```
 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  03/02/22  PAGE   1
#KSAVE   MODULE
```

```
ERR LOC  OBJECT CODE      ADDR STMT SOURCE STATEMENT          VER 15, MOD 00  03/02/22  PAGE   2
    0000                          1 #KSAVE START 0
                                  2        PRINT ON,NODATA
                                  3 *      @SYS  EXP-N
                                214+       PRINT ON
                                215 *      @FXD  EXP-N
                                620+       PRINT ON
                                621 *      @CAN  EXP-N
                                724+       PRINT ON
                                725 *      @WKA  EXP-N
                                795+       PRINT ON
                                796 *      @DIR  EXP-N
                                916+       PRINT ON
                                917 *      @SPF  EXP-N
                               1380+       PRINT ON
                               1381 *      @ERM  EXP-N
                               2003+       PRINT ON
                               2005 ***********************************************************************
                               2006 *  5703-XM1 COPYRIGHT IBM CORP. 1970                                  *
                               2007 *           REFER TO INSTRUCTIONS ON COPY RIGHT NOTICE, 120-2083      *
                               2008 *                                                                     *
                               2009 ***********************************************************************
                               2010 *STATUS                                                               *
                               2011 *   VERSION 1 MODIFICATION 0                                          *
                               2012 *                                                                     *
                               2013 *FUNCTION                                                             *
                               2014 *   * #KSAVE WILL SAVE THE WORK FILE IN A USER FILE LIBRARY OR A **   *
                               2015 *     FILE LIBRARY.                                                   *
                               2016 *   * IF THE FILE SPECIFICATION IS NOT SPECIFIED THE NAME OF THE WORK *
                               2017 *     FILE IS USED WITH THE CURRENT PASSWORD AND DISK.                *
                               2018 *   * IF THE FILE SPECIFICATION IS PRESENT IT DEFINES THE FILENAME    *
                               2019 *     (AND POSSIBLY DISK AND/OR PASSWORD) TO BE USED IN SAVING THE    *
                               2020 *     WORK FILE.                                                      *
                               2021 *   * IF A TWO-STAR FILENAME IS SPECIFIED WITHOUT THE OPTIONAL DISK-  *
                               2022 *     LABEL, ALL TWO-STAR LIBRARIES ON THE DISK ARE SEARCHED FOR THE  *
                               2023 *     SPECIFIED NAME.  THE SEARCH ORDER OF TWO-STAR LIBRARIES IS AS   *
                               2024 *     FOLLOWS: F1, F2, R1, R2.  IF THE FILE NAME IS NOT FOUND ON      *
                               2025 *     ANY DISK THE ORDER IS RE-SEARCHED TO SAVE THE FILE IN THE FIRST *
                               2026 *     AVAILABLE SPACE.                                                *
                               2027 *   * IF A FILE OF THE SAME NAME IS FOUND IT IS DESTROYED BY THE NEW  *
                               2028 *     FILE UNLESS IT HAS BEEN POOLED OR PROTECTED.                    *
                               2029 *   * THE NULL DIRECTORY OF THE LIBRARY IS SEARCHED FOR A SPACE TO    *
                               2030 *     SAVE THE FILE.  IF A SPACE IS FOUND AND THE FILENAME IS ALSO    *
                               2031 *     FOUND, THE FILE WILL BE SAVED IN THE SPACE WHICH IS CLOSEST TO  *
                               2032 *     THE BEGINING OF THE LIBRARY.  IF THE OLD SPACE IS CLOSER AN     *
                               2033 *     ADDITIONAL CHECK MUST ALSO BE MADE TO INSURE IT IS LARGE ENOUGH.*
                               2034 *     IF THE NEW SPACE IS USED THE OLD SPACE IS RETURNED TO THE NULL  *
                               2035 *     DIRECTORY.                                                      *
                               2036 *   * WHEN THERE IS NO SPACE LARGE ENOUGH TO SAVE THE FILE A TEST IS  *
                               2037 *     MADE TO DETERMINE IF THE LIBRARY SHOULD BE PACKED. IF THE TOTAL *
                               2038 *     OF ALL NULL SPACE WILL PROVIDE A LARGE ENOUGH SPACE, SPACKU     *
                               2039 *     IS CALLED TO PACK THE AREA. IF THE TOTAL SPACE IS NOT ENOUGH TO *
                               2040 *     SAVE THE FILE THE COMMAND IS ABORTED.                           *
                               2041 *   * AN OPTIONAL CHARACTER CONSTANT OF 25 CHARACTERS IS ACCEPTED AS  *
                               2042 *     ADDITIONAL INFORMATION. THIS HEADER IS SAVED WITH THE FILE AND  *
                               2043 *     IS PRINTED WHEN THE FILE IS EDITED OR DURING A LISTCAT COMMAND. *
                               2044 *                                                                     *
                               2045 *ENTRY POINTS                                                         *
                               2046 *   * KSAVE - #KSAVE IS ENTERED AT #KSAVE BY THE COMMAND ANALYSER     *
                               2047 *                                                                     *
                               2048 *INPUT                                                                *
                               2049 *   * INPUT TO SAVE IS IN THE FORM OF PARAMETERS IN THE INPUT LINE    *
                               2050 *     BUFFER.                                                         *
                               2051 *   * IF THE DEFAULT OPTION IS USED THE PASSWORD AND VOLUME OF THE    *
                               2052 *     CURRENT LOGGED ON USER IS USED IN ADDITION TO THE CURRENT NAME  *
                               2053 *     OF THE WORK FILE.                                               *
                               2054 *                                                                     *
                               2055 *OUTPUT                                                               *
                               2056 *   OUTPUT FROM SAVE IS IN THE FORM OF THE FILE SAVED IN THE ** OR    *
                               2057 *   USER LIBRARY.                                                     *
                               2058 *                                                                     *
                               2059 *EXTERNAL REFERENCES                                                  *
                               2060 *      $XRSAV - ADDRESS OF 2 BYTE XR SAVE AREA.                       *
                               2061 *      $INDR1 - ADDRESS OF WORK FILE STATUS INDICATOR BYTE.            *
                               2062 *      $CAERR - ADDRESS OF SAVED ERROR CODE.                           *
                               2063 *      $WFNME - ADDRESS OF WORK FILE NAME.                             *
                               2064 *      $CAERK - ADDRESS OF ENTRY TO ERROR ROUTINE.                     *
                               2065 *      $CIMSK - ADDRESS OF INQUIRY REQUEST INDICATOR.                  *
                               2066 *      $INDR3 - ADDRESS OR SYSTEM STATUS INDICATOR BYTE.               *
                               2067 *      $EXFTR - ADDRESS OF CORE EXPANSION FACTOR.                      *
                               2068 *      $$FLIB - ADDRESS OF THE FILE LIBRARY ADDRESS PASS AREA          *
                               2069 *      $DISKN - ADDRESS OF ENTRY TO SYSTEM DISK IOCS.                  *
                               2070 *      $DATE  - ADDRESS OF THE CURRENT DATE.                           *
                               2071 *      $$FITS - ADDRESS OF THE FILE INDEX TABLE.                       *
                               2072 *      $CARPL - ADDRESS OF ENTRY TO ABORT CURRENT OP AND ENABLE IR.    *
                               2073 *      $DPLSV - ADDRESS OF 6 BYTE DPL SAVE AREA.                       *
                               2074 *      $CANIT - ENTRY TO THE DELIMITER SCAN SUBROUTINE.                *
                               2075 *      DL2ICS - ENTRY TO THE 2 SURFACE DISK IOCS.                      *
                               2076 *      DL4ICS - ENTRY TC THE 4 SURFACE DISK IOCS.                      *
                               2077 *      SRCHFN - ENTRY TO THE SEARCH FILENAME SUBROUTINE.               *
                               2078 *      STORIN - ENTRY TO THE STORE IN NULL DIRCTY SUBROUTINE.          *
                               2079 *      STUFID - ENTRY TO THE STORE IN USER DIRCTY BLOCK SUBROUTINE.    *
                               2080 *      SFINDF - ENTRY TO FILE SEARCH CONTROL SUBROUTINE.               *
                               2081 *      SUFFER - ENTRY TO SYNTAX CHECK OF FILE SPEC. SUBROUTINE.        *
                               2082 *      SCSTRG - ENTRY TO CHARACTER STRING ANALYSER SUBROUTINE.         *
                               2083 *      TSMLES - DATA MANAGEMENT COMMON SAVE AREA.                      *
                               2084 *                                                                      *
                               2085 *EXITS, NORMAL                                                         *
                               2086 *   * NORMAL EXIT IS TO $CARPL.  ON A NORMAL EXIT THE FILE HAS BEEN    *
                               2087 *     SAVED IN THE SPECIFIED LIBRARY WITH THE DIRECTORIES UPDATED      *
                               2088 *     AS REQUIRED.                                                     *
                               2089 *   * IF THE LIBRARY MUST BE PACKED TO SAVE THE FILE.  SAVE WILL EXIT  *
                               2090 *     TO SPACKU TO PACK THE LIBRARY.  SPACKU WILL RELOAD SAVE TO SAVE  *
                               2091 *     THE FILE AND THEN TAKE THE NORMAL EXIT.                          *
                               2092 *                                                                      *
                               2093 *EXITS, ERROR                                                          *
                               2094 *   * THE ERROR EXIT FROM #KSAVE IS TO $CAERK.  THE CAUSE OF THE       *
                               2095 *     ERROR IS INDICATED BY THE ERROR CODE PLACED IN $CAERR.           *
                               2096 *   * IF THE ERROR IS A COMMAND SYNTAX ERROR THE ADDRESS IN @XR        *
                               2097 *     POINTS TO THE ERROR.  IF THE ERROR IS NOT A SYNTAX ERROR @XR     *
                               2098 *     IS CLEARED TO DELETE THE UP ARROW.                               *
                               2099 *   * THE FOLLOWING CONDITIONS WILL CAUSE THE COMMAND TO IT ABORTED;   *
                               2100 *      * A FILE OF THE SAME NAME IS IN THE USER LIBRARY AND IS         *
                               2101 *        POOLED OR PROTECT.                                            *
                               2102 *      * NOT ENOUGH SPACE HAS BEEN FOUND TO SAVE THE FILE.             *
                               2103 *      * SYSTEM COMMAND ERROR.                                         *
                               2104 *                                                                      *
                               2105 *TABLES/WORKAREAS                                                      *
                               2106 *   * THE CONSTANT AREA RESIDES AT THE FRONT OF THE PROGRAM AND IS     *
                               2107 *     REFERENCED BY A DISPLACEMENT RELATIVE TO THE VALUE IN THE BASE   *
                               2108 *     REGISTER @BR.                                                    *
                               2109 *   * THE WORKAREA FOR #KSAVE IS ORGED TO THE START OF #KSAVE AND      *
