TBN    $INDR1,$KEYDT           DATA FILE STATEMENT
    0E6E F2 10 0D               2210        JT     GUF750                  YES, GO ENABLE INPUT
    0E71 0C F9 1CFA 06FA        2211        MVC    $$SLIB+@STYPE+@LINSZ(@STEXT+@LINSZ-@B1),$$INND  SAVE STMT
    0E77 F2 87 0A               2212        J      GUF753                  GO ENABLE INPUT
    0E7A 3C 00 1C06             2213 GUF747 MVI    $$SLIB+@STYPE,@ZERO      CLEAR RANGE TEST POSITION
    0E7E 0C 01 1C05 0605        2214 GUF750 MVC    $$SLIB+@SBLN(@SBLNL),$$BNLN  MOVE LINE NR TO 2NDARY BUFFER
    0E84 0C 01 1C03 1A68        2215 GUF753 MVC    $$SLIB+@SDF3(2),GUF645   SET 2ND HALF OF SDF TO ZERO
    0E8A C2 02 0604             2216 GUF756 LA     $$BNLN-1,@XR             USE C2DEC5 TO CONVERT BINARY
    0E8E C0 87 0F5B             2217        B      C2DEC5                  * LINE NO TO DECIMAL
    0E92 0C 03 03CB 0F99        2218        MVC    $TABLN(GUF540),C2DVAL   MOVE IT TO $TABLN
    0E98 06 21 03CB 0F54        2219        AZ     $TABLN(GUF540),GUF813(2)  INCR AUTO LINE NUMBER
    0E9E F2 08 29               2220        JNOZ   GUF762                  NO OVERFLOW, BYPASS MESSAGE
    0EA1 38 01 03C3             2221        TBN    $KEYCD,$CARDI           CARD INPUT ?
    0EA5 F2 90 16               2222        JF     GUF759                  NO, GO GIVE IMMEDIATE MESSAGE
    0EA8 38 40 03C3             2223        TBN    $KEYCD,$DTNMB           AUTO LINE BEING GENERATED ?
    0EAC F2 90 1B               2224        JF     GUF762                  NO, BYPASS PRINTING MESSAGE
    0EAF 3C 87 1B6F             2225        MVI    GUF906+@Q,@UCB          RETURN TO ERRPGM AFTER UPDATE
    0EB3 3C A3 03CD             2226        MVI    $CAERR,@@E574           SET OVERFLOW ERROR CODE
    0EB7 3C 80 03CE             2227        MVI    $ERRPG,$ERKEY           SET ERROR TYPE TO FORCE KEYBD MDE
    0EBB F2 87 43               2228        J      GUF771                  GO CHECK LINE TYPE

    0EBE C0 87 0465             2230 GUF759 B      $SPRNT                  GO PRINT OVERFLOW AUTO LINE
    0EC2 0C13           0EC3    2231        DC     AL(@CADDR)(@@M132)      *
                                2232 *     SPRNT $WAITF                    WAIT FOR PRINT COMPLETE
    0EC4 C0 87 0465             2233        B      $SPRNT                  PRINT ON SYSTEM PRINTER
    0EC8 057F           0EC9    2234        DC     AL2($WAITF)             PPL ADDRESS
                                2235 *** END OF EXPANSION ***
    0ECA 38 10 03D6             2236 GUF762 TBN    $INDR3,$CLBFR           CLEAR PRIME INPUT BUFFER ?
    0ECE F2 90 12               2237        JF     GUF765                  NO, BYPASS...
    0ED1 3C 40 06FF             2238        MVI    $$KLD2-1,@BLANK         CLEAR THE PRIMARY
```

```
    0ED5 0C FE 06FE 06FF        2239      MVC   $$KLD2-2,$$KLD2-1(@SCTSZ-1)  * INPUT BUFFER
    0EDB 3C 00 0602             2240      MVI   $$UPAR,@ZERO                 SET CMD KEY 4 COUNT TO 0
    0EDF 3B 10 03D6             2241      SBF   $INDR3,$CLBFR                SET THE CLEAR INDR OFF
    0EE3 38 01 03C3             2242 GUF765 TBN  $KEYCD,$CARDI               CARD INPUT ?
    0EE7 F2 90 07               2243      JF    GUF768                       NO, GO ENABLE INPUT
    0EEA 3C 80 1B7A             2244      MVI   GUF909+@Q,@NOP               SET FOR CARD READ AFTER UPDATE
    0EEE F2 87 10               2245      J     GUF771                       BYPASS KEYBOARD ENABLE
    0EF1 3B 10 03D2             2246 GUF768 SBF  $IOIND,$PGMST               SET FOR AUTO LINE NUMBER
    0EF5 38 08 03D6             2247      TBN   $INDR3,$NOENB                INPUT ALREADY ENABLED ?
    0EF9 C0 90 0890             2248      BF    $$PRES                       NO, GO DO IT
    0EFD 3B 08 03D6             2249      SBF   $INDR3,$NOENB                SET OFF ALREADY ENABLED INDR
    0F01 38 20 03D5             2250 GUF771 TBN  $INDR2,$FUIND               LINE TO INSERT IN WORK FILE ?
    0F05 F2 10 19               2251      JT    GUF774                       YES, GO CHECK FILE TYPE
    0F08 38 10 03D5             2252      TBN   $INDR2,$FCIND                SINGLE LINE TO DELETE
    0F0C F2 10 26               2253      JT    GUF780                       YES, GO START FIT SEARCH
    0F0F C0 87 0025             2254      B     $DISKN                       WAIT FOR FIT TO READ IN
    0F13 057F         0F14 2255           DC    AL2($WAITF)                  *
    0F15 38 01 03C3             2256      TBN   $KEYCD,$CARDI                INPUT SOURCE = CARDS ?
    0F19 C0 10 1B7C             2257      BT    GUF912                       YES, GO ENABLE INPUT
    0F1D C0 87 1107             2258      B     GUGENT                       GO CRUSH ONLY
    0F21 38 40 03D4             2259 GUF774 TBN  $INDR1,$KEYDT               FILE TYPE = DATA ?
    0F25 F2 10 0D               2260      JT    GUF780                       YES, BYPASS CHAR PACKING
    0F28 C0 87 0C74             2261      B     GCPACK                       GO CHARARCTER PACK BASIC STMT
    0F2C F2 87 06               2262      J     GUF780                       GO WAIT FOR FIT IN
    0F2F 0C 04 1C08 1C0D        2263 GUF777 MVC  GUF927+GUF585,GUF927+GUF585+GUF588(GUF588)  SET 1ST DELETE
    0F35 C0 87 0025             2264 GUF780 B    $DISKN                       WAIT FOR FIT TO READ IN
    0F39 057F         0F3A 2265           DC    AL2($WAITF)                  *
    0F3B C0 87 1A7B             2266      B     GUF867                       GO SEARCH FILE INDEX TABLE
                               2267 *
                               2268 ***   PARAMETERS LISTS AND MESSAGES USED BY GUFUDI'S ENTRY ROUTINES.
                               2269 *
    0F3F 80           0F3F 2270 GUF786 DC  AL1(@RETRN)                      PPL TO RETURN CARRIER ONLY
    0F40 80           0F40 2271        DC  AL1(@RETRN)                      *
    0F41 01           0F41 2272 GUF789 DC  AL1(@DGET)                       DPL TO READ IN #DREAD IF INPUT
    0F42 0200         0F43 2273        DC  AL2(#$DREA)                      * IS FROM CARDS.
    0F44 01           0F44 2274        DC  AL1(#$@DRE)                      *
    0F45 0889         0F46 2275        DC  AL2(#$$DRE)                      *
                               2276 *
    0F47 01           0F47 2277 GUF790 DC  AL1(@DGET)                       DPL TO READ IN $GRAP
    0F48 0690         0F49 2278        DC  AL2(#$GRAP)                      *
    0F4A 03           0F4A 2279        DC  AL1(#$@GRA)                      *
    0F4B 0889         0F4C 2280        DC  AL2(#$$GRA)                      *
                               2281 *
    0F4D 01           0F4D 2282 GUF810 DC  AL1(@DGET)                       DPL TO READ IN #THE FILE INDEX
    0F4E 0500         0F4F 2283        DC  AL2(#@#WFT)                      * TABLE
    0F50 03           0F50 2284        DC  AL1(#@@#WF)                      *
    0F51 1D00         0F52 2285        DC  AL2($$FITS)                      *
                               2286 *
    0F53 F1F0         0F54 2287 GUF813 DC  DL2'10'                          INCREMENT FOR AUTO LINE NUMBER
    0F55 01           0F55 2288 GUF816 DC  AL1(@DGET)                       DPL TO READ THE I/O ROUTINES
    0F56 014C         0F57 2289        DC  AL2(#$DPRI)                      *
                      1C00 2290 GCPBFR EQU $$SLIB
    0F58 05           0F58 2291        DC  AL1(#$@DPR)                      *
    0F59 0700         0F5A 2292        DC  AL2(#$$DPR)                      *
                      0F5B 2293 GUF819 EQU *                               START OF C2DEC5
```

```
      #GUFUD C2DEC5 -- CONVERT BINARY TO DECIMAL


ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT           VER 15, MOD 00  29/02/16  PAGE  53
                           2295 *     ORG    GUF819                      PLACE C2DEC5
                           2296 ********************************************************************
                           2297 *   SERIALLY REUSABLE SUBROUTINE TO CONVERT A 2 BYTE BINARY VALUE TO  *
                           2298 *   A 5 BYTE POSITIVE DECIMAL NUMBER.                                 *
                           2299 *   ON ENTRY @XR POINTS TO THE LEFT BYTE OF THE BINARY VALUE.         *
                           2300 *   ON RETURN C2DVAL IS THE RIGHT BYTE OF THE 5 BYTES DECIMAL VALUE   *
                           2301 *   WITH LEADING ZEROS WHICH MAY BE MODIFIED BY THE USER IN ANY WAY   *
                           2302 *   IN IT'S LOCATION.                                                 *
                           2303 *   THE 2 BYTES BINARY VALUE IS NOT ALTERED.                          *
                           2304 *   @XR IS NOT ALTERED.                                               *
                           2305 *   @BR IS SAVED AND RESTORED AT EXIT.                                *
                           2306 ********************************************************************

                  0F5B 2308 C2DEC5 EQU    *                         MODULE ENTRY POINT
                  0F5B 2309        USING C2DEC5,@BR                  BASE ADDRESS SPECIFICATION
   0F5B 34 01 0F8F           2310        ST    C2D050+@OP1,@BR       SAVE @BR
   0F5F C2 01 0F5B           2311        LA    C2DEC5,@BR            LOAD BASE REGISTER
   0F63 74 08 38             2312        ST    C2D052+@OP1(,@BR),@ARR  SAVE RETURN ADDRESS
                           2313 *       INITIALIZE DECIMAL INCREMENTER AND DECIMAL SUM TO 1 AND 0 RESP.
   0F66 54 90 43 39          2314        ZAZ   C2D903(C2D903-C2D901,@BR),C2D901(C2D902-C2D901,@BR)
   0F6A 7C 01 17             2315        MVI   C2D030+@D1(,@BR),@B1  INITIALIZE DISP TO BYTE 1
   0F6D 7C 01 16             2316 C2D020 MVI   C2D030+@Q(,@BR),@B1   INIT TEST TO BIT 7
                           2317 *
   0F70 B8 00 00             2318 C2D030 TBN   *-*(,@XR),*-*         TEST IF THIS BIT IS OFF
   0F73 F2 90 04             2319        JF    C2D040               * BR AROUND SUM INCREMENT
                           2320 *       INCREMENT DECIMAL SUM BY DECIMAL VALUE OF THIS TESTED BIT
   0F76 56 04 3E 43          2321        AZ    C2DVAL(C2D903-C2DVAL,@BR),C2D903(C2D903-C2DVAL,@BR)
                           2322 *       DOUBLE DECIMAL VALUE OF INCREMENT TO VALUE OF NEXT BIT
   0F7A 56 04 43 43          2323 C2D040 AZ    C2D903(C2D903-C2DVAL,@BR),C2D903(C2D903-C2DVAL,@BR)
   0F7E 5E 00 16 16          2324        ALC   C2D030+@Q(1,@BR),C2D030+@Q(,@BR)   SHIFT BIT MASK LEFT ONE
   0F82 D0 20 15             2325        BNOL  C2D030(,@BR)          CONTINUE LOOP UNLESS ALL BITS
                           2326 *                                   * TESTED
   0F85 5F 00 17 13          2327        SLC   C2D030+@D1(1,@BR),C2D020+@Q(,@BR)   DECR DISP TO BYTE 0
   0F89 D0 81 12             2328        BZ    C2D020(,@BR)          FALL THROUGH IF UNDERFLOW
   0F8C C2 01 0000           2329 C2D050 LA    *-*,@BR               RESTORE @BR
   0F90 C0 87 0000           2330 C2D052 B     *-*                   RETURN TO CALLING PROGRAM
                           2331 *
                           2332 ***   WORK AREA
                           2333 *
   0F94 F1         0F94 2334 C2D901 DC    DL1'1'                      INIT WORK AREA
                  0F95 2335 C2D902 EQU    *                         FIST BYTE OF DECIMAL VALUE
   0F95           0F99 2336 C2DVAL DS    CL5                         5 BYTES DECIMAL VALUE
   0F9A           0F9E 2337 C2D903 DS    CL5                         DECIMAL INCREMENTER
```

```
                          2339 *         PATCH ,3
                          2340 *********************************************************************
                          2341 *         PATCH AREA 3
                          2342 *********************************************************************
                          2343 *
                          2344 ***     CALCULATE AREA LEFT IN THIS SECTOR
                          2345 *
     0F9F 2346 $$$$L3 EQU    *                      START OF PATCH AREA 3
     1000                   2347        ORG    *,256,0             SET LOC CNTR TO NEXT SECTOR
     1000 2348 $$$$T3 EQU    *                      DEFINE ADDR OF SCTR BOUNDARY
     0F9F                   2349        ORG    $$$$L3             SET LOC CNTR TO START OF
                          2350 *                              * PATCH AREA
     0F9F         0FFF 2351 $$$$$3 DS     CL($$$$T3-$$$$L3)      PATCH AREA
                          2352 *********************************************************************
```

```
ERR LOC   OBJECT CODE     ADDR STMT SOURCE STATEMENT            VER 15, MOD 00  29/02/16  PAGE  55

    1007                   2354         ORG    *+@HDRLN                    SET BOUNDARY
    1007 3C FF 104E        2355 GUF837 MVI    GUF846+@Q,GUF615            SET LENGTH FOR MOVE INSTR
    100B 0E 00 104E 1A5F   2356         ALC    GUF846+@Q(1),GUF528         *
    1011 0C 01 1050 1A6A   2357         MVC    GUF846+@OP1(@CADDR),GUF648  CALC ADDR TO MOVE DATA TO IN
    1017 0E 01 1050 1A5F   2358         ALC    GUF846+@OP1(@CADDR),GUF528  * CB2
    101D 0F 00 1A43 1A5F   2359         SLC    GUF489-GUF411(1),GUF528     DECR CB2 UNUSED SPACE COUNTER
    1023 0D 00 1A5D 1C01   2360 GUF840 CLC    GUF525(1),GUF924            WILL NEW LINE FIT IN CB1 ?
    1029 F2 82 62          2361         JL     GUF858                      NO, GO SPLIT IT
    102C 36 02 1C01        2362 GUF843 A      GUF924,@XR                  INCR XR BY LENGTH OF NEW LINE
    1030 0C 01 1069 1A6C   2363         MVC    GUF852+@DOP2(@CADDR),GUF660 CALC. ADDR OF NEW LINE FOR
    1036 0E 01 1069 1C01   2364         ALC    GUF852+@DOP2(@CADDR),GUF924 * MOVE TO CB1
    103C 3C FF 1066        2365         MVI    GUF852+@Q,GUF615            CALC. Q-CODE FOR NEW LINE MOVE
    1040 0E 00 1066 1C01   2366         ALC    GUF852+@Q(1),GUF924         *
    1046 38 01 1A62        2367         TBN    GUF561,@B1                  ANYTHING TI UNSERT IN CB2
    104A F2 10 18          2368         JT     GUF852                      NO, BYPASS INSERTION
    104D 0C 00 0000 0000   2369 GUF846 MVC    *-*(1),*-*                  MOVE SOMETHING TO SOMEWHERE
    1053 0C 01 1062 1050   2370         MVC    GUF849+@OP1(@CADDR),GUF846+@OP1  SET UP MOVE INST TO PLACE
    1059 0E 01 1062 1A55   2371         ALC    GUF849+@OP1(@CADDR),GUF510  * A NULL SDF AFTER THE DATA
    105F 0C 00 0000 1A65   2372 GUF849 MVC    *-*(@B1),GUF645-GUF585      JUST MOVED TO CB2
    1065 8C 00 00 0000     2373 GUF852 MVC    @ZERO(1,@XR),*-*            MOVE SOMETHING TO SOMEWHERE ELSE
    106A 0C 00 1A3B 1A5D   2374         MVC    GUF486(1),GUF525           CORRECT THE UNUSED SPACE COUNTER
    1070 0F 00 1A3B 1C01   2375         SLC    GUF486(1),GUF924            *
    1076 0C 00 1A37 1A3B   2376         MVC    GUF471(1),GUF486           SET FREE SPACE FOR 'PAK'
    107C 0E 00 1A35 1C01   2377         ALC    GUF468(1),GUF924           INCR DISP BY LENGTH OF NEW LINE
    1082 C0 87 1B6A        2378         B      GUF903                     GO TO GO TO 'PAK'
    1086 3C 01 1A62        2379 GUF855 MVI    GUF561,@B1                  SET NO INSERTION TO CB2 INDR
    108A C0 87 1023        2380         B      GUF840                     CHECK IF NEW LINE WILL FIT
                           2381 *
                           2382 ***    THE NEW ADDITION MUST BE SPLIT BETWEEN CB1 AND CB2
                           2383 *
    108E 0C 01 1A61 1C01   2384 GUF858 MVC    GUF531(@CADDR),GUF924       CALC AMT OF NEW LINE THAT GOES
    1094 0F 00 1A61 1A5D   2385         SLC    GUF531(1),GUF525           * TO CB2
    109A C2 01 0D07        2386         LA     GUF690,@BR                 LOAD CB2 ADDR TO XR2
    109E 0C 01 10D6 1A6C   2387         MVC    GUF861+@DOP2(@CADDR),GUF660 CALC ADDR OF NEW LINE IN
    10A4 0E 01 10D6 1C01   2388         ALC    GUF861+@DOP2(@CADDR),GUF924 * PASSED AREA
    10AA 0C 00 1C01 1A5D   2389         MVC    GUF924(1),GUF525           CORRECT NEW LINE LENGTH FOR CB1
    10B0 0E 00 1A61 1A76   2390         ALC    GUF531(1),GUF675           INCR CB2 PORTION BY LENGTH OF SDF
    10B6 0F 00 1A43 1A61   2391         SLC    GUF489-GUF411(1),GUF531    DECR CB2 UNUSED SPACE COUNTER
    10BC 0E 01 1050 1A61   2392         ALC    GUF846+@OP1(@CADDR),GUF531  INCR ADDR FOR CB2 DATA FROM CB
    10C2 3C FF 10D3        2393         MVI    GUF861+@Q,GUF615           SET THE LENTGH CODE TO MOVE PART
    10C6 0E 00 10D3 1A61   2394         ALC    GUF861+@Q(1),GUF531        * OF THEN NEW LINE TO CB2
    10CC 0C 00 10D4 1A61   2395         MVC    GUF861+@D1(1),GUF531       SET THE MOVE TO ADDR IN CB2
    10D2 4C 00 00 0000     2396 GUF861 MVC    0(1,@BR),*-*               MOVE PART OF NEW LINE TO CB2
    10D7 0E 00 10DF 10D4   2397         ALC    GUF864+@D1(1),GUF861+@D1   SET ADDR FOR AND FOLLOW WITH
    10DD 4C 03 04 1A68     2398 GUF864 MVC    GUF633(GUF540,@BR),GUF645  * A NULL SEGMENT SDF
    10E2 4C 01 02 1A61     2399         MVC    GUF627(GUF507,@BR),GUF531  SET SDF IN CB2 FOR PART 2 OF
    10E7 4C 01 04 1A59     2400         MVC    GUF633(2,@BR),GUF516       * NEW LINE
    10EC 3C 01 1C02        2401         MVI    GUF924+1,GUF579            SET SDF SEGMENT INDR FOR PART 1
    10F0 C0 87 102C        2402         B      GUF843                     FINISH INSERTION IN COMMON RTN
```

```
                        2404 *        PATCH ,4
                        2405 ********************************************************************
                        2406 *        PATCH AREA 4
                        2407 ********************************************************************
                        2408 *
                        2409 ***     CALCULATE AREA LEFT IN THIS SECTOR
                        2410 *
           10F4 2411 $$$$L4 EQU    *                        START OF PATCH AREA 4
    1100                 2412        ORG    *,256,0           SET LOC CNTR TO NEXT SECTOR
           1100 2413 $$$$T4 EQU    *                        DEFINE ADDR OF SCTR BOUNDARY
    10F4                 2414        ORG    $$$$L4            SET LOC CNTR TO START OF
                        2415 *                               * PATCH AREA
    10F4      10FF 2416 $$$$$4 DS     CL($$$$T4-$$$$L4)      PATCH AREA
                        2417 ********************************************************************
```

```
      #GUFUD C2DEC5 -- CONVERT BINARY TO DECIMAL


ERR LOC   OBJECT CODE     ADDR STMT SOURCE STATEMENT            VER 15, MOD 00  29/02/16  PAGE  57

      0C07                     2419          ORG    GUFUDI                 START OF BUFFER AREA
      0C07              0C07 2420 GUF684 DS     CL1                    BYTE 1 OF 1ST CORE BLOCK
                        0C08 2421 GUF687 EQU    *                      BYTE 0 OF 1ST SDF IN CB1
      0C08              0D06 2422        DS     CL255                  REMAINDER OF CORE BLOCK 1

      0D07              0D07 2424 GUF690 DS     CL1                    BYTE 1 OF 1ST CORE BLOCK
                        0D08 2425 GUF693 EQU    *                      BYTE 0 OF 1ST SDF IN CB2
      0D08              0E06 2426        DS     CL255                  REMAINDER OF CORE BLOCK 2

      0E07              0E07 2428 GUF696 DS     CL1                    BYTE 1 OF 1ST CORE BLOCK
                        0E08 2429 GUF699 EQU    *                      BYTE 0 OF 1ST SDF IN CB3
      0E08              0F06 2430        DS     CL255                  REMAINDER OF CORE BLOCK 3

      0F07              0F07 2432 GUF702 DS     CL1                    BYTE 1 OF 1ST CORE BLOCK
                        0F08 2433 GUF705 EQU    *                      BYTE 0 OF 1ST SDF IN CB4
      0F08              1006 2434        DS     CL255                  REMAINDER OF CORE BLOCK 4

      1007              1106 2436 GUF708 DS     CL256                  256 BYTES WORK AREA
                             2437 *
                             2438 ***    CRUSHER SECTION OF GUFUDI
                             2439 *
                        1107 2440 GUFCSH EQU    *                      ENTRY TO CRUSHER
                        1107 2441 GUGENT EQU    *                      ENTRY TO CRUSHER
      1107 3A 08 03C3        2442        SBN    $KEYCD,$GUFIR          SET FOR NO FUNC INTERRUPTED MSG
      110B 3B 70 03D5        2443        SBF    $INDR2,$FCIND+$FDIND+$FUIND  SET INDR FOR CRUSH ONLY
      110F F2 87 08          2444 GUF000 J      GUF009                 BYPASS WAIT LOOP UNLESS INST
                        1110 2445 GUF003 EQU    GUF000+@Q              * HAS BEEN MODIFIED FOR NO CRUSH
      1112 C0 87 129D        2446 GUF006 B      GUF087                 GO CHECK FOR SERVICE CALL
      1116 C0 87 1112        2447        B      GUF006                 IF NO, GO CHECK AGAIN
      111A 35 02 1A6E        2448 GUF009 L      GUF663,@XR             LOAD ADDR OF FIT ENTRY TO @XR
      111E 38 80 03E0        2449        TBN    $DBGUF,$CRUSH          TEMP TO KILL CRUSHER IF
      1122 F2 10 15          2450        JT     GUF015                 * INDR IS NOT SET ON !!!
      1125 8D 01 03 1A1F     2451 GUF012 CLC    GUF654(GUF507,@XR),GUF417  IS THIS LAST FIT ENTRY USED ?
      112A F2 81 0D          2452        JE     GUF015                 YES, GO CHECK FOR SERVICE CALL
      112D BD 07 04          2453        CLI    GUF657(,@XR),GUF591    NO, IS NULL SEG GT MIN 'PAK' ?
      1130 F2 84 0E          2454        JH     GUF018                 YES, GO RUSH
      1133 E2 02 04          2455        LA     GUF540(,@XR),@XR       NO, INCR XR TO CHECK NEXT ENTRY
      1136 C0 87 1125        2456        B      GUF012                 *
      113A C0 87 129D        2457 GUF015 B      GUF087                 GO CHECK FOR SERVICECALL, IF NO,
      113E F2 87 50          2458        J      GUF027                 * GO CHECK DISK ADDR ORDER
                             2459 *
                             2460 ***    DB REFERENCED HAS 'PAK'ABLE SPACE AND IS NOT THE LAST DB
                             2461 *
      1141 34 02 1D09        2462 GUF018 ST     GUF942,@XR             SAVE XR FOR WRITE ROUTINE
      1145 C0 87 129D        2463        B      GUF087                 GO CHECK FOR SERVICE CALL
      1149 3C 03 1A28        2464        MVI    GUF438,GUF594          MOVE 3 TO 'PAK' COUNTER
      114D 3C 00 1A52        2465        MVI    GUF501,@ZERO           SET DB'S READ COUNTER TO ZERO
      1151 3C 13 1464        2466        MVI    GUF195,GUF558          NO SC., SET DISK READ PARM
      1155 C0 87 1389        2467        B      GUF138                 GO READ 3 DB'S TO CB1-3
      1159 0C 01 1A35 1A57   2468        MVC    GUF468(@CADDR),GUF513  CALC DISP OF 1ST FREE SPACE
      115F 2F 00 1A35 04     2469        SLC    GUF468,GUF657(1,@XR)   * IN CB1
      1164 2C 00 1A37 04     2470        MVC    GUF471,GUF657(1,@XR)   SET FREE SPACE COUNTER
      1169 2C 03 1A3B 04     2471        MVC    GUF486,GUF657(GUF540,@XR)  MOVE FIT ENTRY TO CIT WORK AREA
      116E C0 87 0025        2472        B      $DISKN                 WAIT FOR DISK COMPLETE
      1172 057F         1173 2473        DC     AL2($WAITF)            *
      1174 C2 02 0C07        2474        LA     GUF684,@XR             SEARCH DB1 FOR NULL SEGMENT
```