                               2110 *     OVERLAYS THE FIRST PORTION OF EXECUTABLE CODE.                   *
                               2111 *                                                                      *
                               2112 *ATTRIBUTES                                                            *
                               2113 *   RELOCATABLE                                                        *
                               2114 *                                                                      *
                               2115 *CHARACTER CODE DEPENDENCY                                             *
                               2116 *   THE OPERATION OF THIS MODULE DEPENDS UPON AS INTERNAL              *
                               2117 *   REPRESENTATION OF THE EXTERNAL CHARACTER SET WHICH IS EQUIVALENT  *
                               2118 *   TO THE ONE USED AT ASSEMBLY TIME.  THE CODING HAS BEEN ARRANGED   *
                               2119 *   SO THAT REDEFINITION OF CHARACTER CONSTANTS, BY REASSEMBLY, WILL  *
                               2120 *   RESULT IN A CORRECT MODULE FOR THE NEW DEFINITIONS.               *
                               2121 *                                                                     *
                               2122 *NOTES                                                                *
                               2123 *   ERROR PROCEDURES                                                  *
                               2124 *      IF AN ERROR IS DETECTED THE APPROPRIATE ERROR CODE IS PLACED   *
                               2125 *      IN $CAERR, AND A BRANCH IS TAKEN TO $CAERK.                    *
                               2126 *                                                                     *
                               2127 *   REGISTER USAGE                                                    *
                               2128 *      @BR IS USED AS THE BASE REGISTER TO REFERENCE THE CONSTANTS    *
                               2129 *      AND WORK AREA.  @XR IS USE AS A GENERAL WORK REGISTER DURING   *
                               2130 *      PROGRAM CLIECUTION.                                            *
                               2131 *                                                                     *
                               2132 *   SAVED/RESTORED AREAS                                              *
                               2133 *      N/A                                                            *
                               2134 *                                                                     *
                               2135 *   MODIFICATION CONSIDERATIONS                                       *
                               2136 *      TO ALLOW THE SAVE KEYWORD TO FUNCTION IN AN 8K SYSTEM, DL4ICS  *
                               2137 *      AND SURCHN ARE LOADED IN DURING EXECUTION.  THE LOCATION OF    *
                               2138 *      THESE ROUTINES AND THE LOCATION OF TSMLES MUST BE ESTABLISHED  *
                               2139 *      WITH EQUATES AND FIXED ORIGIN.  IF TSMLES, DL4ICS. OR SURCHN   *
                               2140 *      ARE MOVED IN #KSAVE OR #KSVLA TO NEW LOCATION MUST BE          *
                               2141 *      SPECIFIED IN THE OTHER MODULE.                                 *
                               2142 *                                                                     *
                               2143 *   REQUIRED MODULES                                                  *
                               2144 *      @SYSEQ - SYSTEM SOFTWARE EQUATES.                              *
                               2145 *      @FXDEQ - FIXED ADDRESSES FOR SYSTEM NUCLEUS.                   *
                               2146 *      @CANEQ - SYSTEM LOCATION EQUATES.                              *
                               2147 *      @WKAEQ - SYSTEM WORK AREA EQUATES.                             *
                               2148 *      @DIREQ - FILE LIBRARY AREA DIRECTORY EQUATES.                  *
                               2149 *      SPFEQ  - SYSTEM PROGRAM FILE EQUATE MACRO.                     *
                               2150 *      ERMEQ  - ERROR MESSAGE EQUATE MACRO.                           *
                               2151 *      DL2ICS - 2 SURFACE DISK IOCS.                                  *
                               2152 *      DL4ICS - 4 SURFACE DISK IOCS.                                  *
                               2153 *      SURCHN - SEARCH NULL DIRECTORY SUBROUTINE.                     *
                               2154 *      SRCHFN - SEARCH USER DIRECTORY SUBROUTINE.                     *
                               2155 *      STORIN - STORE IN NULL DIRECTORY SUBROUTINE.                   *
                               2156 *      STUFID - STORE IN USER DIRECTORY BLOCK SUBROUTINE.             *
                               2157 *      SFINDF - FILE SEARCH SUBROUTINE.                               *
                               2158 *      SVOLID - SEARCH VOLID TABLE SUBROUTINE.                        *
                               2159 *      SUFFER - SYNTAX CHECK FILE SPEC SUBROUTINE.                    *
                               2160 *      SALPHA - ALPHAMERIC CHARACTER CHECKER SUBROUTINE.              *
                               2161 *      SGETDB - GET USER DIRECTORY BLOCK SUBROUTINE.                  *
                               2162 *      TSMLES - DATA MANAGEMENT COMMON SAVE AREA.                     *
                               2163 *      SCSTRG - CHARACTER STRING ANALYSER.                            *
                               2164 *      SCANIT - SCAN DELIMITER SUBROUTINE.                            *
                               2165 *                                                                     *
                               2166 *   OTHER                                                             *
                               2167 *      NONE                                                           *
                               2168 ***********************************************************************
```

```
                              2170 *       HDR   #KSAVE                    PROGRAM NAME
                              2171 ************************************************************************
                              2172 * PROGRAM HEADER FOR DISK LOAD                                         *
                              2173 ************************************************************************
                              2174 *#$KSAV EQU   X'0488'                   DISK ADDR OF #KSAVE
                              2175 *#$$KSA EQU   X'0C00'                   CORE LOAD ADDRESS OF #KSAVE
                              2176 *#$@KSA EQU   017                       SECTOR CNT OF #KSAVE
    0C00                      2177        ORG   #$$KSA                    CORE LOAD ADDRESS
                          0C00 2178 $$$$$$ EQU   *                         FIRST LOCATION IN PROGRAM
    0C00 7BD2E2C1E5C5     0C05 2179        DC    CL6'#KSAVE'               PROGRAM NAME
    0C06 1F               0C06 2180        DC    IL1'031'                  PROGRAM NUMBER OF #KSAVE
                          0C07 2181 $KSAVE EQU   *                         ENTRY POINT TO PROGRAM
                              2182 *** END OF EXPANSION ***

                          0C66 2184        USING KSAVEN,@BR
                          0001 2185 KSAE01 EQU   1
                              2186 *
                              2187 *            SAVE KEYWORD EQUATES
                              2188 *
                          0002 2189 KSAE02 EQU   2                         VALUE FOR LENGTH CODE
                          000C 2190 KSAFFD EQU   @FDE1                     DISP TO FIRST FIT ENTRY
                          0004 2191 KSAFEL EQU   @FLENT                    FIT ENTRY LENGTH
                          000C 2192 KSAMXT EQU   12                        BUFFER SECTOR COUNT
                          0700 2193 KSANBF EQU   $$KLD2                    NULL BUFFER CORE ADDR
                          0802 2194 KSARDT EQU   $$KLD2+258                FIRST ENTRY OF READ TABLE
                          0002 2195 KSAERL EQU   2                         READ TBL ENTRY LENGTH
                          001C 2196 KSABTS EQU   ##MUEX+##MUER+##MUEO      POOLED, PROTECTED, OPEN
                          0004 2197 KSAFLG EQU   @FLENT                    FIT ENTRY LENGTH
                          0987 2198 DL4ICS EQU   #$$KSV+@HDRLN             SET UP ENTRY ADDR TO OVERLAY
                          0A50 2199 SURCHN EQU   X'0A50'                   ENTRY TO SEARCH NULL SUBROUTINE
                              2200 *
    0C07 F2 87 4A             2201        J     KSAVE                     JUMP OVER MESSAGE
                              2202 *
                              2203 *       MTEXT @@M300=@PRETR,PATCH=015
                              2204 ************************************************************************
                              2205 * PPL'S AND TEXT FOR MESSAGE                                           *
                              2206 ************************************************************************
    0C0A C0               0C0A 2207 @@M300 DC    AL1(@PRETR)               PRINT CONTROL FUNCTION
    0C0B 37               0C0B 2208        DC    IL1'55'                   LENGTH OF MESSAGE
    0C0C 0C0E             0C0D 2209        DC    AL(@CADDR)(@@T300)        ADDR OF MESSAGE
                              2210 *
                          0C0E 2211 @@T300 EQU   *                         LEFT BYTE OF MESSAGE
    0C0E C5D9D9D6D940F5F8 0C3F 2212        DC    CL050'ERROR 580 DUPLICATE DISK LABELS - SPECIFY DISK LOC'
    0C40 C1E3C9D6D5       0C44 2213        DC    CL005'ATION'
                              2214 *
                              2215 *  PATCH AREA FOR MESSAGES
                              2216 *
    0C45                  0C53 2217 $$$001 DS    CL015                     MSG EXPANSION PATCH AREA
                              2218 *** END OF EXPANSION ***

    0C54 C2 01 0C66           2220 KSAVE  LA    KSAVEN,@BR                SET UP BASE REGISTER POINTER
    0C58 35 02 03C7           2221        L     $XRSAV,@XR                POINTER TO INPUT LINE
    0C5C 3C 00 1B5F           2222        MVI   SMNSCT-1,@ZERO            CLEAR LEFT BYTE SECTOR COUNT
                              2223 *
                              2224 *            PICK UP DB COUNT FOR NULL SEARCH FROM FIT BEFORE CLOBBERED
                              2225 *            AND SYNTAX CHECK THE COMMAND.
```

```
ERR LOC  OBJECT CODE      ADDR STMT SOURCE STATEMENT          VER 15, MOD 00  03/02/22  PAGE   7
                               2226 *
    0C60 0C 00 1B60 1D00       2227        MVC   SMNSCT,$$FITS+@FDSD(1)    DB COUNT FROM FIT
                          0C66 2228 KSAVEN EQU   *                         ENTRY FROM SYSTEM
    0C66 C0 87 1BE8            2229        B     SCANIT                    SCAN PARAMETER LIST
    0C6A D0 82 94              2230        BL    KSA025(,@BR)              SCANIT ERROR RETURN PSR LOW
    0C6D 3C 18 03CD            2231        MVI   $CAERR,@@E139             INVALID DELIMITER
    0C71 BD 60 00              2232        CLI   @ZERO(,@XR),@MINUS        TEST IF DELIMITER A DASH
    0C74 F2 81 83              2233        JE    KSA025                    JUMP IF ERROR
    0C77 3C 01 1C05            2234        MVI   SCAMMA,SCACOM             SKIP OVER A COMMA
    0C7B 3C 19 1BBE            2235        MVI   SCSLNG,##LUEH             LENGTH OF FILE HEADER
    0C7F 0C 07 1B54 0443       2236        MVC   SMFNAM(##LUEN),$WFNME     WORK AREA NAME
    0C85 7C 40 18              2237        MVI   KSAHED(,@BR),@BLANK       SET UP TO BLANK FILE HEADER
    0C88 5C 17 17 18           2238        MVC   KSAHED-1(##LUEH-1,@BR),KSAHED(,@BR)  BLANK REST OF FIELD
    0C8C BD 1E 00              2239        CLI   @ZERO(,@XR),@EOS          TEST IF NO FILE SPECIFICATION ?
    0C8F F2 81 B4              2240        JE    KSA005                    JUMP IF NO SPECIFICATION
                               2241 *
    0C92 C0 87 1B6D            2242        B     SCSTRG                    GO LOOK FOR A STRING
    0C96 0C66             0C97 2243        DC    AL2(KSAHDR)               HEADERSAVE AREA
    0C98 D0 04 7A              2244        BNH   KSA006(,@BR)              FOUND A HEADER IF LOW/EQUAL
    0C9B 3D 00 1BE2            2245        CLI   SCSCNT,@ZERO              TEST IF ERROR OR NO HEADER ?
    0C9F D0 01 94              2246        BNE   KSA025(,@BR)              COUNT NOT ZERO IS ERROR
    0CA2 7C 80 51              2247        MVI   KSA002+@Q(,@BR),@NOP      SWITCH TO ALLOW HEADER SEARCH
    0CA5 C0 87 1876            2248 KSA001 B     SUFFER                    PROCESS FILE-SPEC
    0CA9 D0 82 94              2249        BL    KSA025(,@BR)              ERROR IF LOW
    0CAC BD 1E 00              2250        CLI   @ZERO(,@XR),@EOS          END OF LINE ?
    0CAF D0 81 E0              2251        BE    KSA005(,@BR)              END OF INPUT
    0CB2 3C 11 03CD            2252        MVI   $CAERR,@@E131             INVALID PARAMETER
    0CB6 D0 80 94              2253 KSA002 BC    KSA025(,@BR),@NOP         SWITCH TO TEST FOR HEADER
    0CB9 BD 7D 00              2254        CLI   @ZERO(,@XR),C''''         IS PARAMETER VALID HEADER ?
    0CBC 3C 11 03CD            2255 KSA003 MVI   $CAERR,@@E131             INVALID PARAMETER
    0CC0 D0 01 94              2256        BNE   KSA025(,@BR)              ERROR IF NOT AN EOS OR QUOTE
    0CC3 C0 87 1B6D            2257        B     SCSTRG                    IF NOT EOS TEST FOR CHAR STRING
    0CC7 0C66             0CC8 2258        DC    AL2(KSAHDR)               POINTER TO SAVE AREA
    0CC9 D0 84 94              2259        BH    KSA025(,@BR)              BRANCH IF ERROR
    0CCC 3C 87 0EC8            2260        MVI   KSA156+@Q,@UCB            SKIP OVER HEADER
    0CD0 C0 87 1BE8            2261        B     SCANIT                    MOVE OVER ANY DELIMITERS
    0CD4 D0 82 94              2262        BL    KSA025(,@BR)              BRANCH IF ERROR
    0CD7 BD 1E 00              2263        CLI   @ZERO(,@XR),@EOS          MUST BE THE END
    0CDA D0 01 56              2264        BNE   KSA003(,@BR)              GO TAKE ERROR RETURN
    0CDD F2 87 66              2265        J     KSA005                    GO START GENERAL PROCESSING
                               2266 *
    0CE0 7C 87 51              2267 KSA006 MVI   KSA002+@Q(,@BR),@UCB      HEADER ALREADY FOUND SET SWITCH
    0CE3 3C 87 0EC8            2268        MVI   KSA156+@Q,@UCB            SKIP OVER OLD HEADER
    0CE7 C0 87 1BE8            2269        B     SCANIT                    GO LOOK FOR NON-DELIMITER
    0CEB D0 82 94              2270        BL    KSA025(,@BR)              BRANCH IF ERROR
    0CEE BD 1E 00              2271        CLI   @ZERO(,@XR),@EOS          END OF LINE ?
    0CF1 F2 81 52              2272        JE    KSA005                    END OF INPUT
    0CF4 D0 87 3F              2273        B     KSA001(,@BR)              LOOP...

    0CF7 75 02 B3              2275 KSA024 L     KSAC00(,@BR),@XR          CLEAR TO DROP UP ARROW
    0CFA C0 87 0469            2276 KSA025 B     $CAERK                    ERROR RETURN
                          0CF7 2277 SFIERR EQU   KSA024                    SFINDF ERROR RETURN
                          0CF7 2278 STUERR EQU   KSA024                    STUFID ERROR RETURN
    0CFE 01               0CFE 2280 KSA4PL DC    AL1(@DGET)                DPL TO READ INPUT FROM WORKANEA
    0CFF 0503             0D00 2281        DC    AL2(@WSTBL)               DADDR OF WORK AREA
    0D01                  0D01 2282        DS    CL1                       SECTOR COUNT
    0D02 1301             0D03 2283 KSA4IA DC    AL2(KSAIOB)               CORE ADDR INPUT BUFFER
    0D04 02               0D04 2284 KSA2PL DC    AL1(@DPUT)                DPL TO WRITE IN LIBRARY AREA
    0D05                  0D06 2285 KSA2DA DS    CL(@DADDR)                DISK ADDR OF SPACE FOR FILE
    0D07 0C               0D07 2286        DC    AL1(KSAMXT+*-*)           TOTAL SECTOR COUNT
    0D08 1301             0D09 2287 KSA2IA DC    AL2(KSAIOB)               CADDR OF OUTPUT BUFFER
    0D0A 01               0D0A 2288 KSAFPL DC    AL1(@DGET)                DPL TO READ IN FIT
    0D0B 0500             0D0C 2289 KSAFSZ DC    AL2(@WSFIT)               DADDR OF FIT
    0D0D 03               0D0D 2290 KSAFLN DC    IL1'3'                    SECTOR COUNT FOR INITIAL FIT
    0D0E 164F             0D0F 2291 KSAFBA DC    AL2(SFINDF)               BUFFER ADDR
    0D10 01               0D10 2292 KSALT4 DC    AL1(@DGET)                DPL TO READ THE USER DIRCTY
    0D11                  0D12 2293 KSAL4A DS    CL(@DADDR)                DADDR OF DIRCTY
    0D13 02               0D13 2294        DC    AL1(##LU)                 SECTOR COUNT
    0D14 1B6D             0D15 2295        DC    AL2(SMUDB1)               BUFFER ADDR
    0D16 0002             0D17 2296 KSATBL DC    AL2(KSAERL)               TABLE ENTRY LENGTH VALUE
    0D18 0000             0D19 2297 KSAC00 DC    AL2(@ZERO)                VALUE TO COMPARE COUNTERS
    0D1A 30               0D1A 2298 KSAC48 DC    IL1'48'                   CYL VALUE
    0D1B 03               0D1B 2299 KSAFDB DC    AL1(@WSTBL-@WSFIT)        FIRST DATA BLOCK
    0D1C 0001             0D1D 2300 KSAC01 DC    IL2'01'                   VALUE TO BUMP POINTERS
    0D1E 3D               0D1E 2301 KSAC61 DC    IL1'61'                   COUNT OF ENTRIES FOR FIRST FIT
    0D1F 40               0D1F 2302 KSAC64 DC    IL1'64'                   COUNT OF ENTRIES NEXT BLOCKS
    0D20 1B5D             0D21 2303 KSASMN DC    AL2(SMNDEA-1)             ADDR OF NEW ENTRY
                               2304 *
                          0D22 2305 KSALT3 EQU   *                         DPL TO GET THE I/O SECTOR
    0D22 01               0D22 2306        DC    AL1(@DGET)                READ 0 ?
    0D23 0459             0D24 2307        DC    AL2(#@#IO1)               ADDR OF I/O SECTOR ON DISK
    0D25 01               0D25 2308        DC    AL1(#@@#IO)               SECTOR COUNT
    0D26                  0D27 2309        DS    CL(@CADDR)                VARIABLE BUFFER ADDR OF I/O SEC
    0D26                       2310        ORG   KSALT3+@DBFR1             INITIALIZE THE BUFFER CADDR
    0D26 164F             0D27 2311 KSAL3A DC    AL2(SFINDF)               INITAL ADDR FOR FIT AND I/O
```

```
                          0D28 2313 KSALT2 EQU   *                         DPL TO READ NULL DIRCTY
    0D28 01               0D28 2314        DC    AL1(@DGET)                REQUESTED FUNCTION
    0D29 0000             0D2A 2315        DC    AL2(@ZERO)                DISK ADDRESS
    0D2B 01               0D2B 2316        DC    AL1(##LN)                 SECTOR COUNT
    0D2C 0700             0D2D 2317        DC    AL2($$KLD2)               CADDR OF BUFFER

                               2319 *KSAPLX DPL   FUNC=@DGET,DADDR=#$KSAV,CNT=#$@KSA,CADDR=#$$KSA
                          0D2E 2320 KSAPLX EQU   *                         DISK PARAMETER LIST
    0D2E 01               0D2E 2321        DC    AL1(@DGET)                REQUESTED FUNCTION
    0D2F 0488             0D30 2322        DC    AL2(#$KSAV)               DISK ADDRESS
    0D31 11               0D31 2323        DC    AL1(#$@KSA)               SECTOR COUNT
    0D32 0C00             0D33 2324        DC    AL2(#$$KSA)               BUFFER ADDRESS
                               2325 *** END OF EXPANSION ***
                          0D33 2326 KSASPU EQU   *-1                       RIGHT END OF DPL

                               2328 *KSAPAC DPL   FUND=@DGET,DADDR=#$SPAC,CNT=#$@SPA,CADDR=#$$SPA
                          0D34 2329 KSAPAC EQU   *                         DISK PARAMETER LIST
    0D34 01               0D34 2330        DC    AL1(@DGET)                REQUESTED FUNCTION
    0D35 04CC             0D36 2331        DC    AL2(#$SPAC)               DISK ADDRESS
    0D37 04               0D37 2332        DC    AL1(#$@SPA)               SECTOR COUNT
    0D38 0C00             0D39 2333        DC    AL2(#$$SPA)               BUFFER ADDRESS
                               2334 *** END OF EXPANSION ***

                               2336 *KSAIOR DPL   FUNC=@DGET,DADDR=#$DPRI,CNT=#$@DPR,CADDR=#$$DPR
                          0D3A 2337 KSAIOR EQU   *                         DISK PARAMETER LIST
    0D3A 01               0D3A 2338        DC    AL1(@DGET)                REQUESTED FUNCTION
    0D3B 014C             0D3C 2339        DC    AL2(#$DPRI)               DISK ADDRESS
    0D3D 05               0D3D 2340        DC    AL1(#$@DPR)               SECTOR COUNT
    0D3E 0700             0D3F 2341        DC    AL2(#$$DPR)               BUFFER ADDRESS
                               2342 *** END OF EXPANSION ***

                               2344 *            DPL TO READ IN THE SAVE OVERLAY
                               2345 *KSAOVL DPL   FUNC=@DGET,DADDR=#$KSVL,CNT=#$@KSV,CADDR=#$SKSV
                          0D40 2346 KSAOVL EQU   *                         DISK PARAMETER LIST
    0D40 01               0D40 2347        DC    AL1(@DGET)                REQUESTED FUNCTION
    0D41 058C             0D42 2348        DC    AL2(#$KSVL)               DISK ADDRESS
    0D43 02               0D43 2349        DC    AL1(#$@KSV)               SECTOR COUNT
    0D44 0980             0D45 2350        DC    AL2(#$$KSV)               BUFFER ADDRESS
                               2351 *** END OF EXPANSION ***
                               2353 *            READ IN THE PASSWORD DIRECTORY AND USER BLOCK TO LOOK
                               2354 *            FOR THE FILENAME
                               2355 *
    0D46 C0 87 164F            2356 KSA005 B     SFINDF                    GO GET PASSWORD USER DIRCTY
    0D4A 38 08 1B3E            2357        TBN   SMIND1,SM1PNF             WAS PASSWORD FOUND ?