```
     #GUFUD C2DEC5 -- CONVERT BINARY TO DECIMAL
```

```
     1178 B8 80 01            2475 GUF021 TBN   GUF624(,@XR),GUF546        * SAVE THE ADDR OF THE SDF
     117B F2 10 0B            2476        JT    GUF024                     * OF THE LAST VALID SEGMENT
     117E 34 02 1A51          2477        ST    GUF498,@XR                 *
     1182 B6 02 02            2478        A     GUF627(,@XR),@XR           *
     1185 C0 87 1178          2479        B     GUF021                     *
     1189 C0 87 1522          2480 GUF024 B     GUF201                     GO TO 'PAK' ROUTINE
     118D C0 87 1107          2481        B     GUGENT                     GO TO GARBAGE COLLECTION ENTRY
                              2482 *
                              2483 ***   FILE IS COMPLETELY 'PAK'ED - CHECK PHYSICAL ORDER OF DISK BLOCKS
                              2484 *
     1191 35 02 1A6E          2485 GUF027 L     GUF663,@XR                 LOAD ADDR OF FIT TO XR
     1195 38 40 03E0          2486        TBN   $DBGUF,$REORD              TEMP TO KILL RE-ORDER IF
     1199 C0 10 113A          2487        BT    GUF015                     INDR IS NOT SET ON !!!
     119D 34 02 1A2B          2488 GUF030 ST    GUF447,@XR                 STORE XR TO CHECK ADDR
     11A1 0D 01 1A2B 1A74     2489        CLC   GUF447(@CADDR),GUF672      END OF FIT
     11A7 C0 81 113A          2490        BE    GUF015                     YES, GO LOOP UNTIL CI. OR CR.
     11AB 8D 01 03 1A1F       2491        CLC   GUF654(GUF507,@XR),GUF417  LAST FIT ENTRY USED ?
     11B0 C0 81 113A          2492        BE    GUF015                     YES, GO CHECK FOR SERVICE CALL
     11B4 AD 00 05 01         2493        CLC   GUF651+GUF540(1,@XR),GUF651(,@XR) IS DB2 OUT OF ORDER ?
     11B8 F2 82 07            2494        JL    GUF033                     YES, GO EXCHANGE ORDER
     11BB E2 02 04            2495        LA    GUF540(,@XR),@XR           NO, INCR XR TO NEXT ENTRY
     11BE C0 87 119D          2496        B     GUF030                     BRANCH TO CHECK IF USED
                              2497 *
                              2498 ***   FILE IS OUT OF PHYSICAL ORDER - EXCHANGE THE DISK BLOCKS
                              2499 *
     11C2 36 02 1A78          2500 GUF033 A     GUF678,@XR                 DECR. XR TO POINT TO PRIOR ENTRY
     11C6 3C 14 1464          2501        MVI   GUF195,GUF597              READ 4 DB'S TO CB1-CB4
     11CA C0 87 1389          2502        B     GUF138                     *
                   11F3       2503        USING GUF036,@BR                 SET LOCAL BASE ADDRESSING
     11CE C2 01 11F3          2504        LA    GUF036,@BR                 *
     11D2 AD 00 0D 05         2505        CLC   GUF651+12(1,@XR),GUF651+4(,@XR)  THE POSITIONING OF THE 1ST
     11D6 D0 84 07            2506        BH    GUF039(,@BR)               * DB CAN NOT BE CHANGED BECAUSE
     11D9 6C 00 A1 0D         2507        MVC   GUF069(1,@BR),GUF651+12(,@XR)   * THE PRIOR DB IS LINKED
     11DD AC 00 0D 05         2508        MVC   GUF651+12(1,@XR),GUF651+4(,@XR) * TO IT
     11E1 6D 00 A1 09         2509        CLC   GUF069(1,@BR),GUF651+8(,@XR)    * THE DISK ADDR DISPLACEMENT
     11E5 D0 82 00            2510        BL    GUF036(,@BR)               * OF THE OTHER 3 DB'S ARE
     11E8 AC 00 05 09         2511        MVC   GUF651+4(1,@XR),GUF651+8(,@XR)  * SORTED INTO ASCENDING
     11EC 9C 00 09 A1         2512        MVC   GUF651+8(1,@XR),GUF069(,@BR)  * ORDER AND PLACED BACK IN
     11F0 D0 87 13            2513        B     GUF042(,@BR)               * TO FIT.

     11F3 9C 00 05 A1         2515 GUF036 MVC   GUF651+4(1,@XR),GUF069(,@BR)  *
     11F7 D0 87 13            2516        B     GUF042(,@BR)               *
     11FA 6C 00 A1 05         2517 GUF039 MVC   GUF069(1,@BR),GUF651+4(,@XR)    *
     11FE AC 00 05 09         2518        MVC   GUF651+4(1,@XR),GUF651+8(,@XR)  *
     1202 9C 00 09 A1         2519        MVC   GUF651+8(1,@XR),GUF069(,@BR)  *
     1206 3C 04 1A28          2520 GUF042 MVI   GUF438,GUF600              SET BLOCK COUNTER TO 4
     120A C0 87 0025          2521        B     $DISKN                     WAIT FOR DISK COMPLETE
     120E 057F           120F 2522        DC    AL2($WAITF)                *
     1210 D0 87 24            2523        B     GUF045(,@BR)               GO WRITE TO DISK
     1213 C0 87 113A          2524        B     GUF015                     GO CHECK FOR SERVICE REQUEST

                              2526 *
                              2527 ***   DISK WRITE ROUTINE
                              2528 *
                   1217       2529 GUF045 EQU   *                          ENTRY POINT
     1217 34 08 1293          2530        ST    GUF066+@OP1,@ARR           SAVE RETURN ADDRESS
```

```
     #GUFUD C2DEC5 -- CONVERT BINARY TO DECIMAL
```

```
    121B 4C 01 81 1A64          2531         MVC    GUF060+@OP1(@CADDR,@BR),GUF642   INIT LINK INDR MOVE
    1220 5C 01 A9 81            2532 GUF048 MVC    GUF084(@CADDR,@BR),GUF060+@OP1(,@BR)   MOVE CADDR TO DPL
    1224 7C 03 A6               2533         MVI    GUF078(,@BR),GUF585             SET SECTOR ADDESS FOR LOGICAL
    1227 6E 00 A6 01            2534         ALC    GUF078(1,@BR),GUF651(,@XR)    * IOCS IN DPL.
    122B 7C 01 A7               2535         MVI    GUF081(,@BR),GUF603            INITIALIZE SECTOR COUNT IN DPL
    122E 6C 00 A1 01            2536         MVC    GUF069(1,@BR),GUF651(,@XR)    MOVE DADDR DISP TO WORK AREA
    1232 4E 00 A1 1A55          2537 GUF051 ALC    GUF069(1,@BR),GUF510          ADD 1 TO IT
    1237 E2 02 04               2538         LA     GUF540(,@XR),@XR             INCR XR TO POINT TO NEXT DADDR
    123A 34 02 1A2B             2539         ST     GUF447,@XR                   IF LAST DB OF FILE IS BEING
    123E 0D 01 1A2B 1A72        2540         CLC    GUF447(@CADDR),GUF669        * WRITTEN OUT, SET THE LINK
    1244 D0 02 7E               2541         BNL    GUF057(,@BR)                 * CODE TO ZERO
    1247 6D 00 A1 01            2542         CLC    GUF069(1,@BR),GUF651(,@XR)   IS IT THE NEXT PHYSICAL DB ?
    124B D0 81 7E               2543         BE     GUF057(,@BR)                 YES, GO INCR COUNTERS
    124E 5C 01 62 81            2544         MVC    GUF054+@OP1(@CADDR,@BR),GUF060+@OP1(,@BR)   SET UP AND
    1252 2C 00 0000 01          2545 GUF054 MVC    *-*,GUF651(1,@XR)            * LINK INDR TO CB
    1257 3D 01 1A28             2546         CLI    GUF438,GUF603                MORE DADDR DISP TO CHECK ?
    125B D0 81 97               2547         BE     GUF063(,@BR)                 NO, GO WRITE AND GET OUT
    125E C0 87 1466             2548         B      DL4ICS                       YES, WRITE CB CHECKED TO DISK
    1262 1297             1263  2549         DC     AL2(GUF075)                  *
    1264 5E 01 81 A3            2550         ALC    GUF060+@OP1(@CADDR,@BR),GUF072(,@BR)   INCR TO NEXT CB
    1268 0F 00 1A28 1A55        2551         SLC    GUF438(1),GUF510             DECREMENT CB COUNTER
    126E D0 87 2D               2552         B      GUF048(,@BR)                 GO CHECK NEXT DADDR DISP
                          1271  2553 GUF057 EQU    *                            LOGICAL DB IS NEXT PHYSICAL DB
    1271 3C 00 0000            2554 GUF060 MVI    *-*,@ZERO                    MOVE ZERO TO CB LINK INDR
    1275 5E 01 81 A3            2555         ALC    GUF060+@OP1(@CADDR,@BR),GUF072(,@BR)   INCR FOR NEXT CB
    1279 0F 00 1A28 1A55        2556         SLC    GUF438(1),GUF510             DECREMENT CB COUNTER
    127F D0 81 97               2557         BZ     GUF063(,@BR)                 NO MORE CBS, WRITE AND GET OUT
    1282 4E 00 A7 1A55          2558         ALC    GUF081(1,@BR),GUF510         INCR DPL SECTOR COUNT
    1287 D0 87 3F               2559         B      GUF051(,@BR)                 GO CHECK NEXT DADDR DISPLACEMENT
    128A C0 87 1466             2560 GUF063 B      DL4ICS                       GO WRITE CB'S TO DISK
    128E 1297             128F  2561         DC     AL2(GUF075)                  *
    1290 C0 87 0000             2562 GUF066 B      *-*                          RETURN TO CALLER

    1294                  1294  2564 GUF069 DS     CL1                          1 BYTE SAVE AREA & WORK AREA
    1295 0100             1296  2565 GUF072 DC     XL2'0100'                    CONSTANT VALUE - 256 DIZE OF DB
    1297 02               1297  2566 GUF075 DC     AL1(@DPUT)                   DISK WRITE FUNC CODE
    1298 05               1298  2567         DC     AL1(@DWBCY)                  DISK CYLINDER ADDRESS
    1299                  1299  2568 GUF078 DS     CL1                          DISK SECTOR DISPLACEMENT
    129A                  129A  2569 GUF081 DS     CL1                          DISK SECTOR COUNT
    129B                  129C  2570 GUF084 DS     CL2                          DISK OPERATION DATA CORE ADDR

                                2572 *
                                2573 ***    CHECK FOR CARRIER RETURN OR UNQUIRY REQUEST
                                2574 *
                          129D  2575 GUF087 EQU    *                            ENTRY POINT
    129D 34 08 12D2             2576         ST     GUF096+@OP1,@ARR             SAVE RETURN ADDRESS
    12A1 38 01 03C3             2577         TBN    $KEYCD,$CARDI                INPUT SOURCE = CARDS ?
    12A5 F2 10 04               2578         JT     GUF090                       YES, BYPASS UNMASK
    12A8 C0 87 048D             2579         B      $UNMSK                       GO CHECK FOR INTERRUPTS
    12AC 38 01 03C3             2580 GUF090 TBN    $KEYCD,$CARDI                CARD INPUT ?
    12B0 F2 90 0B               2581         JF     GUF093                       NO, GO CHECK END OF INPUT
    12B3 C0 87 08C0             2582         B      $$CDBS                       GO CHECK COMPLETION OF CARD INP
    12B7 38 08 03E0             2583         TBN    $DBGUF,$CALLI                PROCEDURE FILE ?
    12BB F2 10 15               2584         JT     GUF097                       YES, LOOP ONE TIME
    12BE 38 10 03C3             2585 GUF093 TBN    $KEYCD,$KYBSY                INPUT COMPLETE ?
    12C2 F2 90 1A               2586         JF     GUF099                       YES, GO CHECK FOR BLANKS ONLY
```

```
    12C5 C0 87 0025              2587 GUF095 B      $DISKN                    WAIT FOR PRIOR DISK
    12C9 057F            12CA    2588        DC     AL2($WAITF)               * WRITE TO COMPLETE
    12CB 3C 80 0476              2589        MVI    $CIMSK,@NOP               RE-MASK INTERRUPTS
    12CF C0 87 0000              2590 GUF096 B      *-*                       RETURN TO CALLER

    12D3 38 10 03C3              2592 GUF097 TBN    $KEYCD,$KYBSY             TEST BUSY
    12D7 C0 87 12DF              2593 GUF098 BC     GUF099,@UCB               BRANCH ON CONDITION
    12DB C0 87 12C5              2594        B      GUF095                    CONTINUE

                                2596 *
                                2597 ***    INPUT IS COMPLETE - CHECK FOR BLANK LINE ENTERED
                                2598 *
    12DF C0 00 0469              2599 GUF099 BC     $CAERK,*-*                IF AN ERROR OCCURS, THIS INST IS
    12DF                        2600        ORG    GUF099                    * CHANGED FROM A @NOP TO A @UCB
    12DF C0 80 0469              2601        BC     $CAERK,@NOP               * TO GO TO THE ERROR PROGRAM
    12E3 3C 90 12D8              2602        MVI    GUF098+@Q,@BF             RESET FALSE CONDITION
    12E7 38 80 03C3              2603        TBN    $KEYCD,$TRUNK             WAS INPUT LINE TRUNCATED ?
    12EB F2 90 0D                2604        JF     GUF102                    NO, BYPASS ERROR HANDLING
    12EE C2 02 03C0              2605        LA     $NUCBS,@XR                GET #XR OUT OF INPUT BUFFER
                         03C0    2606        USING  $NUCBS,@XR                ESTABLISH BASE REGISTER USAGE
    12F2 BC 40 0E                2607        MVI    $ERRPG(,@XR),$ERFIL       SET UP TRUNCATED
    12F5 BC 8E 0D                2608        MVI    $CAERR(,@XR),@@E540       * ERROR MESSAGE
    12F8 E0 87 A9                2609        B      $CAERK(,@XR)              GO TO ERRPGM INTERFACE
                         0002    2610        DROP   @XR                       DROP BASE REGISTER
    12FB C2 02 0607              2611 GUF102 LA     $$INLN,@XR                POINT @XR TO THE INPUT BUFFER
    12FF 39 0F 0603              2612        TBF    $$CKEY,GUF402             COMMAND KEY HIT ?
    1303 F2 90 39                2613        JF     GUF114                    YES, GO FETCH CA
    1306 BD 40 00                2614 GUF105 CLI    @ZERO(,@XR),@BLANK        IS THIS POSITON A BLANK ?
    1309 F2 01 07                2615        JNE    GUF108                    NO, GO CHECK INPUT TYPE
    130C E2 02 01                2616        LA     @B1(,@XR),@XR             INCR @XR TO NEXT POSITION
    130F C0 87 1306              2617        B      GUF105                    GO CHECK FOR A BLANK
    1313 38 01 03C3              2618 GUF108 TBN    $KEYCD,$CARDI             CARD INPUT ?
    1317 F2 90 1F                2619        JF     GUF111                    NO, GO CHECK FOR EOS CODE
    131A 34 02 1384              2620        ST     GUF129,@XR                SAVE @XR FOR COMPARE
    131E 0D 01 1384 1386         2621        CLC    GUF129(@CADDR),GUF132     CARD ALL BLANK ?
    1324 F2 04 18                2622        JNH    GUF114                    NO, GO TO GET CA
    1327 38 04 03C3              2623        TBN    $KEYCD,$NOLST             CARD NO LIST ?
    132B C0 10 1B7C              2624        BT     GUF912                    YES, GO ENABLE INPUT
    132F C0 87 0465              2625        B      $SPRNT                    ADVANCE CARRIER 1 LINE
    1333 1387            1334    2626        DC     AL2(GUF135)               *
    1335 C0 87 1B7C              2627        B      GUF912                    GO ENABLE INPUT
    1339 BD 1E 00                2628 GUF111 CLI    @ZERO(,@XR),@EOS          BLANKS ONLY ENTERED ?
    133C F2 81 22                2629        JE     GUF120                    YES, GO ENABLE INPUT
    133F 3B 08 03C3              2630 GUF114 SBF    $KEYCD,$GUFIR             SET FOR FUNC INTERRUPTED MSG.
    1343 F2 80 08                2631 GUF115 JC     GUF116,@NOP               EXECUTE THE NEXT 2 INST.
    1346 3C 87 1344              2632        MVI    GUF115+@Q,@UCB            * ONCE, THEN BRANCH AROUND
    134A C0 87 1107              2633        B      GUGENT                    * GO CRUSH ONE TIME
    134E 38 02 03D5              2634 GUF116 TBN    $INDR2,$CMODE             IN UTILITY MODE ?
    1352 F2 90 06                2635        JF     GUF117                    YES, USE $RLOAD TO GET CA
    1355 C0 87 0522              2636        B      $BLOAD                    USE $BLOAD TO GET CA
    1359 1377            135A    2637        DC     AL2(GUF123)               *
    135B C0 87 051E              2638 GUF117 B      $RLOAD                    USE $RLOAD TO GET CA
    135F 137D            1360    2639        DC     AL2(GUF126)               *
    1361 3B 10 03D2              2640 GUF120 SBF    $IOIND,$PGMST             SET FOR AUTO LINE NUMBER
    1365 3C 40 06FA              2641        MVI    $$INND,@BLANK             CLEAR THE INPUT LINE BUFFER
    1369 0C F3 06F9 06FA         2642        MVC    $$INND-@B1(@LINSZ),$$INND * TO BLANKS
```

```
     136F C0 87 0890            2643        B     $$PRES                    ENABLE INPUT
     1373 C0 87 1107            2644        B     GUGENT                    GO CRUSH
     1377 01              1377 2645 GUF123 DC    AL1(@DGET)                DPL TO GET CA
     1378 0481            1379 2646        DC    AL2(#@ECMA)               *
     137A 06              137A 2647        DC    AL1(#@@ECM)               *
     137B 0C00            137C 2648        DC    AL2(#$$ECM)               *


     137D 01              137D 2650 GUF126 DC    AL1(@DGET)                DPL FOR GET UTILITY MODE CA
     137E 1900            137F 2651        DC    AL2(#$ECMA)               *
     1380 06              1380 2652        DC    AL1(#$@ECM)               *
     1381 0C00            1382 2653        DC    AL2(#$$ECM)               *


     1383                 1384 2655 GUF129 DS    CL2                       SAVE AREA FOR @XR FOR CARD COMP
     1385 0666            1386 2656 GUF132 DC    AL2($$CDND)               LAST POSITION OF CARD INPUT
     1387 80              1387 2657 GUF135 DC    AL1(@RETRN)               PPL TO ADVANCE CARRIER 1 LINE
     1388 80              1388 2658        DC    AL1(@RETRN)               *


                              2660 *
                              2661 ***    COMMON DISK READ ROUTINE
                              2662 *
                         13BC 2663        USING GUF141,@BR               SET LOCAL BASE
                         1389 2664 GUFDIN EQU   *                         READ ROUTINE ENTRY
                         1389 2665 GUF138 EQU   *                         READ ROUTINE ENTRY
     1389 34 01 1459          2666        ST    GUF171+@OP1,@BR           SAVE BR
     138D C2 01 13BC          2667        LA    GUF141,@BR                LOAD LOCAL BASE
     1391 74 02 99            2668        ST    GUF168+@OP1(,@BR),@XR     SAVE XR
     1394 74 08 A1            2669        ST    GUF174+@OP1(,@BR),@ARR    SAVE RETURN ADDR
     1397 4C 01 74 1A64       2670        MVC   GUF159+@OP1(@CADDR,@BR),GUF642  INIT ADDR OF 1ST CB
     139C 4C 01 2B 1A4D       2671        MVC   GUF153+@OP1(@CADDR,@BR),GUF492  INIT ADDR OF 1ST CIT
     13A1 78 20 A8            2672        TBN   GUF195(,@BR),GUF606       IS DB1 TO BE READ TO CB1 ?
     13A4 D0 90 05            2673        BF    GUF144(,@BR)              YES, GO SHUT OFF INDR
     13A7 4E 01 74 1296       2674        ALC   GUF159+@OP1(@CADDR,@BR),GUF072  INCR FOR CB2
     13AC 78 10 A8            2675        TBN   GUF195(,@BR),GUF609       IS DB1 TO BE READ TO CB2
     13AF D0 90 00            2676        BF    GUF141(,@BR)              YES, GO INCR CIT POINTER
     13B2 4E 01 74 1296       2677        ALC   GUF159+@OP1(@CADDR,@BR),GUF072  INCR FOR CB3
     13B7 4E 01 2B 1A76       2678        ALC   GUF153+@OP1(@CADDR,@BR),GUF675  INCR TO NEXT CIT ENTRY
     13BC 4E 01 2B 1A76       2679 GUF141 ALC   GUF153+@OP1(@CADDR,@BR),GUF675  INCR TO NEXT CIT ENTRY
     13C1 7B F0 A8            2680 GUF144 SBF   GUF195(,@BR),GUF612       TURN OFF CB INDR
     13C4 7C 00 A5            2681 GUF147 MVI   GUF186(,@BR),@ZERO        CLEAR SECTOR COUNT
     13C7 5C 01 A7 74         2682        MVC   GUF189(@CADDR,@BR),GUF159+@OP1(,@BR)  SET CADDR IN DPL
     13CB 34 02 1A2D          2683        ST    GUF453,@XR                IS REFENCED DB NULL ?
     13CF 0D 01 1D0B 1A2D     2684        CLC   GUF945(@CADDR),GUF453     *
     13D5 D0 04 6C            2685        BNH   GUF156(,@BR)              YES, GO SET MATCHING CB TO NULL
     13D8 7C 03 A4            2686        MVI   GUF183(,@BR),GUF585       SET SECTOR ADDRESS FOR
     13DB 6E 00 A4 01         2687        ALC   GUF183(1,@BR),GUF651(,@XR)  * LOGICAL IOCS IN DPL
     13DF 4E 00 A5 1A55       2688 GUF150 ALC   GUF186(1,@BR),GUF510      INCR DPL SECTOR COUNT
     13E4 2C 03 0000 04       2689 GUF153 MVC   *-*,GUF657(GUF540,@XR)    MOVE FIT ENTRY TO CIT SLOT
     13E9 4E 01 2B 1A76       2690        ALC   GUF153+@OP1(@CADDR,@BR),GUF675  INCR TO NEXT CIT ENTR
     13EE 4F 00 A8 1A55       2691        SLC   GUF195(1,@BR),GUF510      DECR CB COUNTER
     13F3 D0 81 83            2692        BZ    GUF162(,@BR)              ALL CB'S CHECKED, GO READ THEM
     13F6 4E 01 74 1296       2693        ALC   GUF159+@OP1(@CADDR,@BR),GUF072  INCR FOR NEXT CB
     13FB 6C 00 A9 01         2694        MVC   GUF198(1,@BR),GUF651(,@XR)  SAVE DADDR DISPLACEMENT
     13FF 4E 00 A9 1A55       2695        ALC   GUF198(1,@BR),GUF510      ADD 1 TO LAST DADDR DISPLACEMENT
     1404 E2 02 04            2696        LA    GUF540(,@XR),@XR          INCR CR FOR NEXT DB
     1407 34 02 1A2D          2697        ST    GUF453,@XR                IS NEXT DB NULL
     140B 0D 01 1D0B 1A2D     2698        CLC   GUF945(@CADDR),GUF453     *
```

```
     1411 D0 04 6C            2699          BNH   GUF156(,@BR)                YES, GO SET MATCHING CB TO NULL
     1414 6D 00 A9 01         2700          CLC   GUF198(1,@BR),GUF651(,@XR)  ARE DB'S IN PHYSICAL ORDER ?
     1418 D0 81 23            2701          BE    GUF150(,@BR)                YES, GO CONTINUE CHECKING
     141B D0 87 AA            2702          B     DL4ICS(,@BR)                GO TO LOGICAL DISK IOCS TO READ
     141E 145E          141F 2703          DC    AL2(GUF177)                 * IN ORDER DB'S
     1420 1E 00 1A52 A5       2704          ALC   GUF501,GUF186(1,@BR)        INCR NUMBER OF DB'S READ
     1425 D0 87 08            2705          B     GUF147(,@BR)                GO SET DPL FOR NEXT LOGICAL DB
     1428 4E 01 74 1A76       2706 GUF156 ALC   GUF159+@OP1(@CADDR,@BR),GUF675  MOVE A NULL SEG. SDF TO
     142D 0C 03 0000 1A68     2707 GUF159 MVC   *-*,GUF645(GUF540)          * CB MATCHING NULL DB
     1433 0F 00 1A28 1A55     2708          SLC   GUF438(1),GUF510            DECR NO. OF CB'S TO BE 'PAK'ED
     1439 7D 00 A5            2709          CLI   GUF186(,@BR),@ZERO          IS DPL SECTOR COUNT ZERO ?
     143C F2 81 0D            2710          JE    GUF165                      YES, GO WAIT AND GET OUT
     143F D0 87 AA            2711 GUF162 B     DL4ICS(,@BR)                GO TO LOGICAL DISK IOCS TO READ
     1442 145E          1443 2712          DC    AL2(GUF177)                 * REST OF DB'S
     1444 1E 00 1A52 A5       2713          ALC   GUF501,GUF186(1,@BR)        INCR NUMBER OF DB'S READ
     1449 F2 87 06            2714          J     GUF168                      BYPASS WAIT OP
     144C C0 87 0025          2715 GUF165 B     $DISKN                      GO TO DISK IOCS TO WAIT FOR
     1450 057F           1451 2716          DC    AL2($WAITF)                 * PRIOR READ TO COMPLETE
     1452 C2 02 0000          2717 GUF168 LA    *-*,@XR                     RESTORE XR
     1456 C2 01 0000          2718 GUF171 LA    *-*,@BR                     RESTORE BR
     145A C0 87 0000          2719 GUF174 B     *-*                         RETURN TO CALLER

                         145E 2721 GUF177 EQU   *                           DISK READ PARAMETER LIST
     145E 01              145E 2722 GUF180 DC    AL1(@DGET)                  * FUNCTION CODE
     145F 05              145F 2723          DC    AL1(@DWBCY)                 * DISK CYLINDER ADDR
     1460                1460 2724 GUF183 DS    CL1                         * DISK SECTOR DISPLACEMENT
     1461                1461 2725 GUF186 DS    CL1                         * SECTOR COUNT
     1462                1463 2726 GUF189 DS    CL2                         * CORE ADDR
                         145F 2727 GUF192 EQU   GUF183-1                    * CYLINDER NUMBER
     1464                1464 2728 GUF195 DS    CL1                         DISK READ PARAMETER
     1465                1465 2729 GUF198 DS    CL1                         1 BYTE WORK AREA
                              2730 *
```

```
                        2732 ****************************************************************************
                        2733 * 5703-XM1  COPYRIGHT IBM CORP. 1970                            *
                        2734 *            REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083 *
                        2735 *                                                               *
                        2736 ****************************************************************************
                        2737 *STATUS                                                         *
                        2738 *    VERSION 1 MODIFICATION 0                                    *
                        2739 *                                                               *
                        2740 *FUNCTION                                                        *
                        2741 *    * DL4ICS WILL CONVERT A RELATIVE DISK ADDRESS TO A PHYSICAL *
                        2742 *      DISK ADDRESS AND CALL $DISKN TO PERFORM THE SPECIFIED FUNCTION *
                        2743 *    * THE DISK ADDRESS IS A ONE BYTE CYLINDER ADDRESS AND A ONE BYTE *
                        2744 *      SECTOR DISPLACEMENT RELATIVE TO SECTOR 0 ON A CYLINDER   *
                        2745 *      BOUNDARY                                                  *
                        2746 *    * WHEN MORE THAN 1 SECTOR IS PROCESSED, DL4ICS WILL MAKE MULTIPLE *
                        2747 *      CALLS TO $DISKN TO CROSS CYLINDER BOUNDARIES IF REQUIRED. *
                        2748 *    * IF 1 SECTOR ONLY IS TO BE PROCESSED, THE USER MAY OVERLAY THE *
                        2749 *      UNUSED CODE BY ORGING HIS NEXT MODULE AT DL4SPT           *
                        2750 *                                                               *
                        2751 *ENTRY POINTS                                                    *
                        2752 *     DL4ICS - ENTRY TO PROCESS A 4 SURFACE FILE. THE CALLING    *
                        2753 *              SEQUENCE IS AS FOLLOWS                            *
                        2754 *                 DSKL4  DPL                                     *
                        2755 *              WHERE DPL IS THE LABEL OF A SIX BYTE DISK PARAMETER *
                        2756 *              LIST AS DESCRIBED FOR $DJSKN EXCEPT FOR THE SECTOR *
                        2757 *              ADDRESS BYTE.                                     *
                        2758 *                                                               *
                        2759 *INPUT                                                           *
                        2760 *    * INPUT TO DL4ICS IS THE ADDRESS OF THE DPL TO BE PROCESSED. *
                        2761 *                                                               *
                        2762 *OUTPUT                                                          *
                        2763 *   * N/A                                                        *
                        2764 *                                                               *
                        2765 *EXTERNAL REFENECES                                              *
                        2766 *     $DISKN - ENTRY TO SYSTEM DISK ROUTINE                      *
                        2767 *                                                               *
                        2768 *EXITS, NORMAL                                                   *
                        2769 *    * NORMAL RETURN IS TO THE 1ST INSTRUCTION FOLLOWING THE TWO BYTE *
                        2770 *      ADDRESS POINTING TO THE DPL.                              *
                        2771 *                                                               *
                        2772 *EXITS, ERROR                                                    *
                        2773 *   * N/A                                                        *
                        2774 *                                                               *
                        2775 *TABLES/WORK AREAS                                               *
                        2776 *   * N/A                                                        *
                        2777 *                                                               *
                        2778 *ATTRIBUTES                                                      *
                        2779 *   * RELOCATABLE                                                *
                        2780 *   * REUSABLE                                                   *
                        2781 *                                                               *
                        2782 *CHARACTER CODE DEPENDENCY                                       *
                        2783 *   * THE OPERATION OF THIS MODULE DOES NOT DEPEND UPON A PARTICULAR *
                        2784 *     INTERNAL REPRESENTATION OF THE EXTERNAL CHARACTER SET.     *
                        2785 *                                                               *
                        2786 *NOTES                                                           *
                        2787 *    ERROR PROCEDURES                                            *
```

```
                              2788 *       N/A                                                      *
                              2789 *    REGISTER USAGE                                              *
                              2790 *       @BR IS SAVED AND RESTORED ON EXIT, @XR IS NOT USED.  @ARR IS   *
                              2791 *       USED TO PROVIDE THE ADDRESS OF THE PARAMETER.  THE @ARR IS     *
                              2792 *       INCREMENTED BT TWO AND SAVED AS THE RETURN ADDRESS.           *
                              2793 *    SAVED/RESTORED AREAS                                        *
                              2794 *       N/A                                                      *
                              2795 *    MODIFICATION CONSIDERATIONS                                 *
                              2796 *       N/A                                                      *
                              2797 *    REQUIRED MODULES                                            *
                              2798 *       @SYSEQ - SYSTEM SOFTWARE EQUATES                          *
                              2799 *       @FXDEQ - SYSTEM NUCLEUS EQUATES                           *
                              2800 *    OTHER                                                       *
                              2801 *       N/A                                                      *
                              2802 ****************************************************************************
```

```
                         1466 2804 DL4ICS EQU    *                        ENTRY TO DL4ICS
                         146A 2805        USING  DL4010,@BR               ESTABLISH BASE REGISTER USAGE
      1466 34 01 14D6         2806        ST     DL4900+@OP1,@BR          SAVE BASE REGISTER FOR EXIT
                         146A 2807 DL4010 EQU    *                        BASE ADDRESSABILITY
      146A C2 01 146A         2808        LA     DL4010,@BR               ESTABLISH BASE
      146E 76 08 78           2809        A      DL4C01(,@BR),@ARR        BUMP TO HIGH END OF ADDR
      1471 74 08 14           2810        ST     DL4020+@DOP2(,@BR),@ARR  SET UP MOVE INSTRUCTION
      1474 76 08 78           2811        A      DL4C01(,@BR),@ARR        BUMP TO RETURN ADDR
      1477 74 08 70           2812        ST     DL4920+@OP1(,@BR),@ARR   SAVE RETURN ADDR
                              2813 *
      147A 4C 01 1D 0000      2814 DL4020 MVC    DL4030+@DOP2(@DADDR,@BR),*-* MOVE DPL ADDR INTO MOVE
      147F 5E 01 1D 7A        2815        ALC    DL4030+@DOP2(@CADDR,@BR),DL4C05(,@BR)  BUMP TO RIGHT END
      1483 4C 05 76 0000      2816 DL4030 MVC    DL4DPL(@DPLNG,@BR),*-*   MOVE USER DPL TO WORK AREA
                              2817 *
      1488 7C 00 5E           2818 DL4035 MVI    DL4100+@Q(,@BR),@ZERO    CLEAR TRACK, DISK SET INST
      148B 7C 80 67           2819        MVI    DL4200+@Q(,@BR),@NOP     TURN OFF TWICE INDICATOR
                              2820 *
      148E 7D 60 73           2821 DL4040 CLI    DL4SCD(,@BR),DL4E96      TEST IF DISPLACEMENT OVER 95 ?
      1491 F2 82 0B           2822        JL     DL4050                   JUMP IF NOT OVER 95
      1494 5E 00 72 78        2823        ALC    DL4CYL(1,@BR),DL4C01(,@BR)  INCREMENT CYLINDER COUNT
      1498 5F 00 73 25        2824        SLC    DL4SCD(1,@BR),DL4C96(,@BR)  DECREMENT DISP BY 96
      149C D0 87 24           2825        B      DL4040(,@BR)             GO BACK CHECK FOR NEXT CYLINDER
                              2826 *
      149F 7D 30 73           2827 DL4050 CLI    DL4SCD(,@BR),DL4E48      TEST IF DISP ON NEXT DISK ?
      14A2 F2 82 07           2828        JL     DL4060                   JUMP IF NOT OVER 48
      14A5 7A 01 5E           2829        SBN    DL4100+@Q(,@BR),DL4EFD   TURN ON BIT FOR FIXED DISK
      14A8 5F 00 73 36        2830        SLC    DL4SCD(1,@BR),DL4C48(,@BR)  DECREMENT DISP 1 DISK
      14AC 7D 01 74           2831 DL4060 CLI    DL4SCT(,@BR),DL4E01      IS SECTOR COUNT GREATER THEN 1 ?
      14AF F2 84 33           2832        JH     DL4SPT                   GO TO SPLIT CALL
      14B2 7D 18 73           2833 DL4070 CLI    DL4SCD(,@BR),DL4E24      DISPLACEMENT OVER 23 ?
      14B5 F2 82 07           2834        JL     DL4080                   JUMP NOT OVER 24
      14B8 7A 80 5E           2835        SBN    DL4100+@Q(,@BR),DL4ETB   SET TRACK BIT ON
      14BB 5F 00 73 49        2836        SLC    DL4SCD(1,@BR),DL4C24(,@BR)  DECR DISP TO NEXT TRACK
      14BF 5E 00 73 73        2837 DL4080 ALC    DL4SCD(1,@BR),DL4SCD(,@BR)  SHIFT LEFT 1 PLACE
      14C3 5E 00 73 73        2838        ALC    DL4SCD(1,@BR),DL4SCD(,@BR)  SHIFT LEFT 1 PLACE
      14C7 7A 00 73           2839 DL4100 SBN    DL4SCD(,@BR),*-*         SET TRACK, DISK BIT
                              2840 *
      14CA C0 87 0025         2841        B      $DISKN                   GO PERFORM DISK I/O
      14CE 14DB          14CF 2842        DC     AL2(DL4LST)              ADDR OF DISK PARAM LIST
                              2843 *
      14D0 F2 00 3C           2844 DL4200 JC     DL4600,*-*               BRANCH OR NOP IF TWICE SET
                              2845 *
      14D3 C2 01 0000         2846 DL4900 LA     *-*,@BR                  RESTORE OLD BASE TO RETURN
      14D7 C0 87 0000         2847 DL4920 B      *-*                      RETURN TO CALLER

                         14DB 2849 DL4LST EQU    *                        LEFT END OF DPL
      14DB               14E0 2850 DL4DPL DS     CL(@DPLNG)               DPL SAVE AREA
                         14DC 2851 DL4CYL EQU    DL4LST+@DCYL             CYLINDER COUNT BYTE
                         14DD 2852 DL4SCD EQU    DL4LST+@DSAD             DISPLACEMENT SECTOR COUNT
                         0060 2853 DL4E96 EQU    96                       TWO DISK SECTOR COUNT PER CYL
                         0030 2854 DL4E48 EQU    48                       ONE DISK SECTOR COUNT PER CYL
                         0018 2855 DL4E24 EQU    24                       TRACK SECTOR COUNT
                         0001 2856 DL4E01 EQU    01                       VALUE TO TEST SECTOR COUNT
                         0001 2857 DL4EFD EQU    01                       VALUE TO SET FIXED DISK BIT
                         0080 2858 DL4ETB EQU    X'80'                    VALUE TO SET TRACK BIT
      14E1 0001          14E2 2859 DL4C01 DC     IL2'1'                   VALUE TO INCR TO CYLINDER
```

```
       DL4ICS - 4 SURFACE DISK IOCS ROUTINE


ERR LOC   OBJECT CODE       ADDR STMT SOURCE STATEMENT         VER 15, MOD 00  29/02/16  PAGE  66

     14E3 0005                   14E4 2860 DL4C05 DC    IL2'5'                   DISP TO RIGHT END OF DPL
                                 148F 2861 DL4C96 EQU   DL4040+@Q                VALUE TO DECR DISPLACEMENT
                                 14B3 2862 DL4C24 EQU   DL4070+@Q                VALUE OF 1 TRACK
                                 14DE 2863 DL4SCT EQU   DL4LST+@DCNT             POINTER TO DPL SECTOR COUNT
                                 14A0 2864 DL4C48 EQU   DL4050+@Q                VALUE TO DECR DISP BY 1 DISK

     14E5 5C 00 14 74                2866 DL4500 MVC   DL4WRK(1,@BR),DL4SCT(,@BR) PICKUP SECTOR COUNT
                                 14E5 2867 DL4SPT EQU   DL4500                   POSSIBLE OVERLAY REFERENCE
     14E9 5E 00 14 73                2868        ALC   DL4WRK(1,@BR),DL4SCD(,@BR) BUMP BY DISPLACEMENT
     14ED 7D 30 14                   2869        CLI   DL4WRK(,@BR),DL4E48      TEST FOR CYLINDER OVERLAP
     14F0 D0 04 48                   2870        BNH   DL4070(,@BR)             BRANCH BACK IF NO OVERLAY
     14F3 5F 00 14 36                2871        SLC   DL4WRK(1,@BR),DL4C48(,@BR) DECREMENT WORK BY 48
     14F7 5F 00 74 14                2872        SLC   DL4SCT(1,@BR),DL4WRK(,@BR) SUBTRACT WORK FROM COUNT
     14FB 7C 87 67                   2873        MVI   DL4200+@Q(,@BR),@UCB     SET TWICE SWITCH
     14FE 5C 00 13 73                2874        MVC   DL4SAV(1,@BR),DL4SCD(,@BR) SAVE SECTOR DISP IN WORK AREA
     1502 78 01 5E                   2875        TBN   DL4100+@Q(,@BR),DL4EFD   DISK BIT ON IN Q CODE ?
     1505 D0 90 48                   2876        BF    DL4070(,@BR)             BRANCH NOT ON
     1508 5E 00 13 36                2877        ALC   DL4SAV(1,@BR),DL4C48(,@BR) BUMP TO NEXT DISK
     150C D0 87 48                   2878        B     DL4070(,@BR)             RETURN TO CALL I/O
                                     2879 *
     150F 5C 00 73 13                2880 DL4600 MVC   DL4SCD(1,@BR),DL4SAV(,@BR) PICKUP NEXT HALF OF I/O
     1513 5E 00 75 74                2881        ALC   DL4LST+@DBFR1(1,@BR),DL4SCT(,@BR)  BUMP CORE ADDRESS
     1517 5E 00 73 74                2882        ALC   DL4SCD(1,@BR),DL4SCT(,@BR)
     151B 5C 00 74 14                2883        MVC   DL4SCT(1,@BR),DL4WRK(,@BR) MOVE IN NEW SECTOR COUNT
     151F D0 87 1E                   2884        B     DL4035(,@BR)             RETURN FOR SECOND PASS
                                     2885 *
                                 147E 2886 DL4WRK EQU   DL4020+@DOP2             1 BYTE WORK AREA FOR SPLIT CALL
                                 147D 2887 DL4SAV EQU   DL4020+@DOP2-1           1 BYTE WORK AREA FOR SPLIT CALL
                                 1522 2888 DL4END EQU   *                        DEFINE END OF CODE

                                     2890 *
                                     2891 ***    COMMON 'PAK' ROUTINE
                                     2892 *
                                 1522 2893 GUFPAK EQU   *                        PAK ENTRY POINT
                                 1522 2894 GUF201 EQU   *                        PAK ENTRY POINT
     1522 34 08 18DB                 2895        ST    GUF354+@OP1,@ARR         SAVE RETURN
     1526 35 01 1A64                 2896        L     GUF642,@BR               SET BR TO POINT TO THE LAST BYTE
     152A 34 01 1A25                 2897        ST    GUF432,@BR               * OF VALID DATA. INITIALIZE CB
     152E 34 01 1A27                 2898        ST    GUF435,@BR               *
     1532 36 01 1A35                 2899        A     GUF468,@BR               * STARTING ADDRESS POINTER
     1536 D2 02 00                   2900        LA    @ZERO(,@BR),@XR          POINT XR TO THE LAST BYTE OF
     1539 36 02 1A37                 2901        A     GUF471,@XR               * FREE SPACE AVAILABLE
     153D 0C 01 1742 1A4D            2902        MVC   GUF474(@CADDR),GUF492    SET RTN ADDR FOR CIT WORK AREA
     1543 3C 00 1A53                 2903        MVI   GUF504,@ZERO             SET 'PAK'ED COUNTER TO ZERO
     1547 0E 00 1A35 1A37            2904        ALC   GUF468(1),GUF471         SET DISP OF NEXT VALID DATA
     154D 3D FB 1A35                 2905 GUF204 CLI   GUF468,GUF552            ANY MORE DATA IN THIS CB ?
     1551 F2 02 6E                   2906        JNL   GUF228                   NO, GO TO NEXT CB
     1554 BD 80 01                   2907        CLI   GUF624(,@XR),GUF546      IS THE DATA NULL ?
     1557 F2 81 68                   2908        JE    GUF228                   YES, GO TO NEXT
     155A 2E 00 1A35 02              2909        ALC   GUF468,GUF627(1,@XR)     UPDATE NEXT VALID DATA DISP
     155F 38 04 1A23                 2910        TBN   GUF429,GUF576            IS A RANGE BEING DELETED ?
     1563 F2 10 3A                   2911        JT    GUF222                   YES, GO CHECK LINE NUMBER
     1566 38 01 1A29                 2912 GUF207 TBN   GUF441,@B1               IS A PARTIAL SEG. TO BE MOVED ?
     156A F2 90 07                   2913        JF    GUF210                   NO, GO USE LINE NUMBER IN SEG.
     156D 3B 01 1A29                 2914        SBF   GUF441,@B1               YES, SET OFF INDR
     1571 F2 87 05                   2915        J     GUF213                   BYPASS USING LINE NUMBER IN SEG
```

```
     1574 2C 01 1A3A 06        2916 GUF210 MVC    GUF483(GUF507),GUF639(,@XR)  MOVE SEG LINE NO TO CIT WORK
     1579 34 01 1A51           2917 GUF213 ST     GUF498,@BR                   SAVE ADDRESS OF SDF
     157D 3C FF 1593           2918        MVI    GUF216+@Q,GUF615             SET Q CODES FOR THE SEGMENT
     1581 2E 00 1593 02        2919        ALC    GUF216+@Q,GUF627(1,@XR)      * MOVE INSTRUCTIONS
     1586 0C 00 1598 1593      2920        MVC    GUF219+@Q(1),GUF216+@Q       *
     158C B6 01 02             2921        A      GUF627(,@XR),@BR             INCREMENT BR FOR MOVE
     158F B6 02 02             2922        A      GUF627(,@XR),@XR             INCREMENT XR FOR MOVE
     1592 2C 00 1106 00        2923 GUF216 MVC    GUF708,@ZERO(1,@XR)          MOVE SEGMENT TO HOLD AREA
     1597 4C 00 00 1106        2924 GUF219 MVC    @ZERO(1,@BR),GUF708          MOVE HOLD AREA TO FREE SPACE CB
     159C C0 87 154D           2925        B      GUF204                       GO CHECK FOR MORE SEG IN THIS C

     15A0 8D 01 06 1C08        2927 GUF222 CLC    GUF639(GUF507,@XR),GUF927+GUF585  LINE NO IN DELETE RANGE ?
     15A5 F2 84 12             2928        JH     GUF225                       NO, GO SET INDR OFF
     15A8 2E 00 1A3B 02        2929        ALC    GUF486,GUF627(1,@XR)         YES, INCR UNUSED SPACE IN CB
     15AD 0F 01 1D02 1A55      2930        SLC    GUF939,GUF510(GUF618)        DECR NUMBER OF LINES ON FILE
     15B3 B6 02 02             2931        A      GUF627(,@XR),@XR             POINT XR TO NEXT SDF
     15B6 C0 87 154D           2932        B      GUF204                       GO CHECK FOR MORE SEG IN THIS C

     15BA 3B 04 1A23           2934 GUF225 SBF    GUF429,GUF576                SET OFF RANGE DELETE INDR
     15BE C0 87 1566           2935        B      GUF207                       GO MOVE THIS SEGMENT
                               2936 *
                               2937 ***    NO MORE SEGMENTS TO BE SLID UP IN THIS CB,
                               2938 ***    GET SEGMENT FROM NEXT CB, IF IT EXISTS.
                               2939 *
     15C2 F2 87 0D             2940 GUF228 J      GUF231                       BYPASS DISK WAIT
     15C3                      2941        ORG    GUF228+@Q                    * THIS INSTR IS USED AS A SWITCH
     15C3 80          15C3     2942        DC     AL1(@NOP)                    * THE FIRST TIME FOLLOWING A
     15C5                      2943        ORG    GUF228+3                     * DISK OP, THE WAIT IS EXECUTED
     15C5 3C 87 15C3           2944        MVI    GUF228+@Q,@UCB               THE INSTR IS THEN SET TO JUMP
     15C9 C0 87 0025           2945        B      $DISKN                       WAIT FOR DISK
     15CD 057F        15CE     2946        DC     AL2($WAITF)                  * OP COMPLETE
     15CF F2 87 0B             2947        J      GUF234                       BYPASS CB+1 DROP TEST
     15D2 38 02 1A23           2948 GUF231 TBN    GUF429,GUF564                WAS DROP NEXT CB+1 INDR SET ?
     15D6 F2 90 04             2949        JF     GUF234                       NO, BYPASS SETTING INDR
     15D9 3C 01 1A23           2950        MVI    GUF429,GUF567                YES, RE-SET TO DROP NEXT CB
     15DD 0F 00 1A28 1A55      2951 GUF234 SLC    GUF438(1),GUF510             DECR 'PAK' COUNTER
     15E3 C0 81 1791           2952        BZ     GUF306                       ZERO! GO CLEAN UP AND GET OUT !
     15E7 0E 01 1A25 1296      2953        ALC    GUF432,GUF072(@CADDR)        INCR CB STARTING ADDR POINTER
     15ED 35 02 1A25           2954        L      GUF432,@XR                   POINTER XR TO 1ST SDF IN NECT CB
     15F1 3C 00 1A35           2955        MVI    GUF468,@ZERO                 CLEAR DISP TO NEXT SEGMENT
     15F5 38 01 1A23           2956        TBN    GUF429,GUF567                DROP 1ST SEGMENT THIS CB ?
     15F9 F2 90 13             2957        JF     GUF237                       NO, GO CHECK FOR NULL SEGMENT
     15FC B9 01 03             2958        TBF    GUF630(,@XR),GUF555          ARE THERE MORE SEG TO THIS CB ?
     15FF C0 90 15DD           2959        BF     GUF234                       YES, GO GET THEM
     1603 3B 01 1A23           2960        SBF    GUF429,GUF567                TURN OFF DROP SEGMENT INDR
     1607 2E 00 1A35 02        2961        ALC    GUF468,GUF627(1,@XR)         INCR DISP OF NEXT SEGMENT
     160C B6 02 02             2962        A      GUF627(,@XR),@XR             INCR XR FOR NEXT SDF
     160F BD 80 01             2963 GUF237 CLI    GUF624(,@XR),GUF546          IS THIS A NULL SEGMENT ?
     1612 C0 81 15D2           2964        BE     GUF231                       YES, GO TO NEXT CB
     1616 3D FB 1A35           2965        CLI    GUF468,GUF552                DID LAST SEGMENT FINISH CB ?
     161A C0 02 15D2           2966        BNL    GUF231                       YES, GO TO NEXT CB
     161E 38 04 1A23           2967        TBN    GUF429,GUF576                IS A RANGE BEING DELETED ?
     1622 F2 90 24             2968        JF     GUF246                       NO, GO CHK FOR SPACE IN PRIOR CB
     1625 B8 02 03             2969        TBN    GUF630(,@XR),GUF549          TEST FOR 1ST OR ONLY SEGMENT
     1628 F2 10 0E             2970        JT     GUF240                       NO, BYPASS RANGE CHECK
     162B 8D 01 06 1C08        2971        CLC    GUF639(GUF507,@XR),GUF930    IS SEG IN RANGE BEING DELETED ?
```