    0D4E D0 10 91              2358        BT    KSA024(,@BR)              ERROR IF NO PASSWORD FOUND
    0D51 3C 80 0476            2359        MVI   $CIMSK,@NOP               MASK CONSOLE INTERRUPTS
                               2360 *       DSKL2 KSALT2,WAIT               START IN NULL DIRCTY
    0D55 C0 87 1268            2361        B     DL2ICS                    PERFORM RELATIVE DISK OP
    0D59 0D28             0D5A 2362        DC    AL2(KSALT2)               DPL ADDRESS
    0D5B C0 87 0025            2363        B     $DISKN                    WAIT AND CHECK DISK ERRORS
    0D5F 057F             0D60 2364        DC    AL2($WAITF)               WAIT DPL ADDRESS
                               2365 *** END OF EXPANSION ***

    0D61 35 02 03C7            2367        L     $XRSAV,@XR                SAVE ORIGINAL POINTER
                               2368 *       LOADR KSAOVL                    LOAD SAVE OVERLAY
    0D65 C0 87 051A            2369        B     $LOADR                    LOAD PROGRAM AND RETURN
    0D69 0D40             0D6A 2370        DC    AL2(KSAOVL)               DPL ADDRESS
                               2371 *** END OF EXPANSION ***

    0D6B C2 01 0C66            2373        LA    KSAVEN,@BR                RESTORE BASE REGISTER
    0D6F 38 80 1B3E            2374        TBN   SMIND1,SM1FNE             WAS NAME FOUND
    0D73 F2 10 1B              2375        JT    KSA007                    NO GO TEST FOR ** FILES
    0D76 35 02 1B56            2376        L     SMUDEA,@XR                POINTER TO DIRCTY ENTRY
    0D7A B8 10 0D              2377        TBN   ##DUES(,@XR),##MUEX       IS FILE POOLED ?
    0D7D 3C 4A 03CD            2378        MVI   $CAERR,@@E310             POOLED OR PROTECTED
    0D81 D0 10 91              2379        BT    KSA024(,@BR)              BRANCH YES
    0D84 B8 08 0D              2380        TBN   ##DUES(,@XR),##MUER       IS FILE PROTECTED ?
    0D87 3C 27 03CD            2381        MVI   $CAERR,@@E215             FILE IS PROTECTED
    0D8B D0 10 91              2382        BT    KSA024(,@BR)              BRANCH YES
    0D8E BB 04 0D              2383        SBF   ##DUES(,@XR),##MUEO       TURN OFF OPEN CLOSE BIT
    0D91 C0 87 110B            2384 KSA007 B     KSASTR                    GO TEST AND PROCESS ** FILE
    0D95 38 08 03D4            2385 KSA008 TBN   $INDR1,$WFLOK             IS WORK AREA PROTECTED ?
    0D99 F2 90 12              2386        JF    KSA030                    JUMP IF NOT PROTECTED
                               2387 *
                               2388 *            IF PROTECTED IS IT A ** FILE EDITED IN
                               2389 *
    0D9C 38 01 03D6            2390        TBN   $INDR3,$DBLOK             DOUBLE STAR INDICATOR
    0DA0 3C 2C 03CD            2391        MVI   $CAERR,@@E222             WORK FILE PROTECTED
    0DA4 D0 90 91              2392        BF    KSA024(,@BR)              BRANCH IF NOT **
                               2393 *
                               2394 *            ** FILE EDITED TO WORK AREA.  SAVE ONLY TO ** LIBRARY.
                               2395 *            CHECK IF ** PASSWORD SPECIFIED.
                               2396 *
    0DA7 3D 5C 1B46            2397        CLI   SMPSWD-##LPEN+2,@ASTER    IS SECOND CHAR AN *
    0DAB D0 01 91              2398        BNE   KSA024(,@BR)              ERROR RETURN
    0DAE 75 02 A9              2399 KSA030 L     KSAFBA(,@BR),@XR          GET POINTER TO FIT
                               2400 *       DSKL4 KSAFPL,WAIT               READ FIT TO CORE
    0DB1 C0 87 0987            2401        B     DL4ICS                    PERFORM RELATIVE DISK OP
    0DB5 0D0A             0DB6 2402        DC    AL2(KSAFPL)               DPL ADDRESS
    0DB7 C0 87 0025            2403        B     $DISKN                    WAIT AND CHECK DISK ERRORS
    0DBB 057F             0DBC 2404        DC    AL2($WAITF)               WAIT DPL ADDRESS
                               2405 *** END OF EXPANSION ***

    0DBD 4E 00 A1 043B         2407        ALC   KSA2PL+@DCNT(1,@BR),$EXFTR  ADD EXTENSION FACTOR
    0DC2 0E 00 0FFF 043B       2408        ALC   KSA235+@Q(1),$EXFTR
```

```
                               2409 *
    0DC8 38 20 03D4            2410 KSA032 TBN   $INDR1,$PGMDT             PAGE DATA FILE
    0DCC F2 90 1F              2411        JF    KSA035                    NO GO CALC FIT LENGTH
    0DCF 7C 00 A7              2412        MVI   KSAFLN(,@BR),@ZERO        NO FIT FOR PG DATA FILE
    0DD2 3C 87 0E38            2413        MVI   KSA050+@Q,@UCB            SET DATA FILE SWITCH
    0DD6 2C 00 0802 00         2414        MVC   KSARDT(1),@ZERO(,@XR)     DB COUNT TO READ TBL SECTOR CNT
    0DDB 3C 00 0801            2415        MVI   KSARDT-1,@ZERO            SET 0 DISPLACEMENT
    0DDF 3C 00 0804            2416        MVI   KSARDT+2,@ZERO            SET LAST ENTRY FLAG
    0DE3 3C 87 0FBC            2417        MVI   KSA222+@Q,@UCB            SET SWITCH SKIP FIT OUTPUT
    0DE7 3C 87 10E1            2418        MVI   KSA620+@Q,@UCB            SWITCH TO SKIP LINK CLEARING
    0DEB F2 87 24              2419        J     KSA045                    GO SET TOTAL FILE LENGTH
                               2420 *
    0DEE 7C 01 A7              2421 KSA035 MVI   KSAFLN(,@BR),KSAE01       INITIALIZE FIT LENGTH
    0DF1 6C 00 4B 00           2422        MVC   KSADBC(1,@BR),@ZERO(,@XR) GET DB COUNT
    0DF5 1C 00 0FBF B8         2423        MVC   KSA224+@Q(1),KSAC61(,@BR) FIT ENTRY COUNT FOR FIRST FIT
    0DFA 5F 00 4B B8           2424        SLC   KSADBC(1,@BR),KSAC61(,@BR)  DECR DB COUNT OVER 1 POSITIVE
    0DFE F2 04 11              2425        JNH   KSA045                      JUMP ONLY 1 SECTOR
    0E01 5E 00 A7 B7           2426 KSA040 ALC   KSAFLN(1,@BR),KSAC01(,@BR)  BUMP TO FIT LENGTH
    0E05 1E 00 0FBF B9         2427        ALC   KSA224+@Q(1),KSAC64(,@BR)   COUNT FOR NEXT FIT
    0E0A 5F 00 4B B9           2428        SLC   KSADBC(1,@BR),KSAC64(,@BR)  ECR FOR NEXT BLOCK OF FIT
    0E0E C0 84 0E01            2429        BH    KSA040                    BACK FOR NEXT BLOCK
                               2430 *
                               2431 *            FIT SIZE CALCULATED, ADD TO DB COUNT FOR TOTAL LENGTH
                               2432 *
    0E12 6C 00 4D 00           2433 KSA045 MVC   KSATFL(KSAE01,@BR),@ZERO(,@XR)  COUNT FROM FIT
    0E16 6C 01 58 02           2434        MVC   KSA#LN(@FLLNC,@BR),@FDLNC(,@XR)  LINE COUNT FROM FIT
    0E1A 5E 00 4D A7           2435        ALC   KSATFL(1,@BR),KSAFLN(,@BR)  FIT SIZE TO TOTAL
    0E1E 7C 00 4C              2436        MVI   KSATFL-1(,@BR),@ZERO      CLEAR HIGH ORDER BYTE
    0E21 39 80 03D4            2437        TBF   $INDR1,$BASIC             TEST IF BASIC FILE
    0E25 F2 10 0F              2438        JT    KSA050                    NO I/O SECTOR
    0E28 38 10 03E0            2439        TBN   $DBGUF,$IOPGS             2 SECTORS OF D1 ?