```
     1630 F2 84 12            2972        JH    GUF243              NO, GO SET OFF INDR
     1633 0F 01 1D02 1A55     2973        SLC   GUF939(GUF618),GUF510  YES, DECR FILE LINE COUNT
     1639 2E 00 1A35 02       2974 GUF240 ALC   GUF468,GUF627(1,@XR)   INCR DISP TO NEXT SEGMENT
     163E B6 02 02            2975        A     GUF627(,@XR),@XR    INCR XR FOR NEXT SEGMENT
     1641 C0 87 160F          2976        B     GUF237              GO CHECK NEXT SEGMENT
     1645 3C 00 1A23          2977 GUF243 MVI   GUF429,@ZERO        TURN OFF DELETE INDR
     1649 3D 07 1A3B          2978 GUF246 CLI   GUF486,GUF591       AT LEAST MIN SPACE LEFT IN CB ?
     164D F2 04 EF            2979        JNH   GUF294              NO, SET TO FILL NEXT CB
     1650 8D 00 02 1A3B       2980        CLC   GUF627(1,@XR),GUF486  WILL SEGMENT FIT IN PRIOR CB ?
     1655 F2 04 1A            2981        JNH   GUF249              YES, GO PUT IT IN
     1658 B8 02 03            2982        TBN   GUF630(,@XR),GUF549 NO, IS SEGMENT 1ST OR ONLY SEG ?
     165B F2 90 A1            2983        JF    GUF276              YES, GO BREAK IT UP
     165E 2C 00 1A2A 02       2984        MVC   GUF444,GUF627(1,@XR)   CHECK IF SEG WILL FIT
     1663 0F 00 1A2A 1A76     2985        SLC   GUF444(1),GUF675    * WHWN SPARE SDF IS
     1669 0D 00 1A3B 1A2A     2986        CLC   GUF486(1),GUF444    * DROPPED ?
     166F F2 82 99            2987        JL    GUF279              NO, GO BREAK THE SEGMENT
     1672 2E 00 1A35 02       2988 GUF249 ALC   GUF468,GUF627(1,@XR)   INCR DISP FOR NEXT SEGMENT
     1677 B8 02 03            2989        TBN   GUF630(,@XR),GUF549 IS SEGMENT 1ST OR ONLY ?
     167A F2 90 39            2990        JF    GUF258              YES, GO UPDATE LINE NUMBER
     167D 3D FF 1A3B          2991        CLI   GUF486,GUF414       IS THIS SEG 1ST IN NEW CB ?
     1681 F2 81 3A            2992        JE    GUF261              YES, GO USE SAVED LINE NUMBER
     1684 3C 04 16E0          2993        MVI   GUF270+@D1,GUF540   SET INST TO ADD SDF LENGTH TO XR
     1688 8F 00 02 1A76       2994        SLC   GUF627(1,@XR),GUF675   DECR XR BY SDF LENGTH
     168D 35 01 1A51          2995        L     GUF498,@BR          SAVE PRIOR SEGMENT LENGTH
     1691 1C 00 16B2 02       2996        MVC   GUF255+@D1,GUF627(1,@BR)  *
     1696 6E 00 02 02         2997        ALC   GUF627(1,@BR),GUF627(,@XR)  INCR PRIOR SDF FOR NEW SEG LNG
     169A B8 01 03            2998        TBN   GUF630(,@BR),GUF555 TEST AND RESET BYTE 2 OF THE
     169D F2 10 10            2999        JT    GUF255              * SDF'S TO REFLECT THE STATUS
     16A0 78 02 03            3000        TBN   GUF630(,@BR),GUF549 * OF THE NEW COMBINED SEGMENT
     16A3 F2 90 07            3001        JF    GUF252              *
     16A6 6C 00 03 03         3002        MVC   GUF630(1,@BR),GUF630(,@XR)
     16AA F2 87 03            3003        J     GUF255              *
     16AD 7C 00 03            3004 GUF252 MVI   GUF630(,@BR),@ZERO  *
     16B0 D2 01 00            3005 GUF255 LA    @ZERO(,@BR),@BR     *
     16B3 F2 87 14            3006        J     GUF267              *
     16B6 2C 01 1A3A 06       3007 GUF258 MVC   GUF483,GUF639(GUF507,@XR) UPDATE HIGH LINE NUMBER IN CB
     16BB F2 87 04            3008        J     GUF264              BYPASS USING SAVED LINE NUMBER
     16BE 3C 00 1A29          3009 GUF261 MVI   GUF441,@ZERO        TURN OFF SEGMENT BREAK INDR
     16C2 34 01 1A51          3010 GUF264 ST    GUF498,@BR          SAVE ADDR OF LAST 'PAK'ED SEGMENT
     16C6 3C 00 16E0          3011        MVI   GUF270+@D1,@ZERO    NOP INST TO ADD SDF LENGTH TO XR
     16CA 2F 00 1A3B 02       3012 GUF267 SLC   GUF486(1,@XR)       DECR SPACE LEFT IN CB
     16CF B6 01 02            3013        A     GUF627(,@XR),@BR    INCR BR TO POINT TO LAST BYTE 0 SEG
     16D2 3C FF 16E2          3014        MVI   GUF273+@Q,GUF615    CALCULATE LENGTH OF SEGMENT
     16D6 2E 00 16E2 02       3015        ALC   GUF273+@Q,GUF627(1,@XR) * TO BE MOVED
     16DB B6 02 02            3016        A     GUF627(,@XR),@XR    INCR XR TO LAST BYTE OF SEGMENT
     16DE E2 02 00            3017 GUF270 LA    @ZERO(,@XR),@XR     ADD SDF LENGTH TO XR IF DROPPING SDF
     16E1 6C 00 00 00         3018 GUF273 MVC   @ZERO(1,@BR),@ZERO(,@XR)  'PAK' SEGMENT TO PRIOR CB
     16E5 38 01 1A29          3019        TBN   GUF441,GUF582       WAS LINE BROKEN ?
     16E9 C0 90 160F          3020        BF    GUF237              NO, GO CHECK NEXT SEGMENT
     16ED 36 02 1A78          3021        A     GUF678,@XR          DECR XR BY SDF LENGTH
     16F1 8C 03 04 1A2D       3022        MVC   GUF633(GUF540,@XR),GUF453 MOVE IN NEW SDF
     16F6 0F 00 1A35 1A76     3023        SLC   GUF468(1),GUF675    DECR DISP TO NEXT SEG BY SEG LENGTH
     16FC F2 87 40            3024        J     GUF294              GO SET TO FILL NEXT CB
     16FF BD 00 03            3025 GUF276 CLI   GUF630(,@XR),@ZERO  THE SEGMENT MUST BE BROKEN OVER
     1702 F2 01 0C            3026        JNE   GUF282              * 2 CB'S.  SET UP 2 SDF'S, ONE SDf
     1705 BC 01 03            3027        MVI   GUF630(,@XR),GUF555 * WILL BE IN A WORK AREA THAT
```

```
     1708 F2 87 10          3028         J     GUF288              * WILL BE MOVED INTO THE 2ND
     170B BD 02 03          3029 GUF279 CLI   GUF630(,@XR),GUF549  * OF THE SEGMENT HAVE BEEN MOVED
     170E F2 81 07          3030         JE    GUF285              * TO THE 1ST CB.
     1711 3C 03 1A2C        3031 GUF282 MVI   GUF450,GUF621        *
     1715 F2 87 07          3032         J     GUF291              *
     1718 BC 03 03          3033 GUF285 MVI   GUF630(,@XR),GUF621  *
     171B 3C 02 1A2C        3034 GUF288 MVI   GUF450,GUF549        *
     171F 8E 00 02 1A76     3035 GUF291 ALC   GUF627(1,@XR),GUF675 *
     1724 2C 01 1A2B 02     3036         MVC   GUF447,GUF627(GUF570,@XR) *
     1729 8C 00 02 1A3B     3037         MVC   GUF627(1,@XR),GUF486 *
     172E 2F 00 1A2B 02     3038         SLC   GUF447,GUF627(1,@XR) *
     1733 3A 01 1A29        3039         SBN   GUF441,GUF582        SET ON THE SEGMENT BREAK INDR
     1737 3C 00 1A2D        3040         MVI   GUF453,@ZERO         CLEAR LAST BYTE OF TEMP SDF
     173B C0 87 1672        3041         B     GUF249               GO MOVE 1ST SEGMENT TO 1ST CB
                            3042 *
                            3043 ***    A CORE BLOCK HAS JUST BEEN COMPLETELY 'PAK'ED - SAVE TEMP FIT
                            3044 ***    ENTRY FOR IT - SET 'PAK' TO FILL INTO THE NEXT NON-NULL CB.
                            3045 *
     173F 0C 03 0000 1A3B   3046 GUF294 MVC   *-*,GUF486(GUF540)    SAVE TEMP FIT ENTRY
     1745 0E 01 1742 1A76   3047         ALC   GUF474(@CADDR),GUF675 INCR FOR NEXT ENTRY TO BE SAVED
     174B 0E 00 1A53 1A55   3048         ALC   GUF504(1),GUF510      INCR ADDRESS OF CB'S USED
     1751 7C 80 01          3049         MVI   GUF624(,@BR),GUF546   MOVE A NULL SDF OF THE CB
     1754 3D 03 1A3B        3050         CLI   GUF486,GUF585         * JUST 'PAK'ED
     1758 F2 82 05          3051         JL    GUF297               *
     175B 4C 03 04 1A68     3052         MVC   GUF633(GUF540,@BR),GUF645 *
     1760 0E 01 1A27 1296   3053 GUF297 ALC   GUF435(@CADDR),GUF072 INCR FILL POINTER
     1766 35 01 1A27        3054         L     GUF435,@BR           PUT IN BR
     176A 0D 01 1A27 1A25   3055         CLC   GUF435(@CADDR),GUF432 FILL TO & FROM SAME CB ?
     1770 F2 81 08          3056         JE    GUF300               YES, GO CALC FREE SPACE
     1773 3C FF 1A3B        3057         MVI   GUF486,GUF414        NO, SET FREE SPACE AT MAX.
     1777 C0 87 1649        3058         B     GUF246               GO FILL FROM DIFFERENT CB
     177B 0C 01 1786 1742   3059 GUF300 MVC   GUF303+@OP2,GUF474(@CADDR) PICK UP NUMBER OF BYTES OF
     1781 0C 00 1A3B 0000   3060 GUF303 MVC   GUF486(1),*-*         * FREE SPACE AT END OF NEXT CB
     1787 0E 00 1A3B 1A35   3061         ALC   GUF486(1),GUF468     * ADD FREE SPACE AT FRONT
     178D C0 87 154D        3062         B     GUF204               GO SLIDE UP CONTENTS OF CB
                            3063 *
                            3064 ***    CORE BLOCKS ARE COMPLETELY 'PAK'ED UP - UPDATE THE FIT
                            3065 ***    AND WRITE THE CONTROL BLOCKS TO DISK.
                            3066 *
     1791 35 02 1D09        3067 GUF306 L     GUF942,@XR           LOAD FIT BASE TO XR
     1795 3C 80 15C3        3068         MVI   GUF228+@Q,@NOP       SET FOR WAIT FOLLOWING READ
     1799 3D 00 1A23        3069         CLI   GUF429,@ZERO         SHOULD MORE SEGMENTS BE DELETED ?
     179D F2 81 4A          3070         JE    GUF318               NO, BYPASS READING MORE DB'S
     17A0 3C 00 1A2A        3071 GUF309 MVI   GUF444,@ZERO         INCR BY NR OF DRS ALREADY READ
     17A4 0C 00 1A2B 1A52   3072         MVC   GUF447(1),GUF501     *
     17AA 0E 01 1A2B 1A2B   3073         ALC   GUF447(@CADDR),GUF447 *
     17B0 0E 01 1A2B 1A2B   3074         ALC   GUF447(@CADDR),GUF447 *
     17B6 36 02 1A2B        3075         A     GUF447,@XR           *
     17BA 0D 01 1742 1A4D   3076         CLC   GUF474(@CADDR),GUF492 IS CB2 BEING USED ?
     17C0 F2 01 11          3077         JNE   GUF312               YES, GO FILL CB3 AND CB4
     17C3 3C 23 1464        3078         MVI   GUF195,GUF405        NO, READ 3 DB INTO CB2 THRU
     17C7 0C 01 1A25 1A64   3079         MVC   GUF432(@CADDR),GUF642 RESET 'PAK' START POINTER
     17CD 3C 04 1A28        3080         MVI   GUF438,GUF540        REST NUMBER TO BE 'PAK'ED
     17D1 F2 87 0E          3081         J     GUF315               GO TO READ DB BLOCK
     17D4 3C 32 1464        3082 GUF312 MVI   GUF195,GUF543        READ 2 DB'S INTO CB3 AND CB4
     17D8 0C 01 1A25 1A6A   3083         MVC   GUF432(@CADDR),GUF648 RESET 'PAK' START POINTER
```

```
    17DE 3C 03 1A28          3084       MVI   GUF438,GUF585          REST NUMBER TO BE 'PAK'ED
    17E2 C0 87 1389          3085 GUF315 B    GUF138                 GO TO READ DB'S
    17E6 C0 87 15C2          3086       B     GUF228                 GO 'PAK' DB'S JUST READ
                             3087 *
                             3088 ***     ALL 'PAK'ING IS DONE UNLESS 1ST OR 2ND DISK BLOCK HAS BECOME UNUSED
                             3089 ***     UPDATE CORE INDEX TABLE AND CHECK FOR ABOVE CONDITION
                             3090 *
    17EA 4C 03 04 1A68       3091 GUF318 MVC  GUF633(GUF540,@BR),GUF645  PUT A NULL SDF IN LAST USED CB
    17EF 0C 01 17F8 1742     3092       MVC   GUF321+@OP1(@CADDR),GUF474 MOVE CIT WORK AREA TO CIT
    17F5 0C 03 0000 1A3B     3093 GUF321 MVC  *-*,GUF486(GUF540)          *
    17FB 3D 00 1A53          3094       CLI   GUF504,@ZERO           CHECK IF 1ST OR 2ND DB HAS
    17FF F2 84 18            3095       JH    GUF324                 * BECOME UNUSED AND IS TO BE
    1802 3D FF 1A3F          3096       CLI   GUF489-GUF408,GUF414   * DROPPED, IF YES AND THE LINE NR
    1806 C0 81 17A0          3097       BE    GUF309                 * IN DB1 OR DB2 IS NOT THE END
    180A BD 01 01            3098       CLI   GUF651(,@XR),@B1       * OF FILE SEGMENT: OR IF THERE
    180D F2 84 0A            3099       JH    GUF324                 * IS NO DB TO WRITE OUT, THEN
    1810 0D 01 1A3E 1A1F     3100       CLC   GUF489-13(@CADDR),GUF417 * READ IN AND 'PAK' MORE DB'S
    1816 C0 01 17A0          3101       BNE   GUF309                 * ELSE CONTINUE...
                             3102 *
                             3103 ***     ALL 'PAK'ING COMPLETE; UPDATE FIT AND WRITE CB'S TO DISK
                             3104 *
    181A 0E 00 1A53 1A55     3105 GUF324 ALC  GUF504(1),GUF510       INCR BIT COUNTER
    1820 0D 00 1A53 1A52     3106       CLC   GUF504(1),GUF501       NUMBER OUT ?
    1826 C0 82 1939          3107       BL    GUF375                 LT NR IN, GO SLIDE DOWN
    182A F2 81 61            3108       JE    GUF342                 EQ NR IN, GO SET FIT
    182D 3D BD 1D00          3109       CLI   GUF936,GUF522          GT NR IN, WILL IT FIT IN FILE ?
    1831 F2 82 23            3110       JL    GUF336                 YES, GO INCR COUNT AND PUT IN
    1834 3C 8B 03CD          3111       MVI   $CAERR,@@E530          NO, SET PHYS FULL ERROR MSG NR
    1838 F2 87 06            3112 GUF327 JC   GUF330,@UCB            DECR COUNT IF NEW LINE SWITCH SET
    183B 0F 01 1D02 1A55     3113       SLC   GUF939(2),GUF510       DECR FILE LINE COUNT
    1841 0C 01 03CF 1C05     3114 GUF330 MVC  $INLNO(@SBLNL),GUF927  SET LINE NUMBER FOR ERROR MSG
    1847 0C 01 12D2 1856     3115       MVC   GUF096+@OP1(@CADDR),GUF333  SET WAIT LOOP RETURN TO LOOP
    184D 3C 87 12E0          3116       MVI   GUF099+@Q,@UCB         SET BR TO ERRPGM
    1851 C0 87 12AC          3117       B     GUF090                 GO WAIT FOR CARRIER RETURN
    1855 12AC           1856 3118 GUF333 DC   AL2(GUF090)            CADDR FOR CARRIER ONLY WAIT LOOP
    1857 0E 00 1D00 1A55     3119 GUF336 ALC  GUF936(1),GUF510       INCR DB'S USED COUNT
    185D 3C 00 1A2A          3120       MVI   GUF444,@ZERO           CHECK IF NEXT
    1861 0C 00 1A2B 1A52     3121       MVC   GUF447(1),GUF501       * LOGICAL DB
    1867 0E 00 1A2B 1A2B     3122       ALC   GUF447(1),GUF447       * IS NULL ?
    186D 0E 00 1A2B 1A2B     3123       ALC   GUF447(1),GUF447       *
    1873 0C 00 1934 1A2B     3124       MVC   GUF372+@D1(1),GUF447   SAVE FOR NULL ENTRY DISP
    1879 0E 01 1A2B 1D09     3125       ALC   GUF447(@CADDR),GUF942  *
    187F 0D 01 1A2B 1D0B     3126       CLC   GUF447(@CADDR),GUF945  *
    1885 F2 01 54            3127       JNE   GUF357                 * NO, GO SLIDE UP
    1888 0E 01 1D0B 1A76     3128 GUF339 ALC  GUF945(@CADDR),GUF675  * YES, INCR NULL DB POINTER
    188E 0C 00 1A28 1A53     3129 GUF342 MVC  GUF438(1),GUF504       SET NUMBER TO BE WRITTEN OUT
    1894 3C 04 18A0          3130       MVI   GUF345+@D1,GUF540      SET CIT TO FIT MOVE
    1898 0C 01 18A2 1A4D     3131       MVC   GUF345+@DOP2(@CADDR),GUF492
    189E 8C 02 04 0000       3132 GUF345 MVC  GUF657(GUF585,@XR),*-* MOVE CIT ENTRY TO FIT ENTRY
    18A3 0F 00 1A53 1A55     3133       SLC   GUF504(1),GUF510       DECR NUMBER OUT COUNTER
    18A9 F2 81 10            3134       JZ    GUF348                 EQUALS ZERO, GO SET BASE
    18AC 0E 00 18A0 1A76     3135       ALC   GUF345+@D1(1),GUF675   NOT EQUAL ZERO, INCR CIT TO
    18B2 0E 01 18A2 1A76     3136       ALC   GUF345+@DOP2(@CADDR),GUF675  * FIT MOVE POINTERS
    18B8 C0 87 189E          3137       B     GUF345                 CONTINUE MOVING CIT
    18BC C2 01 11F3          3138 GUF348 LA   GUF036,@BR             SET BASE REGISTER AND GO WRITE
    18C0 C0 87 1217          3139       B     GUF045                 * CB'S TO DISK
```

```
      DL4ICS - 4 SURFACE DISK IOCS ROUTINE


ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT          VER 15, MOD 00  29/02/16  PAGE  71

    18C4 0D 01 1D02 1A55      3140          CLC    GUF939(2),GUF510           IS FILE EMPTY ?
    18CA F2 81 07             3141          JE     GUF351                     YES, SET FILE INDR TO ZERO
    18CD 3B 04 03D4           3142          SBF    $INDR1,$WSIND              SET FILE INDR ON
    18D1 F2 87 04             3143          J      GUF354                     GO GET OUT
    18D4 3A 04 03D4           3144 GUF351 SBN    $INDR1,$WSIND              SET FILE INDR OFF
    18D8 C0 87 0000           3145 GUF354 B      *-*                        RETURN TO CALLING ROUTINE
                              3146 *
                              3147 ***    SLIDE FIT ENTRIES UP TO INSERT NEW DB.
                              3148 *
    18DC 35 01 1D0B           3149 GUF357 L      GUF945,@BR                 SAVE THE DISK DISPLACEMENT FROM
    18E0 34 01 1A2D           3150          ST     GUF453,@BR                 * THE 1ST NULL ENTRY.  CALC THE
    18E4 0F 01 1A2D 1A2B      3151          SLC    GUF453(@CADDR),GUF447      * ADDRESSES AND SLIDE THE VALID
    18EA 1C 00 1933 01        3152          MVC    GUF372+@Q,GUF651(1,@BR)    * FIT ENTRIES, FOLLOWING THE
    18EF 0D 01 1A2D 1296      3153 GUF360 CLC    GUF453(@CADDR),GUF072      * LAST DB READ, UP 1 SLOT TO
    18F5 F2 84 0F             3154          JH     GUF363                     * MAKE ROOM FOR THE NEW LOGICAL
    18F8 0F 01 1A2D 1A55      3155          SLC    GUF453(@CADDR),GUF510      * DB ENTRY.  SET THE SAVED DISK
    18FE 0C 00 1916 1A2D      3156          MVC    GUF366+@Q(1),GUF453        * DISPLACEMENT FROM THE NULL
    1904 F2 87 0E             3157          J      GUF366                     * ENTRY AS THE DISK DISP FOR
    1907 0F 00 1A2C 1A55      3158 GUF363 SLC    GUF450(1),GUF510           * THE NEW DB
    190D 3C 01 1A20           3159          MVI    GUF420,@B1                 *
    1911 3C FF 1916           3160          MVI    GUF366+@Q,GUF414           *
    1915 5C 00 04 00          3161 GUF366 MVC    GUF657(1,@BR),@ZERO(,@BR)  *
    1919 3D 01 1A20           3162          CLI    GUF420,@B1                 *
    191D F2 01 0C             3163          JNE    GUF369                     *
    1920 36 01 1A22           3164          A      GUF423,@BR                 *
    1924 3C 00 1A20           3165          MVI    GUF420,@ZERO               *
    1928 C0 87 18EF           3166          B      GUF360                     *
    192C 0E 00 1934 1A55      3167 GUF369 ALC    GUF372+@D1(1),GUF510       *
    1932 BC 00 01             3168 GUF372 MVI    GUF651(,@XR),@ZERO         *
    1935 C0 87 1888           3169          B      GUF339                     GO INCR NULL ENTRY POINTER
                              3170 *
                              3171 ***    SLIDE FIT ENTRIES DOWN TO SQUEEZE OUT DB GONE UNUSED
                              3172 *
    1939 C2 01 1A1F           3173 GUF375 LA     GUF417,@BR                 LOAD BASE REGISTER
              1A1F            3174          USING GUF417,@BR                 *
    193D 5F 03 0E 0E          3175          SLC    GUF453(4,@BR),GUF453(,@BR)  CLEAR WORKAREAS
    1941 5C 00 0E 33          3176          MVC    GUF453(1,@BR),GUF501(,@BR)  CALC # OF DB'S
    1945 5F 00 0E 34          3177          SLC    GUF453(1,@BR),GUF504(,@BR)  * GOING NULL
    1949 1F 00 1D00 0E        3178          SLC    GUF936(1),GUF453(,@BR)      SUBT FROM DB'S USED COUNT
    194E 5C 00 11 0E          3179          MVC    GUF459(1,@BR),GUF453(,@BR)  SAVE # GOING NULL
    1952 5C 00 0C 34          3180          MVC    GUF447(1,@BR),GUF504(,@BR)  SET # DB'S BACK OUT TO WORKAREA
    1956 5E 03 0E 0E          3181          ALC    GUF453(4,@BR),GUF453(,@BR)  CALC # GOING NULL * 4 AND
    195A 5E 03 0E 0E          3182          ALC    GUF453(4,@BR),GUF453(,@BR)  * # BACK OUT * 4
    195E 76 02 0C             3183          A      GUF447(,@BR),@XR           INCR @XR TO 1ST ENTRY GOING NULL
    1961 74 02 10             3184          ST     GUF456(,@BR),@XR           CALC CADDR OF LAST READ ENTRY
    1964 5E 01 10 0E          3185          ALC    GUF456(2,@BR),GUF453(,@BR)  * PLUS 1
    1968 4D 01 10 1D0B        3186          CLC    GUF456(@CADDR,@BR),GUF945   IS IT ALSO THE 1ST NULL ?
    196D F2 81 A0             3187          JE     GUF399                     YES, GO RESET 1ST NULL ADDR
    1970 5C 00 14 11          3188          MVC    GUF465(1,@BR),GUF459(,@BR)  SAVE # GOING NULL
    1974 74 02 0C             3189          ST     GUF447(,@BR),@XR           SAVE CADDR FOR THE REST OF USED ENT
    1977 4C 01 13 1D0B        3190          MVC    GUF462(@CADDR,@BR),GUF945   CALC LENGTH OF REST OF USED
    197C 5F 01 13 10          3191          SLC    GUF462(@CADDR,@BR),GUF456(,@BR) * ENTRIES
              0001            3192          DROP   @BR
    1980 C2 01 1007           3193          LA     GUF708-255,@BR             POINT TO DISP SAVE AREA
    1984 6C 00 00 01          3194 GUF378 MVC    @ZERO(1,@BR),GUF651(,@XR)  SAVE DISP OF ENTRY GOING NULL
    1988 0F 00 1A30 1A55      3195          SLC    GUF459(1),GUF510           DECR COUNT OF # GOING NULL
```

```
     198E F2 81 0A              3196          JZ      GUF381                  ALL SAVED, GO TO MOVE USED ENTS
     1991 D2 01 01              3197          LA      @B1(,@BR),@BR           INCR @BR TO NEXT SAVE ADDR
     1994 E2 02 04              3198          LA      GUF540(,@XR),@XR        INCR @XR TO NEXT ENTRY
     1997 C0 87 1984            3199          B       GUF378                  GO SAVE DISPLACEMENT
                                3200 *
                                3201 ***     DISP OF ENTRIES GOING NULL HAVE ALL BEEN SAVED.
                                3202 ***     NOW START MOVING REST OF ENTRIES USED DOWN IN FIT.
                                3203 *
     199B 35 02 1A2B            3204 GUF381 L        GUF447,@XR              SET POINTER TO START OF NEW LOC
     199F 35 01 1A2F            3205        L        GUF456,@BR              SET POINTER TO START OF OLD LOC
     19A3 3C 43 19C5            3206        MVI      GUF387+@Q,GUF636-1      INIT MOVE LENGTH TO MAX
     19A7 3C 44 19C8            3207        MVI      GUF387+@OPD2,GUF636     INIT DISP FOR MAX MOVE
     19AB 0D 01 1A32 1A4F       3208 GUF384 CLC      GUF462,GUF495(2)        IS MAX MOVE REQUIRED ?
     19B1 F2 84 10              3209        JH       GUF387                  YES, GO DO IT
     19B4 3C FF 19C5            3210        MVI      GUF387+@Q,GUF615        NO, CALC EXECT LENGTH REQUIRED
     19B8 0E 00 19C5 1A32       3211        ALC      GUF387+@Q(1),GUF462     *
     19BE 0C 00 19C8 1A32       3212        MVC      GUF387+@OPD2(1),GUF462  SET EXACT DISP REQUIRED
     19C4 1C 00 1106 00         3213 GUF387 MVC      GUF708(*-*),*-*(,@BR)   MOVE USED ENTRIES TO WORKAREA
     19C9 0C 00 19D6 19C5       3214        MVC      GUF390+@Q,GUF387+@Q(1)  SET THE SAME LENGTH AND DISP
     19CF 0C 00 19D7 19C8       3215        MVC      GUF390+@D1,GUF387+@OPD2(1)  * FOR THE MOVE TO THE FIT
     19D5 8C 00 00 1106         3216 GUF390 MVC      *-*(*-*,@XR),GUF708     RESET THE USED ENTRIES IN FIT
     19DA 3C 00 19D6            3217        MVI      GUF390+@Q,@ZERO         SUBTRACT THE LENGTH MOVED FROM
     19DE 0F 01 1A32 19D7       3218        SLC      GUF462(2),GUF390+@D1    * THE MOVE LENGTH REQUIRED
     19E4 F2 81 0A              3219        JZ       GUF393                  IF REQ'D LNG SATISFIED, GET OUT
     19E7 D2 01 44              3220        LA       GUF636(,@BR),@BR        ELSE, INCR OLD LOC POINTER
     19EA E2 02 44              3221        LA       GUF636(,@XR),@XR        INCR NEW LOC POINTER
     19ED C0 87 19AB            3222        B        GUF384                  GO CHECK FOR MAX MOVE
                                3223 *
                                3224 ***     USED ENTRIES ARE NOW JOINED.
                                3225 ***     START PUTTING SAVED DISPLACEMENTS OF NULL ENTRIES BACK IN FIT
                                3226 *
     19F1 C2 01 1007            3227 GUF393 LA       GUF708-255,@BR          SET POINTER TO SAVED DISP
     19F5 36 02 19D7            3228        A        GUF390+@D1,@XR          SET @XR TO NEW FIRST NULL
     19F9 9C 00 01 00           3229 GUF396 MVC      GUF651(1,@XR),@ZERO(,@BR) RESTORE A SAVED DISP
     19FD 0F 00 1A33 1A55       3230        SLC      GUF465(1),GUF510        DECR COUNT OF # SAVED
     1A03 F2 81 0A              3231        JZ       GUF399                  ALL RESTORED, GO LET OUT
     1A06 E2 02 04              3232        LA       GUF540(,@XR),@XR        NO, INCR TO NEXT FIT ENTRY
     1A09 D2 01 01              3233        LA       @B1(,@BR),@BR           INCR TO NEXT SAVED DISP
     1A0C C0 87 19F9            3234        B        GUF396                  GO RESTORE SAVED DISP
                                3235 *
                                3236 ***     SAVED DISPLACEMENTS RESTORED -  RESET POINTERS.
                                3237 *
     1A10 0F 01 1D0B 1A2D       3238 GUF399 SLC      GUF945(@CADDR),GUF453   RESET CADDR OF FIRST NULL
     1A16 35 02 1D09            3239        L        GUF942,@XR              POINT @XR TO 1ST ENTRY READ
     1A1A C0 87 188E            3240        B        GUF342                  GO MOVE CIT TO FIT

                           000F 3242 GUF402 EQU      X'0F'                   COMMAND KEY MASK
                           0023 3243 GUF405 EQU      X'23'                   READ PARM - 3 DBS TO CB2 - CB4
                           000C 3244 GUF408 EQU      12                      LENGTH OF 3 FIT ENTRIES
                           0008 3245 GUF411 EQU      8                       LENGTH OF 2 FIT ENTRIES
                           00FF 3246 GUF414 EQU      X'FF'                   MAX TEXT SPACE IN A DB
     1A1E 2710            1A1F 3247 GUF417 DC       XL2'2710'               END OF FILE SEG LINE NUMBER
     1A20                 1A20 3248 GUF420 DS       CL1                     MOVE MORE THAN 256 BYTE INDR
     1A20                      3249        ORG      *-1                     INITIALIZE TO OFF POSITION
     1A20 00              1A20 3250        DC       XL1'00'                 *
     1A21 FF00            1A22 3251 GUF423 DC       XL2'FF00'               NEGATIVE 256
```

DL4ICS - 4 SURFACE DISK IOCS ROUTINE

```
                            0010 3252 GUF426 EQU   16                    LENGTH OF CORE INDEX TABLE
      1A23                  1A23 3253 GUF429 DS    CL1
      1A25                  1A25 3254 GUF432 DS    CL2                   BYTE 1 OF 1ST CB TO BE 'PAK'ED
      1A26                  1A27 3255 GUF435 DS    CL2                   CB TO BE FILLED POINTER
      1A28                  1A28 3256 GUF438 DS    CL1                   NUMBER OF CB'S TO BE 'PAK'ED
      1A23                       3257        ORG   *-6                   * INITIALIZE ABOVE INDR TO ZERO
      1A23 000000000000     1A28 3258        DC    XL6'00'               *
      1A29                  1A29 3259 GUF441 DS    CL1                   SEGMENT BROKEN OVER CDS INDR
      1A29                       3260        ORG   *-1                   INITIALIZE TO ZERO
      1A29 00               1A29 3261        DC    XL1'00'               *
      1A2A                  1A2A 3262 GUF444 DS    CL1                   'PAK' WORK AREA
      1A2B                  1A2B 3263 GUF447 DS    CL1                   'PAK' WORK AREA
      1A2C                  1A2C 3264 GUF450 DS    CL1                   'PAK' WORK AREA
      1A2D                  1A2D 3265 GUF453 DS    CL1                   'PAK' WORK AREA
      1A2A                       3266        ORG   *-4                   * INITTIALIZE TO ZERO
      1A2A 00000000         1A2D 3267        DC    XL4'00'               *
      1A2E                  1A2F 3268 GUF456 DS    CL2                   WORK AREA
      1A30                  1A30 3269 GUF459 DS    CL1                   WORK AREA
      1A31                  1A32 3270 GUF462 DS    CL2                   WORK AREA
      1A33                  1A33 3271 GUF465 DS    CL1                   WORK AREA
      1A34                  1A35 3272 GUF468 DS    CL2                   DISP TO 1ST BYTE OF FREE SPACE
      1A34                       3273        ORG   *-2                   * 1ST BYTE OF POINTER WILL
      1A34 0000             1A35 3274        DC    XL2'0000'             * ALLWAYS BE ZERO
      1A36                  1A37 3275 GUF471 DS    CL2                   NR OF BYTES OF FREE SPACE IN
      1A36                       3276        ORG   *-2                   * THE MIDDLE OF A CB, 1ST BYTE
      1A36 0000             1A37 3277        DC    XL2'0000'             * WILL ALLWAYS BE ZERO
                            1742 3278 GUF474 EQU   GUF294+@OP1           ADDRESS IN CIT FOR CURRENT CB
      1A38                  1A38 3279 GUF477 DS    CL1                   BYTE 1 OF CIT WORK AREA
      1A39                  1A39 3280 GUF480 DS    CL1                   * BYTE 2
      1A3A                  1A3A 3281 GUF483 DS    CL1                   * BYTE 3
      1A3B                  1A3B 3282 GUF486 DS    CL1                   * BYTE 4
      1A3C                  1A4B 3283 GUF489 DS    CL16                  CORE INDEX TABLE
      1A4C 1A3F             1A4D 3284 GUF492 DC    AL2(GUF489-12)        FIRST ENTRY IN CIT
      1A4E 0044             1A4F 3285 GUF495 DC    IL2'68'               LENGTH OF FIT TRANSFER AREA
      1A50                  1A51 3286 GUF498 DS    CL2                   LAST SDF ADDRESS
      1A52                  1A52 3287 GUF501 DS    CL1                   NUMBER OF DB'S READ
      1A52                       3288        ORG   *-1                   * INITIALIZE TO ZERO
      1A52 00               1A52 3289        DC    XL1'00'               *
      1A53                  1A53 3290 GUF504 DS    CL1                   NUMBER OF CBS USED BY 'PAK'
      1A53                       3291        ORG   *-1                   * INITIALIZE TO ZERO
      1A53 00               1A53 3292        DC    XL1'00'               *
                            0002 3293 GUF507 EQU   2                     LINE NUMBER FIELD LENGTH
      1A54 0001             1A55 3294 GUF510 DC    XL2'01'               CONSTANT VALUE = 1
      1A56 00FF             1A57 3295 GUF513 DC    XL2'FF'               CONSTANT VALUE = 255
      1A58 0200             1A59 3296 GUF516 DC    XL2'0200'             SDF CODE - LAST OF MULTI SEG
      1A5A 03DE             1A5B 3297 GUF519 DC    IL2'0990'             MAX LOGICAL WORK FILE SIZE
                            00BD 3298 GUF522 EQU   #@@#WD                MAX NR OF DB'S IN WORK FILE
      1A5C                  1A5D 3299 GUF525 DS    CL2                   2 BYTE WORK AREAS - USED WHEN
      1A5E                  1A5F 3300 GUF528 DS    CL2                   * INSERTING A NEW OR
      1A60                  1A61 3301 GUF531 DS    CL2                   * REPLACEMENT LINE
                            0011 3302 GUF534 EQU   X'11'                 READ PARAMETER, 1 DB INTO CB1
                            000F 3303 GUF537 EQU   15                    WORK AREA SIZE
                            0004 3304 GUF540 EQU   4                     FIT ENTRY AND SDF LENGTH
                            0032 3305 GUF543 EQU   X'32'                 READ PARM, 2 DB INTO CB3 & 4
                            0080 3306 GUF546 EQU   X'80'                 TEST BYTE FOR NULL SEGMENT
                            0002 3307 GUF549 EQU   X'02'                 TEST BYTE FOR NON-1ST SEGMENT
```

```
                          00FB 3308 GUF552 EQU   251                         DB FULL
                          0001 3309 GUF555 EQU   1                           TEST BYTE FOR MULTIPLE SEGMENTS
                          0013 3310 GUF558 EQU   X'13'                       READ PARM, 3 DB'S INTO CB1 - 3
     1A62                 1A62 3311 GUF561 DS    CL1                         NO INSERTION TO CB2 INDR
     1A62                      3312        ORG   *-1                         * INITIALIZE TO ZERO
     1A62 00              1A62 3313        DC    XL1'00'                     *
                          0002 3314 GUF564 EQU   2                           PDRP CODE-DROP SEGMENT IN N+1 CB
                          0001 3315 GUF567 EQU   1                           PDRP CODE-DROP SEGMENT IN NEXT CB
                          0002 3316 GUF570 EQU   2                           LENGTH OF SEGMENT COUNT FIELD
                          0060 3317 GUF573 EQU   C'-'                        TEST BYTE FOR MINUS SIGN
                          0004 3318 GUF576 EQU   4                           PDRP CODE-DROP A RANGE
                          0001 3319 GUF579 EQU   1                           SDF CODE FOR 1ST SEG OF MULTI
                          0001 3320 GUF582 EQU   1                           LOCATION OF DISPLACEMENT OF 1
                          0003 3321 GUF585 EQU   3                           LOCATION OF DISPLACEMENT OF 3
                          0005 3322 GUF588 EQU   5                           RANGE DROP PARM LENGTH
                          0007 3323 GUF591 EQU   7                           NULL SEG GT 7 BEFORE PAK TRIED
                          0003 3324 GUF594 EQU   3                           NUMBER OF CB'S TO COMPRESS
                          0014 3325 GUF597 EQU   X'14'                       READ PARM 4 BS TO CB21 - CB4
                          0004 3326 GUF600 EQU   4                           NR OF DB'S BEING RE-ARRANGED
                          0001 3327 GUF603 EQU   1                           STARTING VALUE FOR COUNTER
                          0020 3328 GUF606 EQU   X'20'                       TEST BYTE-DISK READ LOCATION
                          0010 3329 GUF609 EQU   X'10'                       TEST BYTE-DISK READ LOCATION
                          00F0 3330 GUF612 EQU   X'F0'                       SET OFF DR LOCATION CODE CONTROL
                          00FF 3331 GUF615 EQU   X'FF'                       MINUS 1 - USED TO CALCULATE Q-CODES
                          0002 3332 GUF618 EQU   X'02'
                          0003 3333 GUF621 EQU   X'03'                       NEITHER 1ST NOR LAST SEGMENT
                          0001 3334 GUF624 EQU   1                           BYTE 0 OF SDF
                          0002 3335 GUF627 EQU   2                           BYTE 1 OF SDF
                          0003 3336 GUF630 EQU   3                           BYTE 2 OF SDF
                          0004 3337 GUF633 EQU   4                           LAST BYTE OF SDF
                          0044 3338 GUF636 EQU   68                          LENGTH OF FIT TRANSFER AREA
                          0006 3339 GUF639 EQU   6                           LINE NUMBER IN SEGMENT
     1A63 0C07            1A64 3340 GUF642 DC    AL2(GUF684)                 ADDRESS OF CB1
     1A65 80000000        1A68 3341 GUF645 DC    XL4'80000000'               NULL SEGMENT SDF
     1A69 0D07            1A6A 3342 GUF648 DC    AL2(GUF690)                 ADDRESS OF CB2
                          0001 3343 GUF651 EQU   1                           DISP OF DISK BLOCK IN WORK FILE
                          0003 3344 GUF654 EQU   3                           HIGHEST LINE NR IN DISK BLOCK
                          0004 3345 GUF657 EQU   4                           LENGTH OF NULL SEGMENT IN DB
     1A6B 1BFF            1A6C 3346 GUF660 DC    AL2(GUF924-2)               START ADDRESS OF NEW LINE
     1A6D 1D0B            1A6E 3347 GUF663 DC    AL2(GUF936+11)              ADDR OF 1ST ENTRY IN FIT
     1A6F 1FFB            1A70 3348 GUF666 DC    AL2(GUF936+763)             ADDR OF NEXT TO LST ENTRY IN FIT
     1A71 1FFF            1A72 3349 GUF669 DC    AL2(GUF936+767)             ADDR OF LAST ENTRY IN FIT
     1A73 1FEF            1A74 3350 GUF672 DC    AL2(GUF936+751)             RE-ORDER STOP CONTROL
     1A75 0004            1A76 3351 GUF675 DC    XL2'04'                     LENGTH OF FIT ENTRY OR SDF
     1A77 FFFC            1A78 3352 GUF678 DC    XL2'FFFC'                   NEGATIVE LENGTH OF FIT ENTRY
     1A79                 1A7A 3353 GUF681 DS    CL2                         LAST DELETE PARM ADDRESS
     1A79                      3354        ORG   *-2                         * INITIALIZE TO POINT TO 1ST SLOT
     1A79 1C0A            1A7A 3355        DC    AL2(GUF927+5)               * IN LIST
```

```
        DL4ICS - 4 SURFACE DISK IOCS ROUTINE


ERR LOC   OBJECT CODE     ADDR STMT SOURCE STATEMENT           VER 15, MOD 00  29/02/16  PAGE  75

                              3357 *
                              3358 ***     UPDATER ENTRY POINT
                              3359 *
         1A7B 3360 GUFUPD EQU    *                              UPDATER ENTRY POINT
         1A7B 3361 GUF867 EQU    *                              UPDATER ENTRY POINT
         1A83 3362        USING GUF870,@BR                      SET LOCAL BASE
    1A7B C2 01 1A83           3363        LA    GUF870,@BR                      *
    1A7F 35 02 1D09           3364        L     GUF942,@XR               POINT XR TO LAST FIT ENTRY USED
                              3365 *
                              3366 ***     FILE INDEX TABLE SEARCH
                              3367 *
    1A83 34 02 1D09           3368 GUF870 ST    GUF942,@XR               SAVE XR FOR COMPARE
    1A87 0D 01 1D09 1A72      3369        CLC   GUF942(@CADDR),GUF669    HAS END OF FIT BEEN REACHED ?
    1A8D F2 81 CE             3370        JE    GUF900                   YES, GO SET ERROR CODE
    1A90 8D 01 03 1C05        3371        CLC   GUF654(GUF507,@XR),GUF927 NEW LINE NUMBER IN THIS DB ?
    1A95 D0 02 1C             3372        BNL   GUF873(,@BR)             COULD BE, GO CHECK FURTHER
    1A98 36 02 1A76           3373        A     GUF675,@XR               NO, INCR XR TO NEXT FIT ENTRY
    1A9C D0 87 00             3374        B     GUF870(,@BR)             GO CHECK NEW LINE NR AGAIN
    1A9F 0D 01 1D09 1A6E      3375 GUF873 CLC   GUF942(@CADDR),GUF663    FIRST FIT ENTRY ?
    1AA5 D0 81 37             3376        BE    GUF879(,@BR)             YES, GO READ DB
    1AA8 36 02 1A78           3377        A     GUF678,@XR               NO, DECR XR TO PRIOR FIT ENTRY
    1AAC 2D 01 1C05 03        3378        CLC   GUF927(GUF507),GUF654(,@XR)  NEW LINE NR IN PRIOR DB ?
    1AB1 D0 84 34             3379        BH    GUF876(,@BR)             NO, GO READ NEXT DB
    1AB4 D0 87 00             3380        B     GUF870(,@BR)             GO CHECK NEW LINE NR AGAIN
                              3381 *
                              3382 ***     READ DB'S AND INITIALIZE 'PAK' WORK FIELDS
                              3383 *
    1AB7 E2 02 04             3384 GUF876 LA    GUF540(,@XR),@XR         INCR FOR NEXT FIT
    1ABA 3C 11 1464           3385 GUF879 MVI   GUF195,GUF534            SET DISK READ PARAMETER
    1ABE C0 87 1389           3386        B     GUF138                   GO READ DB1 INTO CB1
    1AC2 3C FF 1A43           3387        MVI   GUF489-GUF411,GUF414     SET 2ND CIT ENTRY NULL
    1AC6 2C 03 1A3B 04        3388        MVC   GUF486(GUF540),GUF657(,@XR)  MOVE FIT ENTRY TO CIT WORKAREA
    1ACB 0C 03 0D0B 1A68      3389        MVC   GUF690+GUF540(GUF540),GUF645 MOVE NULL SDF TO CB2
    1AD1 3C 04 1A28           3390        MVI   GUF438,GUF540            SET TO 'PAK' 4 CB'S
    1AD5 3C 32 1464           3391        MVI   GUF195,GUF543            SET DISK READ PARM
    1AD9 E2 02 04             3392        LA    GUF540(,@XR),@XR         SET XR FOR READ SUBROUTINE
    1ADC C0 87 1389           3393        B     GUF138                   GO READ FROM DISK
    1AE0 C2 02 0C07           3394        LA    GUF684,@XR               LOAD CB1 ADDRESS TO XR
    1AE4 B8 80 01             3395 GUF882 TBN   GUF624(,@XR),GUF546      TEST FOR NULL SEGMENT
    1AE7 C0 10 1B5E           3396        BT    GUF900                   YES, GO HANDLE HARD ERROR
    1AEB B9 02 03             3397        TBF   GUF630(,@XR),GUF549      1ST OR ONLY SEGMENT ?
    1AEE D0 90 7E             3398        BF    GUF885(,@BR)             NO, BYPASS LINE NUMBER CHECK
    1AF1 8D 01 06 1C05        3399        CLC   GUF639(GUF507,@XR),GUF927 TEST SEGMENT LINE NUMBER
    1AF6 D0 81 91             3400        BE    GUF888(,@BR)             EQUALS NEW LINE NUMBER
    1AF9 D0 84 C1             3401        BH    GUF897(,@BR)             GREATER THAN NEW LINE NUMBER
                              3402 *     LESS THAN NEW LINE NUMBER - CONTINUE SEARCH
    1AFC 2C 01 1A3A 06        3403        MVC   GUF483(GUF507),GUF639(,@XR) REPLACE HIGHEST LINE NUMBER
    1B01 2E 00 1A35 02        3404 GUF885 ALC   GUF468,GUF627(1,@XR)     ADD SEGMENT LENGTH TO COUNTER
    1B06 3D FB 1A35           3405        CLI   GUF468,GUF552            CHECK FOR CB OVERRUN
    1B0A C0 84 1B5E           3406        BH    GUF900                   YES, GO HANDLE HARD ERROR
    1B0E B6 02 02             3407        A     GUF627(,@XR),@XR         INCR XR FOR NEXT SDF
    1B11 D0 87 61             3408        B     GUF882(,@BR)             CONTINUE SEARCH
                              3409 *
                              3410 ***     SEG LINE NR EQUALS NEW LINE NR - NEW LINE IS EITHER A REPLACEMENT
                              3411 ***     OR A DELETION.
                              3412 *
```

```
     1B14 B8 01 03              3413 GUF888 TBN    GUF630(,@XR),GUF555      MORE SEGMENT IN OLD LINE ?
     1B17 D0 90 9B              3414        BF     GUF891(,@BR)            NO, BYPASS NEXT INSTRUCTION
     1B1A 3A 02 1A23            3415        SBN    GUF429,GUF564           YES, SET DROP INDR
     1B1E 2E 00 1A3B 02         3416 GUF891 ALC    GUF486,GUF627(1,@XR)    INCR UNUSED SPACE COUNTER
     1B23 2C 01 1A37 02         3417        MVC    GUF471(GUF570),GUF627(,@XR) SET LENGTH OF SEG IN FREE SPACE
     1B28 38 20 03D5            3418        TBN    $INDR2,$FUIND           IS STATEMENT A REPLACEMENT ?
     1B2C C0 10 0D41            3419        BT     GUF834                  YES, GO INSERT IT
                                3420 *
                                3421 ***    NEW LINE IS A DELETION
                                3422 *
     1B30 0F 01 1D02 1A55       3423        SLC    GUF939(2),GUF510        DECR FILE LINE COUNTER
     1B36 3D 60 1C06            3424        CLI    GUF933+1,GUF573         DELETING A RANGE ?
     1B3A F2 01 2D              3425        JNE    GUF903                  NO, JUMP TO 'PAK' BRANCH
     1B3D 3C 04 1A23            3426 GUF894 MVI    GUF429,GUF576           SET ON DELETE RANGE INDR
     1B41 F2 87 26              3427        J      GUF903                  JUMP TO 'PAK' BRANCH
                                3428 *
                                3429 ***    NEW LINE NUMBER NOT ON FILE
                                3430 *
     1B44 38 20 03D5            3431 GUF897 TBN    $INDR2,$FUIND           WAS A STATEMENT PASSED ?
     1B48 C0 10 0D0C            3432        BT     GUF822                  YES, GO TO NEW LINE ROUTINE
     1B4C 3D 60 1C06            3433        CLI    GUF933+1,GUF573         IS A RANGE BEING DELETED ?
     1B50 F2 01 1B              3434        JNE    GUF906                  NO, GO CHECK FOR DELETE LIST
     1B53 8D 01 06 1C08         3435        CLC    GUF639(GUF507,@XR),GUF930 IS LINE NR WITHIN THE RANGE ?
     1B58 D0 04 9B              3436        BNH    GUF891(,@BR)            YES, GO ADJUST COUNTERS
     1B5B F2 87 10              3437        J      GUF906                  NO, GO CHECK FOR DELETE LIST
                                3438 *
                                3439 ***    THE FIT DOES NOT AGREE WITH DATA BLOCKS IN THE FILE WORK AREA.
                                3440 ***    FURTHER MANIPULATION WITH THIS FILE IS IMPOSSIBLE.
                                3441 *
     1B5E 3C 97 03CD            3442 GUF900 MVI    $CAERR,@@E550           SET BAD WORK FILE ERROR CODE
     1B62 3A 04 03D6            3443        SBN    $INDR3,$ERHRD           SET INDR FOR HARD HALT
     1B66 C0 87 0469            3444        B      $CAERK                  GO TO ERRPGM INTERFACE
                                3445 *
                                3446 ***    FILE HAS BEEN UPDATED.  PASS CONTROL TO 'PAK' TO CONDENSE CORE
                                3447 ***    BLOCKS AND WRITE THEM TO DISK.  'PAK'WILL ALSO CORRECT THE FIT.
                                3448 ***    RETURN FROM 'PAK' IS TO THE INSTRUCTION FOLLOWING THE CALL.
                                3449 *
     1B6A C0 87 1522            3450 GUF903 B      GUF201                  GO TO 'PAK'ROUTINE

                                3452 *
                                3453 ***    ROUTINE TO HANDLE THE DELETE PARAMETER LIST
                                3454 *
                           0001 3455        DROP   1                       DROP BASE ADDRESSING
     1B6E C0 80 0469            3456 GUF906 BC     $CAERK,@NOP             GO TO ERRPGM IF SET TO UCB
     1B72 38 40 03D5            3457        TBN    $INDR2,$FDIND           LINE NR LIST TO DELETE ?
     1B76 F2 10 19              3458        JT     GUF918                  YES, GO TO CHECK FOR MORE NUM'S
     1B79 F2 87 12              3459 GUF909 JC     GUF915,@UCB             BYPASS CARD READ IF KEY MODE
     1B7C C0 87 0465            3460 GUF912 B      $SPRNT                  PRINT ON SYSTEM PRINTER
     1B80 057F            1B81  3461        DC     AL2($WAITF)             PPL ADDRESS
     1B82 C0 87 048D            3462        B      $UNMSK                  HONOR INQUIRY REQUESTS
     1B86 3C 80 0476            3463        MVI    $CIMSK,@NOP             MASK INQUIRY REQUESTS
     1B8A C0 87 0890            3464        B      $$PRES                  GO READ A CARD
     1B8E C0 87 1107            3465 GUF915 B      GUGENT                  GO TO CRUSHER
     1B92 35 02 1A7A            3466 GUF918 L      GUF681,@XR              LOAD ADDR OF LAST PARM TO XR
     1B96 BD 60 01              3467        CLI    GUF582(,@XR),GUF573     WAS IT A RANGE DELETE ?
     1B99 F2 01 03              3468        JNE    GUF921                  NO, GO CHECK FOR END OF LIST
```