    0E2C F2 90 04              2440        JF    KSA047                    NO-BUMP CNT BY 1
    0E2F 5E 01 4D B7           2441        ALC   KSATFL(KSAE02,@BR),KSAC01(,@BR) BUMP FOR I/O SECTOR(S)
    0E33 5E 01 4D B7           2442 KSA047 ALC   KSATFL(KSAE02,@BR),KSAC01(,@BR) BUMP FOR I/O SECTOR(S)
    0E37 F2 80 56              2443 KSA050 JC    KSA140,@NOP               JUMP IF PROG DATA FILE
                               2444 *
                               2445 *            BUILD THE READ TABLE,
                               2446 *
    0E3A 6C 00 4F 00           2447 KSA060 MVC   KSAWRK(1,@BR),@ZERO(,@XR)   GET DB COUNT
    0E3E 6C 00 50 0C           2448        MVC   KSADB1(1,@BR),KSAFFD(,@XR)  GET FIRST DB
    0E42 5C 00 51 50           2449        MVC   KSARDC(1,@BR),KSADB1(,@BR)  SET UP INITIAL READ
    0E46 7C 01 52              2450        MVI   KSASCT(,@BR),@B1          INITALIZE SECTOR COUNT
                               2451 *
    0E49 5F 00 4F B7           2452 KSA070 SLC   KSAWRK(1,@BR),KSAC01(,@BR)  DECR DB COUNT
    0E4D F2 01 07              2453        JNE   KSA080                    NOT ZERO GET NEXT DB
    0E50 3C 87 0E73            2454        MVI   KSA100+@Q,@UCB            SET SWITCH
    0E54 F2 87 16              2455        J     KSA090                    MOVE CMD TO TABLE
                               2456 *
    0E57 E2 02 04              2457 KSA080 LA    KSAFEL(,@XR),@XR          BUMP TO NEXT DB
    0E5A 5E 00 50 B7           2458        ALC   KSADB1(1,@BR),KSAC01(,@BR)  BUMP DISP TEST IF CONTIG
    0E5E 6D 00 50 0C           2459        CLC   KSADB1(1,@BR),KSAFFD(,@XR)  COMPARE THE TWO ENTRIES
    0E62 F2 01 08              2460        JNE   KSA090                    NOT CONTIGUOUS
    0E65 5E 00 52 B7           2461        ALC   KSASCT(1,@BR),KSAC01(,@BR)  BUMP SECTOR COUNT IN READ
    0E69 C0 87 0E49            2462        B     KSA070                    BACK FOR NEXT DB
                               2463 *
    0E6D 1C 01 0802 52         2464 KSA090 MVC   KSARDT(KSAE02),KSASCT(,@BR)  MOVE CMD INTO TABLE
    0E72 F2 80 14              2465 KSA100 JC    KSA130,@NOP               JUMP LAST ENTRY
    0E75 1E 01 0E70 B1         2466        ALC   KSA090+@OP1(@CADDR),KSATBL(,@BR)  BUMP TABLE POINTER
    0E7A 6C 00 50 0C           2467        MVC   KSADB1(,@BR),KSAFFD(,@XR) GET NEXT FIT ENTRY
    0E7E 6C 00 51 0C           2468        MVC   KSARDC(,@BR),KSAFFD(,@XR) SET NEXT READ COMMAND
    0E82 7C 01 52              2469 KSA110 MVI   KSASCT(,@BR),@B1        SET NEW SECTOR COUNT
    0E85 C0 87 0E49            2470        B     KSA070                    BACK FOR NEXT READ
                               2471 *
    0E89 35 02 0E70            2472 KSA130 L     KSA090+@OP1,@XR           PICK UP POINTER TO TABLE
    0E8D BC 00 02              2473        MVI   KSAERL(,@XR),@ZERO        SET LAST FLAG ENTRY
    0E90 0C 01 1B6A 0D2D       2474 KSA140 MVC   SMNDBA,KSALT2+@DBFR2(@CADDR) NULL BUFFER CADDR
    0E96 1C 01 1B60 4D         2475        MVC   SMNSCT(@CADDR),KSATFL(,@BR)  TOTAL FILE SECTOR COUNT
    0E9B C0 87 0A50            2476 KSA145 B     SURCHN                    GO LOOK FOR SPACE
                               2477 *
                               2478 *            TEST IF A SPACE WAS FOUND
                               2479 *
    0E9F 1D 01 1B5E B3         2480        CLC   SMNDEA,KSAC00(@DADDR,@BR) NO SPACE IS ZERO DISK ADDR
    0EA4 F2 01 12              2481        JNE   KSA150                    JUMP IF SPACE WAS FOUND
    0EA7 39 80 1B3E            2482        TBF   SMIND1,SM1FNE             IS THIS AN OLD FILE
    0EAB F2 10 40              2483        JT    KSA166                    YES, GO CHECK OLD SIZE
    0EAE 1D 01 1B5C 4D         2484 KSA144 CLC   SMNULT(KSAE02),KSATFL(,@BR)  TEST FILE LENGTH TO TOTAL
                               2485 *
    0EB3 D0 82 94              2486        BL    KSA025(,@BR)              ERROR RETURN
    0EB6 F2 02 85              2487        JNL   KSA170                    GO PACK
                               2488 *
    0EB9 38 80 1B3E            2489 KSA150 TBN   SMIND1,SM1FNE             TEST 56T FOUND INDICATOR
    0EBD F2 90 03              2490        JF    KSA155                    NOT ON IS NAME FOUND
    0EC0 F2 10 D6              2491        JT    KSA210                    GO MAKE ENTRY FOR STUFID
                               2492 *
                               2493 *            SET UP THE POINTER TO THE USER DIRCTY ENTRY TO
                               2494 *            TEST THE STATUS BYTE AND PROCESS THE NEW ENTRY
                               2495 *
    0EC3 35 02 1B56            2496 KSA155 L     SMUDEA,@XR                GET ENTRY ADDR
    0EC7 F2 80 04              2497 KSA156 JC    KSA157,@NOP               SWITCH NEW/OLD HEADER
    0ECA 6C 18 18 2B           2498        MVC   KSAHED(##LUEH,@BR),##DUEH(,@XR)  GET OLD HEADER IF NO NEW
                               2499 *
    0ECE 6C 01 A0 09           2500 KSA157 MVC   KSA2DA(@DADDR,@BR),##DUEA(,@XR)  OLD FILE DADDR
    0ED2 8D 01 09 1B5E         2501        CLC   ##DUEA(@CADDR,@XR),SMNDEA  TEST WHICH SPACE LOWER
    0ED7 C0 84 0F78            2502        BH    KSA180                    GO SEND OLD SPACE TO NULL
                               2503 *
                               2504 *            THE NEW SPACE IS NOT CLOSER, IF THE NEW FILE IS LARGER
                               2505 *            THE NEW SPACE MUST BE USED ANY WAY.
                               2506 *
    0EDB 6D 01 4D 0B           2507       CLC    KSATFL(KSAE02,@BR),##DUEF(,@XR)  CHECK FILE SIZE
    0EDF C0 84 0F78            2508       BH     KSA180                    GO SET UP TO SUE NEW SPACE
    0EE3 F2 82 A6              2509       JL     KSA200                    RETURN NEW SPACE
                               2510 *
                               2511 *            THE OLD FILE SPACE WILL BE USED.  THE LENGTHS ARE EQUAL
                               2512 *            SO SET SWITCH TO SKIP OVER WRITING THE NULL DIRCTY
                               2513 *            BACK TO DISK.
                               2514 *
    0EE6 3C 87 0FAF            2515       MVI    KSA221+@Q,@UCB            SET SWITCH TO SKIP
    0EEA C0 87 0F5E            2516       B      KSA175                    BACK TO UPDATE DIRCTY ENTRY
                               2517 *
                               2518 *            CHECK IF A NEW SPACE WAS FOUND.  IF FOUND GO TEST WHICH
                               2519 *            SPACE IS CLOSER TO LIBRARY START.
                               2520 *
                               2521 *            NO NEW SPACE WAS FOUND, TEST IF OLD FILE SPACE IS LARGE
                               2522 *            ENOUGH TO HOLD THE NEW FILE.
                               2523 *
    0EEE 35 02 1B56            2524 KSA166 L     SMUDEA,@XR
    0EF2 6D 01 4D 0B           2525        CLC   KSATFL(KSAE02,@BR),##DUEF(,@XR)  TEST FILE SIZE
    0EF6 C0 84 0EAE            2526        BH    KSA144                    TO SMALL GO TEST FOR SPACKU
    0EFA 6C 01 A0 09           2527        MVC   KSA2DA(@DADDR,@BR),##DUEA(,@XR)  SET OLD SPACE ADDR
                               2528 *
                               2529 *            IF NEW ENTRY NOT HIGH TEST IF EQUAL
                               2530 *
    0EFE F2 81 5D              2531        JE    KSA175
                               2532 *
                               2533 *            NEW FILE IS LESS THAN OLD, SEND EXTRA SPACE TO THE NULL
                               2534 *            DIRCTY.  CALCULATE A NEW STARTING ADDR FOR AVAILABLE SPACE.
                               2535 *
    0F01 6C 03 56 0B           2536 KSA16@ MVC   KSANDA(@DADDR+##LUEF,@BR),##DUEF(,@XR)  OLD ADDR AND COUNT
    0F05 5F 01 56 4D           2537        SLC   KSANDA(##LUEF,@BR),KSATFL(,@BR)  DECR BY LENGTH
    0F09 1C 00 0F2A 53         2538        MVC   KSA164+@Q(1),KSAOFA-1(,@BR)  CYL COUNT
    0F0E 7C 00 53              2539        MVI   KSAOFA-1(,@BR),@ZERO      CLEAR HIGH ORDER BYTE
    0F11 5E 01 54 4D           2540        ALC   KSAOFA(@CADDR,@BR),KSATFL(,@BR)  BUMP SECTOR COUNT
    0F15 5F 01 54 B4           2541 KSA163 SLC   KSAOFA(@DADDR,@BR),KSAC48(,@BR)  DECR BY CYL COUNT
    0F19 F2 82 09              2542        JL    KSA16#                    GO MOVE IN CYL VALUE
    0F1C 1E 00 0F2A B7         2543        ALC   KSA164+@Q(1),KSAC01(,@BR) BUMP CYL
    0F21 C0 87 0F15            2544        B     KSA163                    BACK TO GET NEXT VALUE
    0F25 5E 00 54 B4           2545 KSA16# ALC   KSAOFA(1,@BR),KSAC48(,@BR)  RESTORE REMAINDER
    0F29 3C 00 0CB9            2546 KSA164 MVI   KSAOFA-1,*-*              SET IN NEW CYL
    0F2D 0C 01 1B62 0F2C       2547        MVC   SMNETD,KSA164+@OP1(@CADDR)  ENTRY ADDR FOR STORIN
    0F33 C0 87 13B2            2548 KSA165 B     STORIN                    GO MAKE ENTRY TO NULL DIRCTY
                               2549 *
    0F37 38 20 1B3E            2550        TBN   SMIND1,SM1STN             WAS ENTRY MADE
    0F3B F2 90 20              2551        JF    KSA175                    ENTRY MADE GO UPDATE USER BLOCK
                               2552 *
                               2553 *            UNABLE TO PLACE ENTRY IN NULL DIRCTY.