```
ERR LOC   OBJECT CODE      ADDR STMT SOURCE STATEMENT              VER 15, MOD 00  29/02/16  PAGE  77

    1B9C E2 02 03           3469          LA    GUF585(,@XR),@XR         INCR POINTER PAST RANGE ENTRY
    1B9F BD FF 01           3470 GUF921 CLI    @B1(,@XR),@SCTSZ-1       IS NEXT PARM THE END INDR ?
    1BA2 C0 81 04A1         3471          BE    $CARPL                  YES, GO TYPE READY & ENABLE INPUT
    1BA6 C0 87 048D         3472          B     $UNMSK                  HONOR PENDING INQUIRY REQUESTS
    1BAA 3C 80 0476         3473          MVI   $CIMSK,@NOP             RE-MASK FOR NEXT DELETE PASS
    1BAE E2 02 02           3474          LA    GUF618(,@XR),@XR        INCR POINTER TO NEXT LINE NUMBER
    1BB1 34 02 1A7A         3475          ST    GUF681,@XR              SAVE ADDRESS OF PARAMETER
    1BB5 2C 04 1C08 03      3476          MVC   GUF927+GUF585,GUF585(GUF588,@XR)  MOVE PARM TO ACTIVE LOCATION
    1BBA 0F 0E 1A37 1A37    3477          SLC   GUF471(GUF537),GUF471   CLEAR WORK AREAS
    1BC0 3C 00 1A52         3478          MVI   GUF501,@ZERO            *
    1BC4 C0 87 1A7B         3479          B     GUF867                  GO TO UPDATER ENTRY

         1C01 3481 GUF924 EQU    $$SLIB+@SDF1             LENGTH OF NEW LINE, INCLUDES SDF
         1C05 3482 GUF927 EQU    GUF924+4                BINARY LINE NUMBER OF NEW LINE
         1C08 3483 GUF930 EQU    GUF927+3                RANGE DELETE UPPER LIMIT
         1C05 3484 GUF933 EQU    GUF927                  ADDRESS OF 2ND BYTE OF DELETE LIST
         1D00 3485 GUF936 EQU    X'1D00'                 1ST BYTE OF FIT
         1D02 3486 GUF939 EQU    GUF936+2                NUMBER OF LINES IN FILE
         1D09 3487 GUF942 EQU    GUF936+9                XR FOR LAST LINE NUMBER FOUND
         1D0B 3488 GUF945 EQU    GUF936+11               ADDR OF 1ST NULL ENTRY IN FIT
```

```
      DL4ICS - 4 SURFACE DISK IOCS ROUTINE
```

```
                         3490 *        PATCH ,5
                         3491 ********************************************************************
                         3492 *        PATCH AREA 5
                         3493 ********************************************************************
                         3494 *
                         3495 ***     CALCULATE AREA LEFT IN THIS SECTOR
                         3496 *
                    1BC8 3497 $$$$L5 EQU    *                    START OF PATCH AREA 5
     1C00                3498        ORG    *,256,0              SET LOC CNTR TO NEXT SECTOR
                    1C00 3499 $$$$T5 EQU    *                    DEFINE ADDR OF SCTR BOUNDARY
     1BC8                3500        ORG    $$$$L5               SET LOC CNTR TO START OF
                         3501 *                                 * PATCH AREA
     1BC8           1BFF 3502 $$$$$5 DS     CL($$$$T5-$$$$L5)    PATCH AREA
                         3503 ********************************************************************

                    FFFF 3505        END

TOTAL STATEMENTS IN ERROR IN THIS ASSEMBLY =      0
```

```
$$$$$$   001  0C00 1975
$$$$$1   027  0CFF 2086
$$$$$2   025  0D8F 2131
$$$$$3   097  0FFF 2351
$$$$$4   012  10FF 2416
$$$$$5   056  1BFF 3502
$$$$L1   001  0CE5 2081   2084   2086
$$$$L3   001  0F9F 2346   2349   2351
$$$$L4   001  10F4 2411   2414   2416
$$$$L5   001  1BC8 3497   3500   3502
$$$$T1   001  0D00 2083   2086
$$$$T3   001  1000 2348   2351
$$$$T4   001  1100 2413   2416
$$$$T5   001  1C00 3499   3502
$$$CMD   001  0020 1751
$$$DAT   001  0040 1750
$$$EPL   001  0091 1747
$$$ERN   001  0080 1801
$$$FUN   001  0010 1752
$$$NLN   001  00A0 1797
$$$STD   001  0081 1746
$$$001   015  0C73 2009
$$BNLN   001  0605 1727   1729   2214   2216
$$CDBS   001  08C0 1777   2582
$$CDND   001  0666 1736   2656
$$CDRD   001  0890 1775   1777
$$CKEY   001  0603 1725   2612
$$CKFF   001  0B3D 1757
$$COFF   001  0B44 1756
$$CSNS   001  209C 1786
$$DATB   001  0BBF 1758
$$EOSA   001  0AFE 1755
$$ERSK   001  1C00 1796
$$FITS   001  1D00 1804   2285
$$FLIB   001  06FF 1803
$$ILEN   001  0601 1721   1723   1727
$$ILHD   001  0600 1719   1721
$$INLN   001  0607 1734   1736   1738   2611
$$INND   001  06FA 1738   2211   2641*  2642   2642*
$$KBDT   001  09E1 1745   1749
$$KBSN   001  09E2 1749   1754
$$KLD1   001  0600 1809
$$KLD2   001  0700 1811   2238*  2239   2239*
$$KLD3   001  0C00 1813
$$LPOS   001  09EB 1754
$$PCNT   001  07E9 1770
$$PLYN   001  2004 1784
$$PRES   001  0890 1743   1745   1755   1756   1757   1758   1775   2248   2643   3464
$$PRFL   001  2143 1788
$$PRNT   001  0707 1764   1765   1769   1770
$$PRTN   001  0782 1765
$$PSIO   001  07CE 1769
$$PYCD   001  2200 1790
$$PYMP   001  2000 1782   1784   1786   1788   1790
$$SLIB   001  1C00 1799   2211*  2213*  2214*  2215*  2290   3481
$$TPCD   001  0606 1729   1734
```

SYMBOL	LEN	VALUE	DEFN	REFERENCES
$$UPAR	001	0602	1723	1725 2175* 2240*
$$WSPB	001	1E00	1802	
$$XIND	001	06FF	1800	1803
$$ZERO	001	0000	1314	1315 1317 1318 1319 1323 1782
$ABORT	001	0010	1427	
$BASIC	001	0080	1485	
$BIGCD	001	0080	1561	
$BLDPL	001	0579	1694	1696
$BLNOE	001	0569	1684	
$BLOAD	001	0522	1675	1677 1680 1693 1694 2636
$BLRTN	001	0550	1683	1684
$BRSAV	001	03C5	1372	1373
$BSADR	001	0587	1699	1701
$BUFPT	001	03E3	1580	1581
$CABLD	001	04B4	1653	1654
$CAERK	001	0469	1630	1633 2599 2601 2609 3444 3456
$CAERR	001	03CD	1378	1380 2098* 2107* 2226* 2608* 3111* 3442*
$CAIPL	001	049D	1649	1651
$CALLI	001	0008	1570	2154 2583
$CARDI	001	0001	1341	2140 2147 2156 2161 2221 2242 2256 2577 2580 2618
$CARPL	001	04A1	1651	1653 3471
$CIENT	001	0483	1640	1641
$CIEXT	001	0480	1639	1640
$CIMSK	001	0476	1636	1639 2136* 2589* 3463* 3473*
$CISUS	001	0496	1644	1649
$CLBFR	001	0010	1528	2236 2241
$CMDKY	001	0008	1440	
$CMODE	001	0002	1490	2188 2634
$CONFG	001	03DD	1553	1563
$CRPOS	001	03E2	1579	1580
$CRTAD	001	044D	1618	1619
$CRTAV	001	0002	1434	
$CRTDN	001	0002	1458	
$CRTIN	001	03D3	1455	1462
$CRTNO	001	0004	1437	
$CRTPU	001	0004	1459	
$CRTSP	001	0008	1460	
$CRTUP	001	0001	1457	
$CRUSH	001	0080	1566	2449
$CSDPL	001	050E	1665	1666
$C0001	001	0464	1622	1628
$DATE	001	043A	1603	1604
$DBGUF	001	03E0	1565	1574 2145 2148* 2154 2449 2486 2583
$DBLOK	001	0001	1515	
$DFDET	001	03E8	1586	1587
$DISKN	001	0025	1317	2193 2254 2264 2472 2521 2587 2715 2841 2945
$DKERR	001	0008	1496	
$DKSIZ	001	03D7	1540	1548 1589
$DK100	001	0001	1542	
$DK200	001	0002	1543	
$DK400	001	0004	1544	
$DK600	001	0008	1545	
$DK800	001	0010	1546	
$DPLSV	001	0449	1614	1616
$DTNMB	001	0040	1361	2223
$DTRDR	001	0040	1449	

```
$ENDNU  001  0600 1708    1719   1743   1764   1800   1809   1811   1813
$ERDPL  001  046F 1633    1635
$ERFIL  001  0040 1388    2607
$ERHRD  001  0004 1520    3443
$ERKEY  001  0080 1392    2227
$ERLOG  001  0345 1322
$ERMAD  001  0472 1635    1636
$ERPND  001  0004 1493
$ERRCT  001  03CF 1394
$ERRPG  001  03CE 1382    2227*  2607*
$ERSFL  001  0035 1387
$ERSTK  001  0030 1385
$ER050  001  0363 1323
$ER1N2  001  0050 1390
$EXADR  001  0517 1668    1670
$EXCMD  001  0001 1422
$EXFTR  001  043B 1604    1609
$FCIND  001  0010 1500    2167   2205   2252   2443
$FDIND  001  0040 1507    2167   2203   2443   3457
$FEARR  001  0004 1315
$FEMAP  001  0588 1701    1702
$FILIB  001  03DA 1551    1552
$FITIN  001  0010 1476    2190   2196
$FUIND  001  0020 1505    2167   2207   2250   2443   3418   3431
$GUFIO  001  0583 1698    1699
$GUFIR  001  0008 1350    2170   2442   2630
$HISTE  001  042E 1601    1602
$HIST1  001  0435 1602    1603
$HRDER  001  0020 1446
$INDR1  001  03D4 1462    1488   2190   2196*  2197   2209   2259   3142*  3144*
$INDR2  001  03D5 1488    1513   2137   2139*  2167*  2188   2203   2205   2207   2250   2252   2443*  2634
                          3418   3431   3457
$INDR3  001  03D6 1513    1540   2236   2241*  2247   2249*  3443*
$INLNO  001  03CF 1380    1382   1394   1401   3114*
$INRPT  001  0020 1358    2168   2176
$IOIND  001  03D2 1429    1455   2246*  2640*
$IOPGS  001  0010 1569
$IOYES  001  0002 1344    2147   2149   2153
$IPLDV  001  05FF 1705    1708
$IRKEY  001  0020 1568    2145   2148
$KEYBD  001  03E1 1574    1579
$KEYCD  001  03C3 1338    1372   2140   2147*  2149   2153*  2156   2161   2168   2170   2176*  2221   2223
                          2242   2256   2442*  2577   2580   2585   2592   2603   2618   2623   2630*
$KEYDT  001  0040 1482    2209   2259
$KE090  001  00DE 1318
$KE130  001  01D5 1319
$KYBSY  001  0010 1355    2585   2592
$LDRTN  001  0571 1693
$LEVEL  001  03DF 1563    1565
$LIST   001  0002 1517
$LMRGN  001  03C1 1333    1335
$LNPTR  001  0080 1452
$LOADB  001  054A 1677
$LOADR  001  051A 1670    1673   2151   2158   2163
$LPRIO  001  03EA 1587
$LPROS  001  03E5 1582    1584
```

```
$LPRP3  001  03E4 1581    1582
$MOUNT  001  0020 1531
$MPDWN  001  0001 1431
$NEXTB  001  03E6 1584    1585
$NEXTL  001  03E7 1585    1586
$NOENB  001  0008 1523    2247   2249
$NOLST  001  0004 1347    2147   2623
$NUCBS  001  03C0 1330    1331   2605   2606
$NWRKF  001  0080 1536
$NWRKR  001  0040 1533
$PASWD  001  042D 1600    1601
$PAUSD  001  04BA 1654    1656
$PAUSE  001  0002 1424
$PGMDT  001  0020 1479    2197
$PGMST  001  0010 1443    2246   2640
$PKERT  001  0419 1598    1600
$PLST1  001  0454 1619    1620
$PLST2  001  045B 1620    1621
$PLST3  001  0462 1621    1622
$PRDEV  001  044B 1616    1618
$PRESN  001  0002 1467
$PROCI  001  0001 1464
$PRPOS  001  03C2 1335    1338
$PSDBR  001  04FA 1659
$PSDXR  001  04F2 1658    1659
$PSTEP  001  0004 1425
$PSTMT  001  0008 1426
$PTCH1  001  03F5 1589    1593
$READY  001  0080 1509    2137   2139
$REORD  001  0040 1567    2486
$RLOAD  001  051E 1673    1675   2638
$RMRGN  001  03C0 1331    1333
$RSTR   001  04D6 1656    1658   1660   1665
$RUNIT  001  0001 1403
$SFAID  001  050D 1661
$SPRNT  001  0465 1628    1630   2142   2165   2172   2177   2180   2230   2233   2625   3460
$SRTRN  001  04FE 1660    1661
$STEPT  001  0002 1404
$SWPCR  001  0511 1666    1668
$TABLN  001  03CB 1375    1378   2218*  2219*
$TFLOW  001  0008 1410
$TRACE  001  0004 1405
$TRALL  001  0010 1411
$TROVR  001  054E 1680    1683
$TRUNK  001  0080 1363    2603
$TRVAR  001  0020 1412
$UNMSK  001  048D 1641    1644   2579   3462   3472
$USRDR  001  03DC 1552    1553
$VMDEF  001  0080 1416
$VOLF1  001  03FE 1595    1596
$VOLF2  001  040E 1597
$VOLID  001  03F6 1593    1594   1598
$VOLR1  001  03F6 1594    1595
$VOLR2  001  0406 1596    1597
$WAITF  001  057F 1696    1698   2143   2181   2234   2255   2265   2473   2522   2588   2716   2946   3461
$WFDEF  001  0040 1610    2199
```

SYMBOL	LEN	VALUE	DEFN	REFERENCES	
$WFLOK	001	0008	1473		
$WFNME	001	0443	1609	1614	2199
$WSIND	001	0004	1470	3142	3144
$XIND1	001	03D0	1401	1420	
$XIND2	001	03D1	1420	1429	
$XIND3	001	03D8	1548	1551	
$XPREC	001	0040	1413		
$XRSAV	001	03C7	1373	1375	
$ZTRAD	001	05A2	1702		
$12K	001	0004	1557		
$16CKY	001	0008	1559		
$16K	001	0002	1556		
$22IMP	001	0001	1554		
#$$#BL	001	0000	1165		
#$$#CK	001	0000	1293		
#$$#CN	001	0000	1261		
#$$#CO	001	0000	1053		
#$$#CS	001	0000	1113		
#$$#DR	001	0000	0857		
#$$#ER	001	0000	1057		
#$$#FS	001	0000	1153		
#$$#IN	001	0000	1297		
#$$#PW	001	0000	1301		
#$$#RS	001	0000	1133		
#$$#SA	001	0000	1121		
#$$#SS	001	0000	1117		
#$$#VU	001	0600	1077		
#$$#0T	001	0700	0849		
#$$#1T	001	0000	0853		
#$$BCO	001	0600	0865		
#$$BOV	001	0800	1137		
#$$DPR	001	0700	0873	2292	
#$$DRE	001	0889	0889	2275	
#$$DSP	001	2800	0909		
#$$ECM	001	0C00	1169	2648	2653
#$$EFK	001	0C00	1189		
#$$ERR	001	0C00	1161		
#$$EXM	001	0C00	1049		
#$$FIL	001	0E00	1129		
#$$FIS	001	0E00	1125		
#$$FML	001	0200	1257		
#$$FMS	001	0200	1097		
#$$GRA	001	0889	1021	2280	
#$$GUF	001	0C00	1157	1974	
#$$INL	001	0600	1237		
#$$INS	001	0600	0861		
#$$KAL	001	0C00	1025		
#$$KCA	001	0C00	1241		
#$$KCH	001	0C00	0993		
#$$KCN	001	0C00	1109		
#$$KCT	001	0C00	0961		
#$$KDE	001	0C00	0957		
#$$KDI	001	0D00	1037		
#$$KDN	001	0C00	0945		
#$$KDO	001	0E00	1041		
#$$KED	001	0C00	0881		

```
#$$KEN   001  0C00 0885
#$$KEX   001  0C00 0905
#$$KGO   001  0C00 0877
#$$KHE   001  0C00 1061
#$$KKE   001  0C00 1289
#$$KLI   001  0C00 0965
#$$KLL   001  0920 1265
#$$KLO   001  0C00 0969
#$$KME   001  0D00 0949
#$$KMO   001  0C00 0893
#$$KNA   001  0C00 1005
#$$KOV   001  0E00 0925
#$$KPA   001  0C00 0901
#$$KPO   001  0C00 0989
#$$KPR   001  0C00 1013
#$$KRE   001  0C00 0933
#$$KRL   001  0700 1029
#$$KRM   001  0C00 0897
#$$KRN   001  0700 0917
#$$KRO   001  0D00 0921
#$$KRS   001  0C00 1245
#$$KRU   001  0C00 0941
#$$KRV   001  0800 1033
#$$KSA   001  0C00 0977
#$$KSE   001  0E00 1017
#$$KSO   001  0C20 1069
#$$KSS   001  0C00 1001
#$$KSV   001  0980 0997
#$$KSY   001  0C00 1009
#$$KWI   001  0C00 0937
#$$KWR   001  0C00 0929
#$$LOA   001  0600 0869
#$$MIP   001  0C00 1065
#$$SDS   001  0C00 1177
#$$SFF   001  0E00 1181
#$$SFL   001  0F00 1173
#$$SFO   001  1500 1145
#$$SFS   001  0C00 1141
#$$SPA   001  0C00 0981
#$$SPO   001  0806 0985
#$$SPS   001  0C00 0973
#$$STR   001  1600 1149
#$$TDC   001  1000 0953
#$$TSY   001  1000 0913
#$$TVK   001  0FC0 1089
#$$UAL   001  0C00 1105
#$$UAT   001  0900 1201
#$$UCD   001  0900 1209
#$$UCN   001  0C00 1193
#$$UCP   001  0700 1197
#$$UDE   001  0C00 1213
#$$UDI   001  0C00 1217
#$$UEX   001  0C00 1101
#$$UIN   001  0C00 1205
#$$UPA   001  0C00 1185
#$$UPO   001  0C00 1253
```

```
#$$UPT   001  0C00 1249
#$$VCR   001  2000 1045
#$$VLO   001  0600 1081
#$$VOD   001  0600 1085
#$$VVM   001  0000 1093
#$$VXI   001  0600 1073
#$$ZDU   001  1100 1225
#$$ZLB   001  1100 1269
#$$ZLO   001  1100 1229
#$$ZLV   001  0F00 1285
#$$ZL1   001  0F00 1273
#$$ZL2   001  0F00 1277
#$$ZL3   001  0C00 1281
#$$ZTR   001  1000 1221
#$$ZUT   001  0C00 1233
#$#BLN   001  18D4 1164
#$#CKT   001  2118 1292
#$#CNF   001  2000 1260
#$#COR   001  0800 1052
#$#CSA   001  1000 1112
#$#DRT   001  0000 0856
#$#ERM   001  0928 1056
#$#FSP   001  1880 1152
#$#INV   001  212C 1296
#$#PWR   001  2300 1300
#$#RSP   001  1780 1132
#$#SAV   001  1180 1120
#$#SSA   001  1128 1116
#$#VUF   001  0B08 1076
#$#0TR   001  0000 0848
#$#1TR   001  0080 0852
#$@#BL   001  0001 1166
#$@#CK   001  0004 1294
#$@#CN   001  0001 1262
#$@#CO   001  003A 1054
#$@#CS   001  003A 1114
#$@#DR   001  0008 0858
#$@#ER   001  0032 1058
#$@#FS   001  0030 1154
#$@#IN   001  003A 1298
#$@#PW   001  00C0 1302
#$@#RS   001  0030 1134
#$@#SA   001  0108 1122
#$@#SS   001  0001 1118
#$@#VU   001  0002 1078
#$@#0T   001  0018 0850
#$@#1T   001  0018 0854
#$@BCO   001  0018 0866
#$@BOV   001  0018 1138
#$@DPR   001  0005 0874   2291
#$@DRE   001  0001 0890   2274
#$@DSP   001  0004 0910
#$@ECM   001  0006 1170   2652
#$@EFK   001  0002 1190
#$@ERR   001  0003 1162
#$@EXM   001  0003 1050
```

```
SYMBOL   LEN VALUE DEFN   REFERENCES                          VER 15, MOD 00  29/02/16  PAGE  86

#$@FIL   001  0009 1130
#$@FIS   001  0009 1126
#$@FML   001  0052 1258
#$@FMS   001  0052 1098
#$@GRA   001  0003 1022   2279
#$@GUF   001  0010 1158
#$@INL   001  0010 1238
#$@INS   001  0010 0862
#$@KAL   001  000F 1026
#$@KCA   001  000C 1242
#$@KCH   001  000C 0994
#$@KCN   001  0010 1110
#$@KCT   001  0009 0962
#$@KDE   001  0010 0958
#$@KDI   001  0005 1038
#$@KDN   001  0010 0946
#$@KDO   001  000C 1042
#$@KED   001  000E 0882
#$@KEN   001  0006 0886
#$@KEX   001  0003 0906
#$@KGO   001  0002 0878
#$@KHE   001  000C 1062
#$@KKE   001  0006 1290
#$@KLI   001  0011 0966
#$@KLL   001  0001 1266
#$@KLO   001  0008 0970
#$@KME   001  0003 0950
#$@KMO   001  0004 0894
#$@KNA   001  0008 1006
#$@KOV   001  0009 0926
#$@KPA   001  0005 0902
#$@KPO   001  000D 0990
#$@KPR   001  0009 1014
#$@KRE   001  0002 0934
#$@KRL   001  0004 1030
#$@KRM   001  0003 0898
#$@KRN   001  0003 0918
#$@KRO   001  000A 0922
#$@KRS   001  000A 1246
#$@KRU   001  0003 0942
#$@KRV   001  000D 1034
#$@KSA   001  0011 0978
#$@KSE   001  0004 1018
#$@KSO   001  000D 1070
#$@KSS   001  000B 1002
#$@KSV   001  0002 0998
#$@KSY   001  000F 1010
#$@KWI   001  0002 0938
#$@KWR   001  0002 0930
#$@LOA   001  0013 0870
#$@MIP   001  000D 1066
#$@SDS   001  0004 1178
#$@SFF   001  0008 1182
#$@SFL   001  0005 1174
#$@SFO   001  0003 1146
#$@SFS   001  0011 1142
```

SYMBOL	LEN	VALUE	DEFN	REFERENCES	VER 15, MOD 00 29/02/16 PAGE 87
#$@SPA	001	0004	0982		
#$@SPO	001	0003	0986		
#$@SPS	001	0001	0974		
#$@STR	001	0002	1150		
#$@TDC	001	0003	0954		
#$@TSY	001	0003	0914		
#$@TVK	001	0001	1090		
#$@UAL	001	0011	1106		
#$@UAT	001	000C	1202		
#$@UCD	001	000B	1210		
#$@UCN	001	0009	1194		
#$@UCP	001	000F	1198		
#$@UDE	001	000E	1214		
#$@UDI	001	0008	1218		
#$@UEX	001	000E	1102		
#$@UIN	001	000F	1206		
#$@UPA	001	0004	1186		
#$@UPO	001	0005	1254		
#$@UPT	001	0012	1250		
#$@VCR	001	0008	1046		
#$@VLO	001	0002	1082		
#$@VOD	001	0016	1086		
#$@VVM	001	0030	1094		
#$@VXI	001	0002	1074		
#$@ZDU	001	0008	1226		
#$@ZLB	001	0002	1270		
#$@ZLO	001	000C	1230		
#$@ZLV	001	0006	1286		
#$@ZL1	001	0007	1274		
#$@ZL2	001	000D	1278		
#$@ZL3	001	000A	1282		
#$@ZTR	001	0001	1222		
#$@ZUT	001	0014	1234		
#$BCOM	001	0080	0864		
#$BOLV	001	1780	1136		
#$DPRI	001	014C	0872	2289	
#$DREA	001	0200	0888	2273	
#$DSPL	001	0240	0908		
#$ECMA	001	1900	1168	2651	
#$EFKE	001	1990	1188		
#$ERRP	001	18C0	1160		
#$EXMS	001	07D4	1048		
#$FILN	001	1724	1128		
#$FIST	001	1700	1124		
#$FMLN	001	1E00	1256		
#$FMST	001	0D00	1096		
#$GRAP	001	0690	1020	2278	
#$GUFU	001	1880	1156		
#$INLN	001	1C84	1236		
#$INST	001	0020	0860		
#$KALL	001	06A4	1024		
#$KCAL	001	1CC4	1240		
#$KCHA	001	053C	0992		
#$KCND	001	0F80	1108		
#$KCTL	001	03BC	0960		
#$KDEL	001	035C	0956		

```
                             CROSS REFERENCE

SYMBOL   LEN VALUE DEFN   REFERENCES                          VER 15, MOD 00  29/02/16  PAGE  88

#$KDIS   001  0744 1036
#$KDNT   001  0300 0944
#$KDOV   001  0780 1040
#$KEDI   001  0188 0880
#$KENA   001  01C4 0884
#$KEXT   001  0234 0904
#$KGOS   001  0180 0876
#$KHEL   001  0A30 1060
#$KKEY   001  2100 1288
#$KLIS   001  0400 0964
#$KLLA   001  2004 1264
#$KLOG   001  0444 0968
#$KMER   001  030C 0948
#$KMOU   001  0204 0892
#$KNAM   001  05C0 1004
#$KOVM   001  0290 0924
#$KPAS   001  0220 0900
#$KPOO   001  0508 0988
#$KPRT   001  063C 1012
#$KREA   001  02BC 0932
#$KRLA   001  0700 1028
#$KRMO   001  0214 0896
#$KRNU   001  0280 0916
#$KROV   001  028C 0920
#$KRSU   001  1D24 1244
#$KRUN   001  02CC 0940
#$KRVL   001  0710 1032
#$KSAV   001  0488 0976
#$KSET   001  0680 1016
#$KSOV   001  0AC8 1068
#$KSSP   001  0594 1000
#$KSVL   001  058C 0996
#$KSYM   001  0600 1008
#$KWID   001  02C4 0936
#$KWRI   001  02B4 0928
#$LOAD   001  0100 0868
#$MIPP   001  0A80 1064
#$SDSY   001  192C 1176
#$SFFI   001  193C 1180
#$SFLO   001  1918 1172
#$SFOV   001  1844 1144
#$SFSY   001  1800 1140
#$SPAC   001  04CC 0980
#$SPOV   001  04DC 0984
#$SPSY   001  0484 0972
#$STRO   001  1850 1148
#$TDCK   001  0350 0952
#$TSYK   001  0250 0912
#$TVKB   001  0BAC 1088
#$UALL   001  0F00 1104
#$UATR   001  1A38 1200
#$UCDI   001  1AD8 1208
#$UCNF   001  19B8 1192
#$UCPL   001  19DC 1196
#$UDEL   001  1B24 1212
#$UDIS   001  1B5C 1216
```

```
#$UEXL  001  0EA8 1100
#$UINI  001  1A88 1204
#$UPAC  001  1980 1184
#$UPOV  001  1D24 1252
#$UPTF  001  1D5C 1248
#$VCRT  001  07B4 1044
#$VLOA  001  0B80 1080
#$VODK  001  0B88 1084
#$VVMR  001  0C00 1092
#$VXIT  001  0B00 1072
#$ZDUM  001  1BA4 1224
#$ZLBM  001  2008 1268
#$ZLOA  001  1BC4 1228
#$ZLVR  001  20B0 1284
#$ZL1M  001  2010 1272
#$ZL2M  001  2030 1276
#$ZL3M  001  2088 1280
#$ZTRA  001  1B9C 1220
#$ZUTM  001  1C14 1232
#@#BAD  001  0455 1842
#@#IO1  001  0459 1850
#@#IO2  001  045D 1851
#@#TAT  001  0941 1878
#@#TBA  001  09A1 1882
#@#TFS  001  0941 1876
#@#TSY  001  0941 1880
#@#VFP  001  0700 1868
#@#VLP  001  093D 1871
#@#WDB  001  050C 1863
#@#WFT  001  0500 1861   2283
#@@#BA  001  0001 1843
#@@#IO  001  0001 1855
#@@#SC  001  0002 1852
#@@#TA  001  0010 1879
#@@#TB  001  0010 1883
#@@#TS  001  0005 1881
#@@#TW  001  0020 1877
#@@#VM  001  0100 1872
#@@#WD  001  00BD 1864   3298
#@@#WF  001  0003 1862   2284
#@@#04  001  0004 1854
#@@#08  001  0008 1853
#@@BOV  001  0018 1831
#@@ECM  001  0006 1845   2647
#@@ERR  001  0003 1839
#@@GUF  001  0010 1835
#@@LDS  001  0002 1841
#@@SDS  001  0004 1837
#@@SFF  001  0008 1849
#@@SFL  001  0005 1847
#@@SFO  001  0005 1857
#@@SFS  001  0011 1833
#@@VSF  001  0010 1885
#@@VSL  001  000F 1886
#@@VTR  001  0001 1870
#@BOVL  001  0400 1830
```

```
                            CROSS REFERENCE

SYMBOL  LEN VALUE DEFN   REFERENCES              VER 15, MOD 00  29/02/16  PAGE  90

#@ECMA  001  0481 1844   2646
#@ERRP  001  0441 1838
#@GUFU  001  0401 1834
#@LDSV  001  044D 1840
#@SDSY  001  04AD 1836
#@SFFI  001  04BD 1848
#@SFLO  001  0499 1846
#@SFOV  001  04C4 1856
#@SFSY  001  0480 1832
#@VSFI  001  09A1 1884
#@VTRL  001  0708 1869
#@WAF1  001  0401 1829
#@WAR1  001  0400 1828
#GUFUD  001  0000 0001
@@E001  001  0000 0751   0753
@@E003  001  0001 0753   0755
@@E004  001  0002 0755   0757
@@E005  001  0003 0757   0759
@@E006  001  0004 0759   0761
@@E007  001  0005 0761   0763
@@E008  001  0006 0763   0765
@@E009  001  0007 0765   0767
@@E010  001  0008 0767   0769
@@E011  001  0009 0769   0771
@@E012  001  000A 0771   0773
@@E013  001  000B 0773   0775
@@E014  001  000C 0775   0777
@@E015  001  000D 0777   0779
@@E016  001  000E 0779   0781
@@E017  001  000F 0781   0783
@@E018  001  0010 0783   0785
@@E019  001  0011 0785   0787
@@E020  001  0012 0787   0789
@@E021  001  0013 0789   0791
@@E023  001  0014 0791   0793
@@E024  001  0015 0793   0795
@@E025  001  0016 0795   0797
@@E026  001  0017 0797   0799
@@E027  001  0018 0799   0801
@@E028  001  0019 0801   0803
@@E029  001  001A 0803   0805
@@E030  001  001B 0805   0807
@@E031  001  001C 0807   0809
@@E032  001  001D 0809   0811
@@E035  001  001E 0811   0813
@@E036  001  001F 0813   0815
@@E037  001  0020 0815   0817
@@E038  001  0021 0817   0819
@@E039  001  0022 0819   0821
@@E040  001  0023 0821   0823
@@E041  001  0024 0823   0825
@@E042  001  0025 0825   0827
@@E043  001  0026 0827   0829
@@E044  001  0027 0829   0831
@@E045  001  0028 0831   0833
@@E046  001  0029 0833   0835
```

```
@@E060  001  002A 0835   0837
@@E080  001  002B 0837
@@E100  001  0000 0223   0225
@@E101  001  0001 0225   0227
@@E102  001  0002 0227   0229
@@E103  001  0003 0229   0231
@@E110  001  0004 0231   0233
@@E112  001  0005 0233   0235
@@E113  001  0006 0235   0237
@@E114  001  0007 0237   0239
@@E115  001  0008 0239   0241
@@E116  001  0009 0241   0243
@@E117  001  000A 0243   0245
@@E120  001  000B 0245   0247
@@E122  001  000C 0247   0249
@@E123  001  000D 0249   0251
@@E124  001  000E 0251   0253
@@E129  001  000F 0253   0255
@@E130  001  0010 0255   0257
@@E131  001  0011 0257   0259
@@E133  001  0012 0259   0261
@@E134  001  0013 0261   0263
@@E135  001  0014 0263   0265
@@E136  001  0015 0265   0267
@@E137  001  0016 0267   0269
@@E138  001  0017 0269   0271
@@E139  001  0018 0271   0273
@@E142  001  0019 0273   0275
@@E143  001  001A 0275   0277
@@E150  001  001B 0277   0279
@@E151  001  001C 0279   0281
@@E160  001  001D 0281   0283
@@E162  001  001E 0283   0285
@@E163  001  001F 0285   0287
@@E164  001  0020 0287   0289
@@E200  001  0021 0289   0291
@@E205  001  0022 0291   0293
@@E210  001  0023 0293   0295
@@E211  001  0024 0295   0297
@@E212  001  0025 0297   0299
@@E213  001  0026 0299   0301
@@E215  001  0027 0301   0303
@@E216  001  0028 0303   0305
@@E217  001  0029 0305   0307
@@E220  001  002A 0307   0309
@@E221  001  002B 0309   0311
@@E222  001  002C 0311   0313
@@E223  001  002D 0313   0315
@@E225  001  002E 0315   0317
@@E226  001  002F 0317   0319
@@E227  001  0030 0319   0321
@@E228  001  0031 0321   0323
@@E229  001  0032 0323   0325
@@E230  001  0033 0325   0327
@@E232  001  0034 0327   0329
@@E234  001  0035 0329   0331
```

```
                              CROSS REFERENCE

SYMBOL   LEN VALUE DEFN    REFERENCES                    VER 15, MOD 00  29/02/16  PAGE  92

@@E237   001  0036 0331    0333
@@E240   001  0037 0333    0335
@@E241   001  0038 0335    0337
@@E242   001  0039 0337    0339
@@E248   001  003A 0339    0341
@@E249   001  003B 0341    0343
@@E250   001  003C 0343    0345
@@E251   001  003D 0345    0347
@@E252   001  003E 0347    0349
@@E253   001  003F 0349    0351
@@E254   001  0040 0351    0353
@@E255   001  0041 0353    0355
@@E256   001  0042 0355    0357
@@E300   001  0043 0357    0359
@@E301   001  0044 0359    0361
@@E302   001  0045 0361    0363
@@E303   001  0046 0363    0365
@@E304   001  0047 0365    0367
@@E305   001  0048 0367    0369
@@E308   001  0049 0369    0371
@@E310   001  004A 0371    0373
@@E315   001  004B 0373    0375
@@E316   001  004C 0375    0377
@@E320   001  004D 0377    0379
@@E325   001  004E 0379    0381
@@E330   001  004F 0381    0383
@@E335   001  0050 0383    0385
@@E338   001  0051 0385    0387
@@E340   001  0052 0387    0389
@@E350   001  0053 0389    0391
@@E351   001  0054 0391    0393
@@E352   001  0055 0393    0395
@@E360   001  0056 0395    0397
@@E361   001  0057 0397    0399
@@E362   001  0058 0399    0401
@@E371   001  0059 0401    0403
@@E380   001  005A 0403    0405
@@E390   001  005B 0405    0407
@@E400   001  005C 0407    0409
@@E410   001  005D 0409    0411
@@E415   001  005E 0411    0413
@@E417   001  005F 0413    0415
@@E420   001  0060 0415    0417
@@E430   001  0061 0417    0419
@@E432   001  0062 0419    0421
@@E433   001  0063 0421    0423
@@E450   001  0064 0423    0425
@@E451   001  0065 0425    0427
@@E460   001  0066 0427    0429
@@E461   001  0067 0429    0431
@@E464   001  0068 0431    0433
@@E465   001  0069 0433    0435
@@E466   001  006A 0435    0437
@@E467   001  006B 0437    0439
@@E469   001  006C 0439    0441
@@E470   001  006D 0441    0443
```

```
@@E471   001  006E 0443    0445
@@E473   001  006F 0445    0447
@@E474   001  0070 0447    0449
@@E475   001  0071 0449    0451
@@E476   001  0072 0451    0453
@@E477   001  0073 0453    0455
@@E478   001  0074 0455    0457
@@E479   001  0075 0457    0459
@@E480   001  0076 0459    0461
@@E481   001  0077 0461    0463
@@E482   001  0078 0463    0465
@@E483   001  0079 0465    0467
@@E484   001  007A 0467    0469
@@E485   001  007B 0469    0471
@@E486   001  007C 0471    0473
@@E487   001  007D 0473    0475
@@E488   001  007E 0475    0477
@@E489   001  007F 0477    0479
@@E490   001  0080 0479    0481
@@E491   001  0081 0481    0483
@@E492   001  0082 0483    0485
@@E493   001  0083 0485    0487
@@E494   001  0084 0487    0489
@@E495   001  0085 0489    0491
@@E496   001  0086 0491    0493
@@E497   001  0087 0493    0495
@@E498   001  0088 0495    0497
@@E500   001  0089 0497    0499
@@E501   001  008A 0499    0501
@@E530   001  008B 0501    0503  2107  3111
@@E531   001  008C 0503    0505  2098
@@E535   001  008D 0505    0507
@@E540   001  008E 0507    0509  2608
@@E541   001  008F 0509    0511
@@E542   001  0090 0511    0513
@@E543   001  0091 0513    0515
@@E544   001  0092 0515    0517
@@E545   001  0093 0517    0519
@@E546   001  0094 0519    0521
@@E547   001  0095 0521    0523
@@E548   001  FFFF 0727
@@E549   001  0096 0523    0525
@@E550   001  0097 0525    0527  3442
@@E551   001  0098 0527    0529
@@E552   001  0099 0529    0531
@@E553   001  009A 0531    0533
@@E554   001  009B 0533    0535
@@E555   001  009C 0535    0537
@@E556   001  009D 0537    0539
@@E558   001  009E 0539    0541
@@E570   001  009F 0541    0543
@@E571   001  00A0 0543    0545
@@E572   001  00A1 0545    0547
@@E573   001  00A2 0547    0549
@@E574   001  00A3 0549    0551  2226
@@E575   001  FFFF 0729
```

SYMBOL	LEN	VALUE	DEFN	REFERENCES	VER 15, MOD 00 29/02/16 PAGE 94
@@E578	001	00A4	0551	0553	
@@E579	001	FFFF	0731		
@@E580	001	FFFF	0733		
@@E585	001	00A5	0553	0555	
@@E595	001	FFFF	0735		
@@E597	001	FFFF	0737		
@@E598	001	FFFF	0739		
@@E600	001	00A6	0555	0557	
@@E601	001	00A7	0557	0559	
@@E602	001	00A8	0559	0561	
@@E603	001	00A9	0561	0563	
@@E604	001	00AA	0563	0565	
@@E606	001	00AB	0565	0567	
@@E607	001	00AC	0567	0569	
@@E608	001	00AD	0569	0571	
@@E609	001	00AE	0571	0573	
@@E610	001	00AF	0573	0575	
@@E611	001	00B0	0575	0577	
@@E612	001	00B1	0577	0579	
@@E613	001	00B2	0579	0581	
@@E614	001	00B3	0581	0583	
@@E700	001	00B4	0583	0585	
@@E701	001	00B5	0585	0587	
@@E710	001	00B6	0587	0589	
@@E712	001	00B7	0589	0591	
@@E713	001	00B8	0591	0593	
@@E714	001	00B9	0593	0595	
@@E715	001	00BA	0595	0597	
@@E716	001	00BB	0597	0599	
@@E717	001	00BC	0599	0601	
@@E718	001	00BD	0601	0603	
@@E720	001	00BE	0603	0605	
@@E721	001	00BF	0605	0607	
@@E723	001	00C0	0607	0609	
@@E724	001	00C1	0609	0611	
@@E725	001	00C2	0611	0613	
@@E726	001	00C3	0613	0615	
@@E727	001	00C4	0615	0617	
@@E728	001	00C5	0617	0619	
@@E729	001	00C6	0619	0621	
@@E730	001	00C7	0621	0623	
@@E732	001	00C8	0623	0625	
@@E752	001	00C9	0625	0627	
@@E753	001	00CA	0627	0629	
@@E754	001	00CB	0629	0631	
@@E755	001	00CC	0631	0633	
@@E756	001	00CD	0633	0635	
@@E757	001	00CE	0635	0637	
@@E758	001	00CF	0637	0639	
@@E759	001	00D0	0639	0641	
@@E760	001	00D1	0641	0643	
@@E761	001	00D2	0643	0645	
@@E762	001	00D3	0645	0647	
@@E763	001	00D4	0647	0649	
@@E764	001	00D5	0649	0651	
@@E765	001	00D6	0651	0653	

```
@@E766  001  00D7 0653    0655
@@E767  001  00D8 0655    0657
@@E768  001  00D9 0657    0659
@@E769  001  00DA 0659    0661
@@E770  001  00DB 0661    0663
@@E771  001  00DC 0663    0665
@@E772  001  00DD 0665    0667
@@E773  001  00DE 0667    0669
@@E774  001  00DF 0669    0671
@@E775  001  00E0 0671    0673
@@E776  001  00E1 0673    0675
@@E777  001  00E2 0675    0677
@@E778  001  00E3 0677    0679
@@E779  001  00E4 0679    0681
@@E780  001  00E5 0681    0683
@@E781  001  00E6 0683    0685
@@E782  001  00E7 0685    0687
@@E783  001  00E8 0687    0689
@@E784  001  00E9 0689    0691
@@E785  001  00EA 0691    0693
@@E786  001  00EB 0693    0695
@@E790  001  00EC 0695    0697
@@E791  001  00ED 0697    0699
@@E792  001  00EE 0699    0701
@@E793  001  00EF 0701    0703
@@E794  001  00F0 0703    0705
@@E795  001  00F1 0705    0707
@@E796  001  00F2 0707    0709
@@E797  001  00F3 0709    0711
@@E798  001  00F4 0711    0713
@@E800  001  FFFF 0741
@@E801  001  FFFF 0743
@@E802  001  FFFF 0745
@@E803  001  FFFF 0747
@@E804  001  FFFF 0749
@@E900  001  00F5 0713    0715
@@E901  001  00F6 0715    0717
@@E902  001  00F7 0717    0719
@@E903  001  00F8 0719    0721
@@E905  001  00F9 0721    0723
@@E906  001  00FA 0723    0725
@@E910  001  00FB 0725
@@M130  001  0C0B 1987    2173
@@M131  001  0C0F 1991    2178
@@M132  001  0C13 1995    2231
@@T130  001  0C17 1999    1989
@@T131  001  0C2E 2001    1993
@@T132  001  0C33 2003    1997
@ARR    001  0008 0017    2020  2312  2530  2576  2669  2809* 2810  2811* 2812  2895
@ASIGN  001  007C 0072
@ASTER  001  005C 0070
@BCRDL  001  0050 0089
@BE     001  0081 0044
@BF     001  0090 0053    2602
@BH     001  0084 0042
@BL     001  0082 0043
```

SYMBOL	LEN	VALUE	DEFN	REFERENCES											
@BLANK	001	0040	0066	2238	2614	2641									
@BM	001	0082	0055												
@BNE	001	0001	0047												
@BNH	001	0004	0045												
@BNL	001	0002	0046												
@BNM	001	0002	0058												
@BNOL	001	0020	0051												
@BNOZ	001	0008	0050												
@BNP	001	0004	0057												
@BNZ	001	0001	0059												
@BOL	001	00A0	0049												
@BOZ	001	0088	0048												
@BP	001	0084	0054												
@BR	001	0001	0014	2022	2023*	2037	2044	2046	2048	2048*	2050	2051	2051*	2057	2059*
				2309	2310	2311*	2312	2314	2314	2315	2316	2321	2321	2323	2323
				2324	2324	2325	2327	2327	2328	2329*	2386*	2396	2398	2399	2400
				2503	2504*	2506	2507	2509	2510	2512	2513	2515	2516	2517	2519
				2523	2531	2532	2532	2533	2534	2535	2536	2537	2541	2542	2543
				2544	2544	2547	2550	2550	2552	2555	2555	2557	2558	2559	2663
				2666	2667*	2668	2669	2670	2671	2672	2673	2674	2675	2676	2677
				2678	2679	2680	2681	2682	2682	2685	2686	2687	2688	2690	2691
				2692	2693	2694	2695	2699	2700	2701	2702	2704	2705	2706	2709
				2711	2713	2718*	2805	2806	2808*	2809	2810	2811	2812	2814	2815
				2815	2816	2818	2819	2821	2823	2823	2824	2824	2825	2827	2829
				2830	2830	2831	2833	2835	2836	2836	2837	2837	2838	2838	2839
				2846*	2866	2866	2868	2868	2869	2870	2871	2871	2872	2872	2873
				2874	2874	2875	2876	2877	2877	2878	2880	2880	2881	2881	2882
				2882	2883	2883	2884	2896*	2897	2898	2899*	2900	2917	2921*	2924
				2995*	2996	2997	3000	3002	3004	3005	3005*	3010	3013*	3018	3049
				3052	3054*	3091	3138*	3149*	3150	3152	3161	3161	3164*	3173*	3174
				3175	3175	3176	3176	3177	3177	3178	3179	3179	3180	3180	3181
				3181	3182	3182	3183	3184	3185	3185	3186	3188	3188	3189	3190
				3191	3191	3192	3193*	3194	3197	3197*	3205*	3213	3220	3220*	3227*
				3229	3233	3233*	3362	3363*	3372	3374	3376	3379	3380	3398	3400
				3401	3408	3414	3436								
@BT	001	0010	0052												
@BZ	001	0081	0056												
@B1	001	0001	0064	2023	2030	2032	2038	2039	2048	2049	2050	2051	2052	2211	2315
				2316	2367	2372	2379	2616	2642*	2912	2914	3098	3159	3162	3197
				3233	3470										
@CADDR	001	0002	0143	1989	1993	1997	2058	2100	2117	2118	2119	2123	2124	2164	2173
				2178	2231	2357	2358	2363	2364	2370	2371	2384	2387	2388	2392
				2468	2489	2531	2532	2540	2544	2550	2555	2621	2670	2671	2674
				2677	2678	2679	2682	2684	2690	2693	2698	2706	2815	2902	2953
				3047	3053	3055	3059	3073	3074	3076	3079	3083	3092	3100	3115
				3125	3126	3128	3131	3136	3151	3153	3155	3186	3190	3191	3238
				3369	3375										
@CARDL	001	0060	0088	1736											
@CHARA	001	00C1	0073												
@CHARF	001	00C6	0074												
@CHARR	001	00D9	0075												
@CHARZ	001	00E9	0076												
@CLOFF	001	0010	0095												
@CLON	001	0011	0094												
@COMMA	001	006B	0067												
@CPLUS	001	004E	0080												

```
SYMBOL   LEN VALUE DEFN   REFERENCES                          VER 15, MOD 00  29/02/16  PAGE  97

@DADDR   001  0002 0141   2814
@DBFR1   001  0004 0130   2881*
@DBFR2   001  0005 0131
@DCALK   001  0001 0082
@DCBCY   001  0009 0116
@DCBT1   001  0050 0118
@DCNT    001  0003 0129   2863
@DCST1   001  0040 0117
@DCTRL   001  0000 0126
@DCYL    001  0001 0127   2851
@DD2     001  0003 0031
@DGET    001  0001 0135   2272  2277  2282  2288  2645  2650  2722
@DOLAR   001  005B 0069
@DOP2    001  0004 0029   2363* 2364* 2387* 2388* 2810* 2814* 2815* 2886  2887  3131* 3136*
@DPLNG   001  0006 0133   2816  2850
@DPOS    001  0000 0134
@DPUT    001  0002 0136   2566
@DSAD    001  0002 0128   2852
@DSBCY   001  0004 0107
@DSCS1   001  0000 0108
@DSIVF   001  0003 0139
@DSPIN   001  0002 0132
@DTRSZ   001  0018 0086
@DVBCY   001  0007 0109
@DVRFY   001  0031 0137
@DWAIT   001  00FF 0138
@DWBCY   001  0005 0104   2567  2723
@DWSIZ   001  00C0 0106
@DWTB1   001  0003 0105
@DZERO   001  00F0 0065
@D1      001  0002 0027   2315* 2327* 2395* 2397  2397* 2993* 2996* 3011* 3124* 3130* 3135* 3167*
                          3215* 3218  3228
@EOF     001  001C 0078
@EOFTC   001  0075 0163
@EOS     001  001E 0077   2028  2628
@FDDBC   001  0000 0196
@FDE1    001  000C 0201
@FDFNA   001  000B 0199
@FDHLN   001  0002 0209
@FDLNC   001  0002 0194
@FDNSC   001  0003 0211
@FDSD    001  0000 0207
@FLACE   001  0009 0198
@FLDBC   001  0001 0197
@FLENT   001  0004 0202
@FLFNA   001  0002 0200
@FLHLN   001  0002 0210
@FLLNC   001  0002 0195
@FLNSC   001  0001 0212
@FLSD    001  0001 0208
@HDRLN   001  0007 0093   1764  2089  2354
@IAR     001  0010 0018
@INDEX   001  0001 0157   0158
@INST3   001  0003 0033
@INST4   001  0004 0034
@INST5   001  0005 0035
```

SYMBOL	LEN	VALUE	DEFN	REFERENCES
@INST6	001	0006	0036	
@I1IAR	001	00C0	0021	
@LINSZ	001	00F4	0085	1738 2211 2211* 2642
@MAPEN	001	0005	0090	
@MINCR	001	2000	0084	
@MINUS	001	0060	0081	
@NOP	001	0080	0041	2110 2136 2201 2244 2589 2601 2631 2819 2942 3068 3456 3463 3473
@NUMBR	001	007B	0071	
@OPD2	001	0004	0030	3207* 3212* 3215
@OP1	001	0003	0028	2020* 2021* 2022* 2310* 2312* 2357* 2358* 2370 2370* 2371* 2392* 2530* 2531* 2532 2544 2544* 2550* 2555* 2576* 2666* 2668* 2669* 2670* 2671* 2674* 2677* 2678* 2679* 2682 2690* 2693* 2706* 2806* 2812* 2895* 3092* 3115* 3278
@OP2	001	0005	0032	2122* 2123* 2124* 3059*
@PCTRL	001	0000	0150	
@PDATA	001	0003	0152	
@PGCSZ	001	0020	0083	0084
@PPLNG	001	0004	0149	
@PRCNT	001	0001	0151	
@PRETR	001	00C0	0155	1991 1995
@PRINT	001	0040	0153	0155 1987
@PSR	001	0004	0016	
@PWAIT	001	00FF	0159	
@P1IAR	001	0020	0019	
@P2IAR	001	0040	0020	
@Q	001	0001	0025	2110* 2225* 2244* 2316* 2324 2324* 2327 2355* 2356* 2365* 2366* 2393* 2394* 2445 2602* 2632* 2818* 2819* 2829* 2835* 2861 2862 2864 2873* 2875 2918* 2919* 2920 2920* 2941 2944* 3014* 3015* 3068* 3116* 3152* 3156* 3160* 3206* 3210* 3211* 3214 3214* 3217*
@REGL	001	0002	0013	
@RETRN	001	0080	0154	0155 2270 2271 2657 2658
@RLDWN	001	004F	0160	
@RTRNC	001	0080	0162	
@SBLN	001	0005	0171	2214*
@SBLNL	001	0002	0185	2214 3114
@SCTSZ	001	0100	0101	2239 3470
@SDFLN	001	0007	0091	
@SDF0	001	0000	0167	
@SDF1	001	0001	0168	2057* 2058* 3481
@SDF2	001	0002	0169	
@SDF3	001	0003	0170	2215*
@SECCY	001	0030	0087	
@SIST	001	0001	0182	
@SLASH	001	0061	0068	
@SLAST	001	0002	0184	
@SMIDL	001	0003	0183	
@SNULL	001	0080	0174	
@SONLY	001	0000	0181	
@STEXT	001	0007	0173	2023 2024 2211
@STYPE	001	0006	0172	2211* 2213*
@TBCNT	001	0000	0161	
@TBLEF	001	0010	0156	0158
@TBLIX	001	0011	0158	
@UCB	001	0087	0040	2225 2593 2632 2873 2944 3112 3116 3459
@UPARW	001	005A	0079	