                               2554 *            CALL THE LIBRARY PACKING ROUTINE.
                               2555 *
    0F3E 0C 05 0449 0D33       2556 KSA170 MVC   $DPLSV(@DPLNG),KSASPU     SET DPL TO RELOAD SAVE
    0F44 0C 01 06FF 1B58       2557        MVC   $$FLIB(@DADDR),SMBFDA     LIBR BASE ADDR
    0F4A 35 02 03C7            2558        L     $XRSAV,@XR                RESTORE INPUT LINE POINTER
                               2559 *       LOADR KSAIOR                    RELOAD THE 10 ROUTINES
    0F4E C0 87 051A            2560        B     $LOADR                    LOAD PROGRAM AND RETURN
    0F52 0D3A             0F53 2561        DC    AL2(KSAIOR)               DPL ADDRESS
                               2562 *** END OF EXPANSION ***

    0F54 35 02 03C7            2564        L     $XRSAV,@XR                SAVE INPUT LINE POINTER
                               2565 *       RLOAD KSAPAC                    LOAD AND EXECUTE SPACKU
    0F58 C0 87 051E            2566        B     $RLOAD                    LOAD AND EXECUTE PGM
    0F5C 0D34             0F5D 2567        DC    AL2(KSAPAC)               DPL ADDRESS
                               2568 *** END OF EXPANSION ***

                               2570 *
                               2571 *            UPDATE OLD DIRCTY ENTRY
                               2572 *
    0F5E C0 87 1081            2573 KSA175 B     KSA500                    GO CREATE DIRCTY ENTRY
                               2574 *
    0F62 7C 02 AA              2575        MVI   KSALT4(,@BR),@DPUT        CHANGE OP CODE TO WRITE
    0F65 4C 01 AF 1B5A         2576        MVC   KSALT4+@DBFR2(@CADDR,@BR),SMUDBA  GET BUFFER CADDR
    0F6A 4C 01 AC 1B6C         2577         MVC   KSALT4+@DSAD(@DADDR,@BR),SMDAAD  ACTIVE DIRCTY RDADDR
    0F6F C0 87 1268            2578         B     DL2ICS                    DISK IOCS
    0F73 0D10             0F74 2579         DC    AL2(KSALT4)               DPL FOR USER DIRCTY
                               2580 *
    0F75 F2 87 36              2581         J     KSA221                    GO START I/O
                               2582 *
                               2583 *             SET UP TO USE THE NEW SPACE.  SEND THE OLD SPACE TO THE
                               2584 *             NULL DIRCTY.
                               2585 *
    0F78 2C 03 0CBC 0B         2586 KSA180 MVC   KSANDA(@DADDR+##LUEF),##DUEF(,@XR)  GET OLD SPACE
    0F7D 0C 01 1B62 0F2C       2587         MVC   SMNETD(@CADDR),KSA164+@OP1  ENTRY ADDR
    0F83 4C 01 A0 1B5E         2588         MVC   KSA2DA(@CADDR,@BR),SMNDEA   NEW SPACE DADDR
    0F88 C0 87 0F33            2589         B     KSA165                    BACK TO CALL STORIN
                               2590 *
                               2591 *      NEW SPACE FOR ENTRY FOUND, TEST TO SEE WHICH IS CLOSER
                               2592 *      TO THE BEGINING OF THE LIBRARY.
                               2593 *
    0F8C 1C 01 1B62 BB         2594 KSA200 MVC   SMNETD,KSASMN(@CADDR,@BR) SEND NEW SPACE BACK TO NULL
    0F91 C0 87 13B2            2595         B     STORIN                    PUT IN DIRCTY
                               2596 *
                               2597 *             GO BACK TO PLACE THE END OF THE OLD SPACE IN THE NULL
                               2598 *             DIRCTY.
                               2599 *
    0F95 C0 87 0F01            2600         B     KSA16@                    BACK TO SPLIT SIZE
```

```
                               2602 *
                               2603 *            THE FILE TO BE SAVED IS A NEW FILE.  SET UP NEW ENTRY
                               2604 *            TO GO INTO THE USER DIRCTY.
                               2605 *
    0F99 C2 02 0C7F            2606 KSA210 LA    KSAUDE,@XR                ADDR OF NEW ENTRY
    0F9D 4C 01 A0 1B5E         2607        MVC   KSA2DA(@DADDR,@BR),SMNDEA GET NEW SPACE DADDR
    0FA2 C0 87 1081            2608        B     KSA500                    GO MAKE NEW ENTRY
    0FA6 34 02 1B64            2609        ST    SMUPEN,@XR                SET ENTRY ADDR
    0FAA C0 87 1566            2610        B     STUFID
                               2611 *
                               2612 *            IF THE NULL DIRCTY WAS CHANGED FALL THROUGH AND WRITE
                               2613 *            THE DIRCTY BACK ON DISK, IF NOT CHANGE JUMP OVER WRITE
                               2614 *            AND START I/O PROCESSING.
                               2615 *
    0FAE F2 80 0A              2616 KSA221 JC    KSA222,@NOP               WRITE NULL DIRCTY SWITCH
    0FB1 3C 02 0D28            2617        MVI   KSALT2,@DPUT              CHANGE READ OP TO WRITE
    0FB5 C0 87 1268            2618        B     DL2ICS                    WRITE NULL DIRCTY BACK
    0FB9 0D28             0FBA 2619        DC    AL2(KSALT2)               PARAMETER LIST
                               2620 *
                               2621 *            SET UP TO START OUTPUT OF THE FILE.  SET DPL TO WRITE
                               2622 *            FIT AND I/O SECTOR IF THERE IS ONE.  IF FILE IS PROGRAM
                               2623 *            GENERATED THERE IS NO FIT OR I/O SECTOR.  IF THE FILE
                               2624 *            IS A CARD OR KEYBOARD DATA FILE THERE IS NO I/O SECTOR.
                               2625 *
    0FBB F2 80 2B              2626 KSA222 JC    KSA229,@NOP               FIT OUTPUT SWITCH
    0FBE 7C 00 4F              2627 KSA224 MVI   KSAWRK(,@BR),*-*          TOTAL FIT ENTRIES
    0FC1 C2 02 164F            2628        LA    SFINDF,@XR                BUFFER POINTER
    0FC5 BC 00 0C              2629 KSA225 MVI   KSAFFD(,@XR),*-*          SET DISP IN FIT ENTRY
    0FC8 1E 00 0FC6 B7         2630        ALC   KSA225+@Q(1),KSAC01(,@BR) BUMP DISP
    0FCD E2 02 04              2631        LA    KSAFLG(,@XR),@XR          BUMP TO NEXT FIT
    0FD0 5F 00 4F B7           2632        SLC   KSAWRK(1,@BR),KSAC01(,@BR)  DECR DB COUNT
    0FD4 C0 01 0FC5            2633        BNE   KSA225                    NEXT DISP IF NOT ZERO
    0FD8 7C 02 A4              2634        MVI   KSAFPL(,@BR),@DPUT        CHANGE OP TO WRITE
    0FDB 5C 01 A6 A0           2635        MVC   KSAFSZ(@DADDR,@BR),KSA2DA(,@BR)  OUTPUT DADDR
    0FDF C0 87 1268            2636        B     DL2ICS                    WRITE FIT AND I/O IF THERE IS 1
    0FE3 0D0A             0FE4 2637        DC    AL2(KSAFPL)               FIT DPL
    0FE5 5E 00 A0 A7           2638        ALC   KSA2DA(1,@BR),KSAFLN(,@BR)  BUMP DISK ADDR
    0FE9 7C 00 5D              2639 KSA229 MVI   KSABCT(,@BR),@ZERO        CLEAR BUFFER COUNT
    0FEC C2 02 0801            2640        LA    KSARDT-1,@XR              SET POINTER TO READ TABLE
    0FF0 BD 00 01              2641 KSA230 CLI   @B1(,@XR),@ZERO           CHECK IF LAST ENTRY
    0FF3 F2 81 73              2642        JE    KSA300                    YES LAST
    0FF6 6C 01 9B 01           2643        MVC   KSA4PL+@DCNT(KSAE02,@BR),@B1(,@XR)  PLUG INTO DPL
    0FFA 6E 00 5D 01           2644        ALC   KSABCT(1,@BR),@B1(,@XR)   CE-SECTOR COUNT TO BUFFER COUNT
    0FFE 7D 0C 5D              2645 KSA235 CLI   KSABCT(,@BR),KSAMXT       BUFFER EXCEEDED
    1001 F2 81 1C              2646        JE    KSA260                    EXACTLY FULL
    1004 F2 82 44              2647        JL    KSA280                    NOT FULL YET
                               2648 *
                               2649 *            BUFFER WILL OVERFLOW.  SPLIT THE READ
                               2650 *
                          0FFF 2651 KSABSZ EQU   KSA235+@Q                 POINTER TO BUFFER SIZE
    1007 4F 00 5D 0FFF         2652        SLC   KSABCT(1,@BR),KSABSZ      GET OVER SECTORS
    100C 9F 00 01 5D           2653        SLC   @B1(1,@XR),KSABCT(,@BR)   GET OVERFLOW COUNT
    1010 AE 00 00 01           2654        ALC   @ZERO(1,@XR),@B1(,@XR)    BUMP DISP
    1014 5F 00 9B 5D           2655        SLC   KSA4PL+@DCNT(1,@BR),KSABCT(,@BR)  ADJUST CURRENT READ
    1018 9C 00 01 5D           2656        MVC   @B1(1,@XR),KSABCT(,@BR)   SET NEXT READ SECTOR COUNJ
    101C 3C 87 103A            2657        MVI   KSA275+@Q,@UCB            TABLE BUMP SWITCH
    1020 C0 87 1053            2659 KSA260 B     KSA290                    GO READ
    1024 C0 87 10DC            2660        B     KSA610                    GO CLEAR LINK BYTES
    1028 C0 87 1268            2661        B     DL2ICS                    WRITE TO LIBRARY
    102C 0D04             102D 2662        DC    AL2(KSA2PL)               POINTER TO DPL
                               2663 *
                               2664 *            RESET I/O POINTERS TO THE BUFFER AREA
                               2665 *
    102E 5E 00 A0 A1           2666        ALC   KSA2PL+@DSAD(1,@BR),KSA2PL+@DCNT(,@BR)  BUMP DADDR
    1032 7C 00 5D              2667        MVI   KSABCT(,@BR),@ZERO        CLEAR BUFFER COUNT
    1035 5C 01 9D A3           2668        MVC   KSA4IA(@CADDR,@BR),KSA2IA(,@BR)  RESTORE INPUT CADDR
    1039 F2 80 07              2669 KSA275 JC    KSA276,@NOP
    103C E2 02 02              2670        LA    KSAERL(,@XR),@XR          BUMP TO NEXT ENTRY
    103F C0 87 0FF0            2671        B     KSA230                    GO GET NEXT READ TBL ENTRY
                               2672 *
                               2673 *            IF SPLIT READ DO NOT BUMP TABLE ADDR
                               2674 *
    1043 3C 80 103A            2675 KSA276 MVI   KSA275+@Q,@NOP            RESET SWITCH
    1047 C0 87 0FF0            2676        B     KSA230                    BACK TO GET ENTRY
                               2677 *
                               2678 *            BUFFER NOT FULL READ IN A GROUP OF BLOCKS FROM WORK AREA
                               2679 *
    104B C0 87 1053            2680 KSA280 B     KSA290                    GO READ
    104F C0 87 1039            2681        B     KSA275                    GO BUMP TO NEXT ENTRY
    1053 34 08 1068            2682 KSA290 ST    KSA295+@OP1,@ARR          SET RETURN
    1057 5E 00 9A B5           2683        ALC   KSA4PL+@DSAD(1,@BR),KSAFDB(,@BR)  FIRST DATA BLOCK
    105B C0 87 0987            2684        B     DL4ICS                    GO READ FROM WORK AREA
    105F 0CFE             1060 2685        DC    AL2(KSA4PL)
    1061 5E 00 9C 9B           2686        ALC   KSA4PL+@DBFR1(1,@BR),KSA4PL+@DCNT(,@BR)
    1065 C0 87 0000            2687 KSA295 B     *-*                       RETURN
                               2688 *
                               2689 *            END OF THE READ TABLE, TEST IF ANY SECTORS ARE STILL IN
                               2690 *            THE BUFFER, IF SO WRITE THEM OUT THEN RETURN.  IF NOT
                               2691 *            THE JUST RETURN
                               2692 *
    1069 7D 00 5D              2693 KSA300 CLI   KSABCT(,@BR),@ZERO        CHECK ZERO BUFFER COUNT
    106C F2 81 0E              2694        JE    KSA320                    YES, GO RETURN
    106F 5C 00 A1 5D           2695        MVC   KSA2PL+@DCNT(1,@BR),KSABCT(,@BR)  REMAINING COUNT
                               2696 *
    1073 C0 87 10DC            2697        B     KSA610                    GO CLEAR LINK BYTES
    1077 C0 87 1268            2698        B     DL2ICS                    2 SURFACE IOCS
    107B 0D04             107C 2699        DC    AL2(KSA2PL)               PARAMETER LIST
    107D C0 87 04A1            2700 KSA320 B     $CARPL                    RETURN TO SYSTEM
                               2701 *
                               2702 *            ROUTINE TO CREATE A NEW USER DIRCTY ENTRY
                               2703 *
    1081 34 08 10DB            2704 KSA500 ST    KSA560+@OP1,@ARR          SAVE RETURN ADDR
    1085 8C 02 12 043A         2705        MVC   ##DUED(##LUED,@XR),$DATE  CURRENT DATE
    108A 9C 01 0B 4D           2706        MVC   ##DUEF(##LUEF,@XR),KSATFL(,@BR)  FILE LENGHT
    108E 9C 00 0C A7           2707        MVC   ##DUEI(##LUEI,@XR),KSAFLN(,@BR)  FIT LENGHT
    1092 9C 01 09 A0           2708        MVC   ##DUEA(@DADDR,@XR),KSA2DA(,@BR)  STARTING ADDR
    1096 9C 01 0F 58           2709        MVC   ##DUEL(##LUEL,@XR),KSA#LN(,@BR)  NO. OF LINES
    109A 8C 07 07 1B54         2710        MVC   ##DUEN(##LUEN,@XR),SMFNAM  MOVE IN FILE NAME
    109F 9C 18 2B 18           2711        MVC   ##DUEH(##LUEH,@XR),KSAHDR+##LUEH-1(,@BR)  HEADER
    10A3 8C 00 0D 03D4         2712        MVC   ##DUES(,@XR),$INDR1       SET STATUS
    10A8 BB 1C 0D              2713        SBF   ##DUES(,@XR),KSABTS       CLEAR EXTRA BITS
    10AB 38 01 03D6            2714        TBN   $INDR3,$DBLOK             IS THIS A PROTECTED ** FILE
    10AF F2 90 03              2715        JF    KSA555                    NO, SO RETURN
    10B2 BA 08 0D              2716        SBN   ##DUES(,@XR),##MUER       YES, SO SET PROTECT STATUS
                               2717 *
    10B5 39 80 03D4            2718 KSA555 TBF   $INDR1,$BASIC             TEST IF BASIC FILE
    10B9 F2 10 1C              2719        JT    KSA560                    RETURN ITS DATA
    10BC 5E 00 C0 A7           2720        ALC   KSAL3A-1(1,@BR),KSAFLN(,@BR) BUMP CADDR OF BUFFER
    10C0 38 10 03E0            2721        TBN   $DBGUF,$IOPGS             2 SECTORS OF D1 ?
    10C4 F2 90 07              2722        JF    KSA557                    NO-READ 1 SECTOR
    10C7 5E 00 A7 B7           2723        ALC   KSAFLN(1,@BR),KSAC01(,@BR)  BUMP SECTOR CNT FIT+I/O
    10CB 7C 02 BF              2724        MVI   KSALT3+@DCNT(,@BR),#@@#SC READ 2 I/O SECTORS
    10CE C0 87 0025            2725 KSA557 B     $DISKN                    READ I/O SECTOR(S) IN
    10D2 0D22             10D3 2726        DC    AL2(KSALT3)               DPL TO READ I/O SECTOR(S)
    10D4 5E 00 A7 B7           2727        ALC   KSAFLN(1,@BR),KSAC01(,@BR) SECTOR COUNT FIT+I/O OUTPUT
    10D8 C0 87 0000            2728 KSA560 B     *-*                       RETURN
                               2729 *
                               2730 *            ROUTINE TO CLEAR THE LINK BYTES IN THE FIT
                               2731 *
    10DC 34 08 110A            2732 KSA610 ST    KSA690+@OP1,@ARR          SAVE RETURN ADDR
    10E0 F2 80 24              2733 KSA620 JC    KSA690,@NOP               SKIP IF DATA FILE
    10E3 C0 87 0025            2734        B     $DISKN                    GO WAIT FOR LAST READ
    10E7 057F             10E8 2735        DC    AL2($WAITF)               WAIT DPL
    10E9 3C 00 1301            2736 KSA630 MVI   KSAIOB,@ZERO              CLEAR DB LINK BYTE
    10ED 1E 00 10EB B7         2737        ALC   KSA630+@OP1-1(1),KSAC01(,@BR)  BUMP TO NEXT SECTOR
    10F2 1E 00 10F8 B7         2738        ALC   KSA640+@Q,KSAC01(1,@BR)   BUMP COUNTER
    10F7 7D 00 A1              2739 KSA640 CLI   KSA2PL+@DCNT(,@BR),*-*    ALL LINKS CLEARED
    10FA C0 01 10E9            2740        BNE   KSA630                    NOT ALL CLEARED
    10FE 1C 01 10EC A3         2741        MVC   KSA630+@OP1,KSA2IA(@CADDR,@BR)  RESET BUFFER CADDR
    1103 3C 00 10F8            2742        MVI   KSA640+@Q,@ZERO           CLEAR COUNT
    1107 C0 87 0000            2743 KSA690 B     *-*                       RETURN
```

```
                               2745 ***********************************************************************
                               2746 *            KSASTR - ACTION TO PROCESS A ** FILE SPECIFICATION.      *
                               2747 *            CALL SFINDF TO LOCATE THE FILE OR FIRST LIBRARY WITH     *
                               2748 *            AVAILABLE SPACE.                                         *
                               2749 ***********************************************************************
    110B 3D 5C 1B45            2750 KSASTR CLI   SMPSWD-##LPEN+1,@ASTER    IS THIS A * OR ** FILE
    110F C0 01 0D95            2751        BNE   KSA008                    NO RETURN TO MAIN FLOW
    1113 3D 5C 1B46            2752        CLI   SMPSWD-##LPEN+2,@ASTER    MUST BE A ** FILE NAME
    1117 3C 14 03CD            2753        MVI   $CAERR,@@E135             INVALID USE OF * OR ** NAME
    111B D0 01 91              2754        BNE   KSA024(,@BR)              GO TO ERROR RETURN
    111E 3D 40 1B3F            2755        CLI   SMVOID-@VOLID+1,@BLANK    WAS A VOLUME SPECIFIED
    1122 F2 01 27              2756        JNE   KSAST1                    JUMP YES NULL DIRCTY IS IN CORE
    1125 39 80 1B3E            2757        TBF   SMIND1,SM1FNE             WAS FILENAME FOUND
    1129 F2 10 20              2758        JT    KSAST1                    YES GO READ NULL DIRCTY
                               2759 *
                               2760 *            NAME WAS NOT FOUND, GET FIRST AVAIL LIBR ON SYSTEM
                               2761 *
    112C 3C 80 1738            2762 KSAST@ MVI   SFISTR,@NOP               INHIBIT DISK SWITCHING
    1130 3C 80 173B            2763        MVI   SFIFND,@NOP               SET SFINDF ** SWITCH
    1134 3C 80 1670            2764        MVI   SFIVOL,@NOP               INHIBIT CALL TO SVOLID
    1138 3C 01 1769            2765        MVI   SFINTR,@B1                SET COUNTER FOR NEXT DISK
    113C C0 87 164F            2766        B     SFINDF                    SET UP LIBR POINTERS
                               2767 *
                               2768 *            READ THE NULL DIRCTY FOR THE APPROPRIATE DISK
                               2769 *
                               2770 *KSAST0 DSKL2 KSALTZWAIT                 READ NULL DIRCTY
    1140 C0 87 1268            2771 KSAST0 B     DL2ICS                    PERFORM RELATIVE DISK OP
    1144 0D28             1145 2772        DC    AL2(KSALT2)               DPL ADDRESS
    1146 C0 87 0025            2773        B     $DISKN                    WAIT AND CHECK DISK ERRORS
    114A 057F             114B 2774        DC    AL2($WAITF)               WAIT DPL ADDRESS
                               2775 *** END OF EXPANSION ***

                               2777 *
    114C 38 20 03D4            2778 KSAST1 TBN   $INDR1,$PGMDT             IS NEW FILE PROGRAM GENERATED
    1150 F2 10 06              2779        JT    KSAST9                    SKIP BUMP FOR FIT AND I/O
                               2780 *
    1153 0E 00 1B60 1203       2781        ALC   SMNSCT,KSAC04             ADJUST FOR POSSIBLE FIT+I/O
    1159 0C 01 1B6A 0D2D       2782 KSAST9 MVC   SMNDBA(@CADDR),KSALT2+@DBFR2  SET UP NULL DIRCTY CADDR
    115F C0 87 0A50            2783        B     SURCHN                    TEST IF SPACE AVAIL
    1163 1D 01 1B5E B3         2784        CLC   SMNDEA(@CADDR),KSAC00(,@BR)  WAS SPACE FOUND
    1168 C0 01 11EC            2785        BNE   KSAST3                    GO RETURN TEST SPACE FOUND
                               2786 *
                               2787 *            TEST IF LIBRARY COULD BE PACKED TO SAVE FILE
                               2788 *
    116C 0D 00 1B60 1B5C       2789        CLC   SMNSCT,SMNULT             REQUESTED LESS THAN TOTAL
    1172 C0 82 0F3E            2790        BL    KSA170                    GO PACK LIBR
                               2791 *
                               2792 *            TOTAL NOT ENOUGH, IF NAME FOUND WILL TOTAL PLUS THE
                               2793 *            SPACE OF THE OLD FILE BE ENOUGH. IF NOT ABORT. OTHERWISE
                               2794 *            SEND OLD SPACE TO NULL AND THEN PACK. IF NAME NOT FOUND
                               2795 *            GO TO NEXT AVAILABLE DISK.