```
@VADDR  001  0002 0142
@VENTA  001  0056 0114
@VMDDV  001  00FE 0115
@VMFD1  001  0000 0110
@VMFD2  001  0001 0111
@VMRS3  001  0002 0113
@VMTRL  001  0001 0112
@VOLID  001  0006 0092
@VQ     001  0001 0026
@WSFIT  001  0500 0102
@WSTBL  001  0503 0103
@XR     001  0002 0015   2021  2024* 2028  2030  2030  2032  2032  2038  2038* 2039  2039  2049
                         2049* 2050  2052  2052* 2060* 2122  2216* 2318  2362* 2373  2448* 2451
                         2453  2455  2455* 2462  2469  2470  2471  2474* 2475  2477  2478  2478*
                         2485* 2488  2491  2493  2493  2495  2495* 2500* 2505  2505  2507  2508
                         2508  2509  2511  2511  2512  2515  2517  2518  2518  2519  2534  2536
                         2538  2538* 2539  2542  2545  2605* 2606  2607  2608  2609  2610  2611*
                         2614  2616  2616* 2620  2628  2668  2683  2687  2689  2694  2696  2696*
                         2697  2700  2717* 2900* 2901* 2907  2909  2916  2919  2921  2922  2922*
                         2923  2927  2929  2931  2931* 2954* 2958  2961  2962  2962* 2963  2969
                         2971  2974  2975  2975* 2980  2982  2984  2988  2989  2994  2997  2998
                         3002  3007  3012  3013  3015  3016  3016* 3017  3017* 3018  3021* 3022
                         3025  3027  3029  3033  3035  3036  3037  3038  3067* 3075* 3098  3132
                         3168  3183* 3184  3189  3194  3198  3198* 3204* 3216  3221  3221* 3228*
                         3229  3232  3232* 3239* 3364* 3368  3371  3373* 3377* 3378  3384  3384*
                         3388  3392  3392* 3394* 3395  3397  3399  3403  3404  3407  3407* 3413
                         3416  3417  3435  3466* 3467  3469  3469* 3470  3474  3474* 3475  3476
@ZERO   001  0000 0063   2028  2030  2037* 2044  2046* 2050* 2175  2213  2240  2373* 2465  2554
                         2614  2628  2681  2709  2818  2900  2903  2923  2924* 2955  2977  3004
                         3005  3009  3011  3017  3018  3018* 3025  3040  3069  3071  3094  3120
                         3161  3165  3168  3194* 3217  3229  3478
C2DEC5  001  0F5B 2308   2217  2309  2311
C2DVAL  005  0F99 2336   2218  2321  2321  2321* 2323  2323
C2D020  003  0F6D 2316   2327  2328
C2D030  003  0F70 2318   2315* 2316* 2324  2324* 2325  2327*
C2D040  004  0F7A 2323   2319
C2D050  004  0F8C 2329   2310*
C2D052  004  0F90 2330   2312*
C2D901  001  0F94 2334   2314  2314  2314
C2D902  001  0F95 2335   2314
C2D903  005  0F9E 2337   2314  2314* 2321  2321  2321  2323  2323  2323  2323*
DL4CYL  001  14DC 2851   2823*
DL4C01  002  14E2 2859   2809  2811  2823
DL4C05  002  14E4 2860   2815
DL4C24  003  14B3 2862   2836
DL4C48  003  14A0 2864   2830  2871  2877
DL4C96  003  148F 2861   2824
DL4DPL  006  14E0 2850   2816*
DL4EFD  001  0001 2857   2829  2875
DL4END  001  1522 2888
DL4ETB  001  0080 2858   2835
DL4E01  001  0001 2856   2831
DL4E24  001  0018 2855   2833
DL4E48  001  0030 2854   2827  2869
DL4E96  001  0060 2853   2821
DL4ICS  001  1466 2804   2548  2560  2702  2711
```

```
                              CROSS REFERENCE

SYMBOL  LEN VALUE DEFN   REFERENCES                        VER 15, MOD 00   29/02/16   PAGE 100

DL4LST  001 14DB 2849   2842   2851   2852   2863   2881*
DL4SAV  005 147D 2887   2874*  2877*  2880
DL4SCD  001 14DD 2852   2821   2824*  2827   2830*  2833   2836*  2837   2837*  2838   2838*  2839*  2868
                        2874   2880*  2882*
DL4SCT  001 14DE 2863   2831   2866   2872*  2881   2882   2883*
DL4SPT  004 14E5 2867   2832
DL4WRK  005 147E 2886   2866*  2868*  2869   2871*  2872   2883
DL4010  001 146A 2807   2805   2808
DL4020  005 147A 2814   2810*  2886   2887
DL4030  005 1483 2816   2814*  2815*
DL4035  003 1488 2818   2884
DL4040  003 148E 2821   2825   2861
DL4050  003 149F 2827   2822   2864
DL4060  003 14AC 2831   2828
DL4070  003 14B2 2833   2862   2870   2876   2878
DL4080  004 14BF 2837   2834
DL4100  003 14C7 2839   2818*  2829*  2835*  2875
DL4200  003 14D0 2844   2819*  2873*
DL4500  004 14E5 2866   2867
DL4600  004 150F 2880   2844
DL4900  004 14D3 2846   2806*
DL4920  004 14D7 2847   2812*
GCPACK  001 0C74 2016   2261
GCPBFR  001 1C00 2290   2023   2024   2057*  2058*  2067
GCPMAX  001 001B 2071   2044
GCPONE  001 0CE2 2066   2046
GCPSTL  002 0CE4 2067   2058
GCPTWO  001 0002 2070   2032   2037   2039
GCP020  003 0C88 2028   2053
GCP050  003 0C9F 2038   2047
GCP080  003 0CB8 2048   2045
GCP090  004 0CBE 2050   2031   2033
GCP100  003 0CC2 2051   2040
GCP110  004 0CCC 2057   2029
GCP120  004 0CD6 2059   2022*
GCP130  004 0CDA 2060   2021*
GCP140  004 0CDE 2061   2020*
GUFCSH  001 1107 2440
GUFDIN  001 1389 2664
GUFENT  001 0D90 2132
GUFPAK  001 1522 2893
GUFUDI  001 0C07 1981   2419
GUFUPD  001 1A7B 3360
GUF000  003 110F 2444   2445
GUF003  003 1110 2445   2201*
GUF006  004 1112 2446   2447
GUF009  004 111A 2448   2444
GUF012  005 1125 2451   2456
GUF015  004 113A 2457   2450   2452   2487   2490   2492   2524
GUF018  004 1141 2462   2454
GUF021  003 1178 2475   2479
GUF024  004 1189 2480   2476
GUF027  004 1191 2485   2458
GUF030  004 119D 2488   2496
GUF033  004 11C2 2500   2494
GUF036  004 11F3 2515   2503   2504   2510   3138
```

```
                               CROSS REFERENCE

SYMBOL   LEN VALUE DEFN   REFERENCES                          VER 15, MOD 00  29/02/16  PAGE 101

GUF039   004  11FA 2517   2506
GUF042   004  1206 2520   2513   2516
GUF045   001  1217 2529   2523   3139
GUF048   004  1220 2532   2552
GUF051   005  1232 2537   2559
GUF054   005  1252 2545   2544*
GUF057   001  1271 2553   2541   2543
GUF060   004  1271 2554   2531* 2532   2544   2550* 2555*
GUF063   004  128A 2560   2547   2557
GUF066   004  1290 2562   2530*
GUF069   001  1294 2564   2507* 2509   2512   2515   2517* 2519   2536* 2537* 2542
GUF072   002  1296 2565   2550   2555   2674   2677   2693   2953   3053   3153
GUF075   001  1297 2566   2549   2561
GUF078   001  1299 2568   2533* 2534*
GUF081   001  129A 2569   2535* 2558*
GUF084   002  129C 2570   2532*
GUF087   001  129D 2575   2446   2457   2463
GUF090   004  12AC 2580   2578   3117   3118
GUF093   004  12BE 2585   2581
GUF095   004  12C5 2587   2594
GUF096   004  12CF 2590   2576* 3115*
GUF097   004  12D3 2592   2584
GUF098   004  12D7 2593   2602*
GUF099   004  12DF 2599   2586   2593   2600   3116*
GUF102   004  12FB 2611   2604
GUF105   003  1306 2614   2617
GUF108   004  1313 2618   2615
GUF111   003  1339 2628   2619
GUF114   004  133F 2630   2613   2622
GUF115   003  1343 2631   2632*
GUF116   004  134E 2634   2631
GUF117   004  135B 2638   2635
GUF120   004  1361 2640   2629
GUF123   001  1377 2645   2637
GUF126   001  137D 2650   2639
GUF129   002  1384 2655   2620* 2621
GUF132   002  1386 2656   2621
GUF135   001  1387 2657   2626
GUF138   001  1389 2665   2467   2502   3085   3386   3393
GUF141   005  13BC 2679   2663   2667   2676
GUF144   003  13C1 2680   2673
GUF147   003  13C4 2681   2705
GUF150   005  13DF 2688   2701
GUF153   005  13E4 2689   2671* 2678* 2679* 2690*
GUF156   005  1428 2706   2685   2699
GUF159   006  142D 2707   2670* 2674* 2677* 2682   2693* 2706*
GUF162   003  143F 2711   2692
GUF165   004  144C 2715   2710
GUF168   004  1452 2717   2668* 2714
GUF171   004  1456 2718   2666*
GUF174   004  145A 2719   2669*
GUF177   001  145E 2721   2703   2712
GUF180   001  145E 2722
GUF183   001  1460 2724   2686* 2687* 2727
GUF186   001  1461 2725   2681* 2688* 2704   2709   2713
GUF189   002  1463 2726   2682*
```

```
GUF192   001  145F  2727
GUF195   001  1464  2728    2466* 2501* 2672  2675  2680* 2691* 3078* 3082* 3385* 3391*
GUF198   001  1465  2729    2694* 2695* 2700
GUF201   001  1522  2894    2480  3450
GUF204   004  154D  2905    2925  2932  3062
GUF207   004  1566  2912    2935
GUF210   005  1574  2916    2913
GUF213   004  1579  2917    2915
GUF216   005  1592  2923    2918* 2919* 2920
GUF219   005  1597  2924    2920*
GUF222   005  15A0  2927    2911
GUF225   004  15BA  2934    2928
GUF228   003  15C2  2940    2906  2908  2941  2943  2944* 3068* 3086
GUF231   004  15D2  2948    2940  2964  2966
GUF234   006  15DD  2951    2947  2949  2959
GUF237   003  160F  2963    2957  2976  3020
GUF240   005  1639  2974    2970
GUF243   004  1645  2977    2972
GUF246   004  1649  2978    2968  3058
GUF249   005  1672  2988    2981  3041
GUF252   003  16AD  3004    3001
GUF255   003  16B0  3005    2996* 2999  3003
GUF258   005  16B6  3007    2990
GUF261   004  16BE  3009    2992
GUF264   004  16C2  3010    3008
GUF267   005  16CA  3012    3006
GUF270   003  16DE  3017    2993* 3011*
GUF273   004  16E1  3018    3014* 3015*
GUF276   003  16FF  3025    2983
GUF279   003  170B  3029    2987
GUF282   004  1711  3031    3026
GUF285   003  1718  3033    3030
GUF288   004  171B  3034    3028
GUF291   005  171F  3035    3032
GUF294   006  173F  3046    2979  3024  3278
GUF297   006  1760  3053    3051
GUF300   006  177B  3059    3056
GUF303   006  1781  3060    3059*
GUF306   004  1791  3067    2952
GUF309   004  17A0  3071    3097  3101
GUF312   004  17D4  3082    3077
GUF315   004  17E2  3085    3081
GUF318   005  17EA  3091    3070
GUF321   006  17F5  3093    3092*
GUF324   006  181A  3105    3095  3099
GUF327   003  1838  3112    2110*
GUF330   006  1841  3114    2099  2108  3112
GUF333   002  1856  3118    3115
GUF336   006  1857  3119    3110
GUF339   006  1888  3128    3169
GUF342   006  188E  3129    3108  3240
GUF345   005  189E  3132    3130* 3131* 3135* 3136* 3137
GUF348   004  18BC  3138    3134
GUF351   004  18D4  3144    3141
GUF354   004  18D8  3145    2895* 3143
GUF357   004  18DC  3149    3127
```

```
GUF360  006  18EF 3153   3166
GUF363  006  1907 3158   3154
GUF366  004  1915 3161   3156* 3157  3160*
GUF369  006  192C 3167   3163
GUF372  003  1932 3168   3124* 3152* 3167*
GUF375  004  1939 3173   3107
GUF378  004  1984 3194   3199
GUF381  004  199B 3204   3196
GUF384  006  19AB 3208   3222
GUF387  005  19C4 3213   3206* 3207* 3209  3210* 3211* 3212* 3214  3215
GUF390  005  19D5 3216   3214* 3215* 3217* 3218  3228
GUF393  004  19F1 3227   3219
GUF396  004  19F9 3229   3234
GUF399  006  1A10 3238   3187  3231
GUF402  001  000F 3242   2612
GUF405  001  0023 3243   3078
GUF408  001  000C 3244   3096
GUF411  001  0008 3245   2359* 2391* 3387*
GUF414  001  00FF 3246   2991  3057  3096  3160  3387
GUF417  002  1A1F 3247   2451  2491  3100  3173  3174
GUF420  001  1A20 3248   3159* 3162  3165*
GUF423  002  1A22 3251   3164
GUF426  001  0010 3252
GUF429  001  1A23 3253   2910  2934* 2948  2950* 2956  2960* 2967  2977* 3069  3415* 3426*
GUF432  002  1A25 3254   2897* 2953* 2954  3055  3079* 3083*
GUF435  002  1A27 3255   2898* 3053* 3054  3055
GUF438  001  1A28 3256   2464* 2520* 2546  2551* 2556* 2708* 2951* 3080* 3084* 3129* 3390*
GUF441  001  1A29 3259   2912  2914* 3009* 3019  3039*
GUF444  001  1A2A 3262   2984* 2985* 2986  3071* 3120*
GUF447  001  1A2B 3263   2488* 2489  2539* 2540  3036* 3038* 3072* 3073  3073* 3074  3074* 3075
                         3121* 3122  3122* 3123  3123* 3124  3125* 3126  3151  3180* 3183  3189*
                         3204
GUF450  001  1A2C 3264   3031* 3034* 3158*
GUF453  001  1A2D 3265   2683* 2684  2697* 2698  3022  3040* 3150* 3151* 3153  3155* 3156  3175
                         3175* 3176* 3177* 3178  3179  3181  3181* 3182  3182* 3185  3238
GUF456  002  1A2F 3268   3184* 3185* 3186  3191  3205
GUF459  001  1A30 3269   3179* 3188  3195*
GUF462  002  1A32 3270   3190* 3191* 3208  3211  3212  3218*
GUF465  001  1A33 3271   3188* 3230*
GUF468  002  1A35 3272   2118  2377* 2468* 2469* 2899  2904* 2905  2909* 2955* 2961* 2965  2974*
                         2988* 3023* 3061  3404* 3405
GUF471  002  1A37 3275   2124  2376* 2470* 2901  2904  3417* 3477  3477*
GUF474  006  1742 3278   2902* 3047* 3059  3076  3092
GUF477  001  1A38 3279
GUF480  001  1A39 3280
GUF483  001  1A3A 3281   2115* 2916* 3007* 3403*
GUF486  001  1A3B 3282   2102  2120  2374* 2375* 2376  2471* 2929* 2978  2980  2986  2991  3012*
                         3037  3046  3050  3057* 3060* 3061* 3093  3388* 3416*
GUF489  016  1A4B 3283   2359* 2391* 3096  3100  3284  3387*
GUF492  002  1A4D 3284   2671  2902  3076  3131
GUF495  002  1A4F 3285   3208
GUF498  002  1A51 3286   2477* 2917* 2995  3010*
GUF501  001  1A52 3287   2465* 2704* 2713* 3072  3106  3121  3176  3478*
GUF504  001  1A53 3290   2903* 3048* 3094  3105* 3106  3129  3133* 3177  3180
GUF507  001  0002 3293   2115  2399  2451  2491  2916  2927  2971  3007  3371  3378  3399  3403
                         3435
```

```
GUF510  002  1A55 3294    2109  2371  2537  2551  2556  2558  2688  2691  2695  2708  2930  2951
                         2973  3048  3105  3113  3119  3133  3140  3155  3158  3167  3195  3230
                         3423
GUF513  002  1A57 3295    2117  2468
GUF516  002  1A59 3296    2400
GUF519  002  1A5B 3297    2093
GUF522  001  00BD 3298    3109
GUF525  002  1A5D 3299    2117* 2118* 2119  2360  2374  2385  2389
GUF528  002  1A5F 3300    2119* 2120* 2123  2356  2358  2359
GUF531  002  1A61 3301    2384* 2385* 2390* 2391  2392  2394  2395  2399
GUF534  001  0011 3302    3385
GUF537  001  000F 3303    3477
GUF540  001  0004 3304    2218  2219  2398  2455  2471  2493  2495  2538  2689  2696  2707  2993
                         3022  3046  3052  3080  3091  3093  3130  3198  3232  3384  3388  3389
                         3389* 3390  3392
GUF543  001  0032 3305    3082  3391
GUF546  001  0080 3306    2475  2907  2963  3049  3395
GUF549  001  0002 3307    2969  2982  2989  3000  3029  3034  3397
GUF552  001  00FB 3308    2905  2965  3405
GUF555  001  0001 3309    2958  2998  3027  3413
GUF558  001  0013 3310    2466
GUF561  001  1A62 3311    2367  2379*
GUF564  001  0002 3314    2948  3415
GUF567  001  0001 3315    2950  2956  2960
GUF570  001  0002 3316    3036  3417
GUF573  001  0060 3317    3424  3433  3467
GUF576  001  0004 3318    2910  2934  2967  3426
GUF579  001  0001 3319    2401
GUF582  001  0001 3320    3019  3039  3467
GUF585  001  0003 3321    2263  2263* 2372  2533  2686  2927  3050  3084  3132  3469  3476  3476*
GUF588  001  0005 3322    2263  2263  3476
GUF591  001  0007 3323    2453  2978
GUF594  001  0003 3324    2464
GUF597  001  0014 3325    2501
GUF600  001  0004 3326    2520
GUF603  001  0001 3327    2535  2546
GUF606  001  0020 3328    2672
GUF609  001  0010 3329    2675
GUF612  001  00F0 3330    2680
GUF615  001  00FF 3331    2355  2365  2393  2918  3014  3210
GUF618  001  0002 3332    2930  2973  3474
GUF621  001  0003 3333    3031  3033
GUF624  001  0001 3334    2475  2907  2963  3049* 3395
GUF627  001  0002 3335    2399* 2478  2909  2919  2921  2922  2929  2931  2961  2962  2974  2975
                         2980  2984  2988  2994* 2996  2997  2997* 3012  3013  3015  3016  3035*
                         3036  3037* 3038  3404  3407  3416  3417
GUF630  001  0003 3336    2958  2969  2982  2989  2998  3000  3002  3002* 3004* 3025  3027* 3029
                         3033* 3397  3413
GUF633  001  0004 3337    2398* 2400* 3022* 3052* 3091*
GUF636  001  0044 3338    3206  3207  3220  3221
GUF639  001  0006 3339    2916  2927  2971  3007  3399  3403  3435
GUF642  002  1A64 3340    2531  2670  2896  3079
GUF645  004  1A68 3341    2215  2372  2398  2707  3052  3091  3389
GUF648  002  1A6A 3342    2357  3083
GUF651  001  0001 3343    2493  2493  2505  2505  2507  2508  2508* 2509  2511  2511* 2512* 2515*
                         2517  2518  2518* 2519* 2534  2536  2542  2545  2687  2694  2700  3098
```

```
                            3152   3168* 3194   3229*
GUF654  001  0003 3344    2451   2491  3371   3378
GUF657  001  0004 3345    2453   2469  2470   2471   2689   3132* 3161* 3388
GUF660  002  1A6C 3346    2363   2387
GUF663  002  1A6E 3347    2448   2485  3375
GUF666  002  1A70 3348    2100
GUF669  002  1A72 3349    2540   3369
GUF672  002  1A74 3350    2489
GUF675  002  1A76 3351    2390   2678  2679   2690   2706   2985   2994   3023   3035   3047   3128   3135
                            3136   3373
GUF678  002  1A78 3352    2500   3021  3377
GUF681  002  1A7A 3353    3466   3475*
GUF684  001  0C07 2420    2474   3340  3394
GUF687  001  0C08 2421
GUF690  001  0D07 2424    2386   3342  3389*
GUF693  001  0D08 2425
GUF696  001  0E07 2428
GUF699  001  0E08 2429
GUF702  001  0F07 2432
GUF705  001  0F08 2433
GUF708  256  1106 2436    2923*  2924  3193   3213*  3216   3227
GUF711  004  0D90 2136    1982
GUF713  004  0DAF 2145    2138
GUF714  004  0DBE 2149    2146
GUF715  004  0DE6 2161    2155   2157
GUF717  004  0DF3 2165    2150   2160  2162
GUF720  004  0DFD 2168
GUF723  004  0E14 2175    2171
GUF726  004  0E18 2176    2174
GUF729  004  0E1C 2177    2169
GUF732  004  0E28 2188    2141   2144
GUF735  004  0E2F 2190
GUF738  004  0E40 2197    2191
GUF741  004  0E4E 2201    2189   2198
GUF744  004  0E55 2203    2200
GUF747  004  0E7A 2213    2206
GUF750  006  0E7E 2214    2210
GUF753  006  0E84 2215    2212
GUF756  004  0E8A 2216
GUF759  004  0EBE 2230    2222
GUF762  004  0ECA 2236    2202   2208  2220   2224
GUF765  004  0EE3 2242    2237
GUF768  004  0EF1 2246    2243
GUF771  004  0F01 2250    2228   2245
GUF774  004  0F21 2259    2251
GUF777  006  0F2F 2263    2204
GUF780  004  0F35 2264    2253   2260  2262
GUF786  001  0F3F 2270    2166
GUF789  001  0F41 2272    2164
GUF790  001  0F47 2277    2159
GUF810  001  0F4D 2282    2194
GUF813  002  0F54 2287    2219
GUF816  001  0F55 2288    2152
GUF819  001  0F5B 2293
GUF822  006  0D0C 2093    3432
GUF825  006  0D1D 2100    2094
```

```
GUF828  004  0D33 2108
GUF831  006  0D37 2109    2101  2103
GUF834  006  0D41 2115    3419
GUF837  004  1007 2355    2125
GUF840  006  1023 2360    2380
GUF843  004  102C 2362    2402
GUF846  006  104D 2369    2122* 2123* 2124* 2355* 2356* 2357* 2358* 2370  2392*
GUF849  006  105F 2372    2370* 2371*
GUF852  005  1065 2373    2363* 2364* 2365* 2366* 2368
GUF855  004  1086 2379    2121
GUF858  006  108E 2384    2361
GUF861  005  10D2 2396    2387* 2388* 2393* 2394* 2395* 2397
GUF864  005  10DD 2398    2397*
GUF867  001  1A7B 3361    2266  3479
GUF870  004  1A83 3368    3362  3363  3374  3380
GUF873  006  1A9F 3375    3372
GUF876  003  1AB7 3384    3379
GUF879  004  1ABA 3385    3376
GUF882  003  1AE4 3395    3408
GUF885  005  1B01 3404    3398
GUF888  003  1B14 3413    3400
GUF891  005  1B1E 3416    3414  3436
GUF894  004  1B3D 3426
GUF897  004  1B44 3431    3401
GUF900  004  1B5E 3442    3370  3396  3406
GUF903  004  1B6A 3450    2378  3425  3427
GUF906  004  1B6E 3456    2225* 3434  3437
GUF909  003  1B79 3459    2244*
GUF912  004  1B7C 3460    2257  2624  2627
GUF915  004  1B8E 3465    3459
GUF918  004  1B92 3466    3458
GUF921  003  1B9F 3470    3468
GUF924  001  1C01 3481    2102  2360  2362  2364  2366  2375  2377  2384  2388  2389* 2401* 3346
                         3482
GUF927  001  1C05 3482    2115  2263  2263* 2927  3114  3355  3371  3378  3399  3476* 3483  3484
GUF930  001  1C08 3483    2971  3435
GUF933  001  1C05 3484    3424  3433
GUF936  001  1D00 3485    3109  3119* 3178* 3347  3348  3349  3350  3486  3487  3488
GUF939  001  1D02 3486    2093  2109* 2930* 2973* 3113* 3140  3423*
GUF942  001  1D09 3487    2100  2462* 3067  3125  3239  3364  3368* 3369  3375
GUF945  001  1D0B 3488    2684  2698  3126  3128* 3149  3186  3190  3238*
GUGENT  001  1107 2441    2258  2481  2633  2644  3465
```

TOTAL STATEMENTS IN ERROR IN THIS ASSEMBLY =   0

OL105 I   THE CODE LENGTH OF #GUFUD IS 7168 DECIMAL.
OL103 I   TOTAL NUMBER OF LIBRARY SECTORS REQUIRED IS   21
          NAME-#GUFUD,PACK-R1R1R1,UNIT-R1,RETAIN-P,LIBRARY-R,CATEGORY-000

```
        OVERLAY LINKAGE EDITOR CORE USAGE MAP     29/02/16


   START      CATEGORY   NAME AND        CODE LENGTH
  ADDRESS                 ENTRY      HEXADECIMAL DECIMAL

  0C00          0       #GUFUD          1C00     7168


 OL100 I   THE TOTAL CORE USED BY #GUFUD IS   7168 DECIMAL.
 OL101 I   THE START CONTROL ADDRESS OF THIS MODULE IS 0C00.
 OL104 I   TOTAL NUMBER OF LIBRARY SECTORS REQUIRED IS    29
          NAME-#GUFUD,PACK-R1R1R1,UNIT-R1,RETAIN-P,LIBRARY-O
```