                               2796 *
    1176 39 80 1B3E            2797        TBF   SMIND1,SM1FNE             WAS NAME FOUND
    117A F2 10 23              2798        JT    KSAST2                    YES
                               2799 *
                               2800 *            GO FIND THE NEXT AVAIL DISK
                               2801 *
    117D 3D 40 1B3F            2802        CLI   SMVOID-@VOLID+1,@BLANK    WAS A VOLUME SPECIFIED
    1181 3C 43 03CD            2803        MVI   $CAERR,@@E300             LIBRARY SPACE NOT AVAILABLE
    1185 D0 01 91              2804        BNE   KSA024(,@BR)              ERROR RETURN
    1188 1E 00 1769 B7         2805        ALC   SFINTR(1),KSAC01(,@BR)    BUMP COUNTER FOR NEXT DISK
    118D C0 87 164F            2806        B     SFINDF                    GO TO NEXT DISK
    1191 3D 06 1769            2807        CLI   SFINTR,SFIETD             TEST IF MAX TRIES
    1195 3C 43 03CD            2808        MVI   $CAERR,@@E300             LIBRARY SPACE NOT AVAILABLE
    1199 D0 81 91              2809        BE    KSA024(,@BR)              ERROR EXIT
    119C C0 87 1140            2810        B     KSAST0                    BACK FOR NEXT NULL
                               2811 *
                               2812 *            TEST IF OLD SPACE WILL ALLOW FILE TO BE SAVED
                               2813 *
    11A0 2E 00 1B5C 0B         2814 KSAST2 ALC   SMNULT,##DUEF(,@XR)       ADD FILE LNGTH TO NULL TOTAL
    11A5 0D 00 1B5C 1B60       2815        CLC   SMNULT,SMNSCT             IS TOTAL ENOUGH NOW ?
    11AB 3C 43 03CD            2816        MVI   $CAERR,@@E300             LIBRARY SPACE NOT AVAILABLE
    11AF D0 82 91              2817        BL    KSA024(,@BR)              BRANCH NOT ENOUGH
                               2818 *
                               2819 *            SEND OLD FILE SPACE TO THE NULL DIRCTY
                               2820 *
    11B2 E2 02 08              2821        LA    ##DUEA-1(,@XR),@XR        CALC ADDR OF ENTRY FOR STORIN
    11B5 34 02 1B62            2822        ST    SMNETD,@XR                POINTER TO OLD ADDR AND SIZE
    11B9 C0 87 13B2            2823        B     STORIN                    PLACE OLD FILE IN NULL DIRCTY
    11BD 38 20 1B3E            2824        TBN   SMIND1,SM1STN             WAS THE ENTRY MADE
    11C1 D0 10 91              2825        BT    KSA024(,@BR)              NO, GO TO ERROR EXIT
                               2826 *
                               2827 *            OLD SPACE IN NULL DIRCTY, CLEAR COUNT IN OLD FILE DIRCTY
                               2828 *            ENTRY AND THEN GO TO SPACKU
                               2829 *
    11C4 E2 02 03              2830        LA    ##LAAA+##LUEF-1(,@XR),@XR POINT TO RIGHT END FILE LENGTH
    11C7 AF 01 00 00           2831        SLC   @ZERO(##LUEF,@XR),@ZERO(,@XR)  CLEAR SECTOR COUNT
    11CB 3C 02 0D28            2832        MVI   KSALT2,@DPUT              SET WRITE OP FOR NULL DIRCTY
                               2833 *       DSKL2 KSALT2                    WRITE NULL BACK
    11CF C0 87 1268            2834        B     DL2ICS                    PERFORM RELATIVE DISK OP
    11D3 0D28             11D4 2835        DC    AL2(KSALT2)               DPL ADDRESS
                               2836 *** END OF EXPANSION ***

    11D5 35 02 1B5A            2838        L     SMUDBA,@XR                POINTER TO USER DIRCTY CADDR
    11D9 34 02 11FF            2839        ST    KSAUDB+@DBFR2,@XR         SET BUFFER CADDR IN DPL
    11DD 2C 01 11FC 01         2840        MVC   KSAUDB+@DSAD,##DUHA(@DADDR,@XR)  DADDR FROM HEADER TO DPL
                               2841 *       DSKL2 KSAUDB                    WRITE OUT THE USER DIRCTY
    11E2 C0 87 1268            2842        B     DL2ICS                    PERFORM RELATIVE DISK OP
    11E6 11FA             11E7 2843        DC    AL2(KSAUDB)               DPL ADDRESS
                               2844 *** END OF EXPANSION ***

    11E8 C0 87 0F3E            2846        B     KSA170                    GO PACK THE LIBRARY
    11EC 0C 01 1B62 1201       2847 KSAST3 MVC   SMNETD(@CADDR),KSASTA     SET POINTER TO NEW NULL ENTRY
    11F2 C0 87 13B2            2848        B     STORIN                    GO RETURN THE SPACE
    11F6 C0 87 0D95            2849        B     KSA008                    RETURN TO MAIN LINE

                               2851 *KSAUDB DPL   FUNC=@DPUT,DADDR=*-*,CNT=@@MAAA,CADDR=*-*
                          11FA 2852 KSAUDB EQU   *                         DISK PARAMETER LIST
    11FA 02               11FA 2853        DC    AL1(@DPUT)                REQUESTED FUNCTION
    11FB 0000             11FC 2854        DC    AL2(*-*)                  DISK ADDRESS
    11FD 02               11FD 2855        DC    AL1(##LAAA)               SECTOR COUNT
    11FE 0000             11FF 2856        DC    AL2(*-*)                  BUFFER ADDRESS
```

```
                               2857 *** END OF EXPANSION ***

    1200 1B5D             1201 2859 KSASTA DC    AL2(SMNDEA-1)             ADDR OF NEW ENTRY
    1202 0004             1203 2860 KSAC04 DC    IL2'4'                    MAX COUNT FOR FIT AND I/O
    1204                  1267 2861 KSAPCH DS    CL100                     KSAVEN PATCH AREA
                          1268 2862 KSAEND EQU   *                         END OF CODING
```

```
                               2864 *
                               2865 *            WORK AREA FOR SAVE FUNCTION, THE WORKAREA OVERLAYS THE
                               2866 *            BEGINNING OF THE PROGRAM WHICH IS NOT USED ONCE THE SYNTAX
                               2867 *            HAS BEEN CHECKED.
                               2868 *
    0C66                       2869        ORG   KSAVEN
                          0C66 2870 KSAHDR EQU   *                         FILE HEADER
    0C66                  0C7E 2871 KSAHED DS    CL25                      SAVE AREA FOR HEADER
                          0C7F 2872 KSAUDE EQU   *                         NEW ENTRY
    0C7F                  0CB0 2873 KSADIR DS    CL(##LUE)                 SPACE FOR NEW ENTRY
    0CB1                  0CB1 2874 KSADBC DS    CL1                       DB COUNT
    0CB2                  0CB3 2875 KSATFL DS    CL2                       TOTAL FILE LENGTH
    0CB4                  0CB5 2876 KSAWRK DS    CL2                       WORK COUNTER
                               2877 *
    0CB6                  0CB6 2878 KSADB1 DS    CL1                       DB DISP 1
    0CB7                  0CB7 2879 KSARDC DS    CL1                       READ COMMAND
    0CB8                  0CB8 2880 KSASCT DS    CL1                       READ SECTOR COUNT
    0CB9                  0CBA 2881 KSAOFA DS    CL(@CADDR)                OLD FILE ADDR
    0CBB                  0CBC 2882 KSANDA DS    CL(@DADDR)                NEW DISK ADDR
                          0CBB 2883 KSANCC EQU   KSANDA-1                  CYL COUNT OF NEW DISK ADDR
    0CBD                  0CBE 2884 KSA#LN DS    CL(##LUEL)                SPACE FOR NO. OF LINES IN FILE
    0CBF                  0CC0 2885 KSANSA DS    CL(@DADDR)                NEW START ADDR OF ENTRY
    0CC1                  0CC2 2886 KSAOFS DS    CL(##LUEF)                OLD FILE SPACE SAVE AREA
    0CC3                  0CC3 2887 KSABCT DS    CL1                       SAVE AREA FOR BUFFER COUNT
    0CC4                  0CC4 2888 KSASAV DS    CL1                       SAVE FOR OVERFLOW OF BUFFER
    0CC5                  0CC5 2889 KSASP2 DS    CL1                       PART2 OF SPLIT READ
    1268                       2890        ORG   KSAEND
                               2891 *      $DL2P
```

```
2893+***********************************************************************
2894+*  5703-XM1  COPYRIGHT IBM CORP 1970                                  *
2895+*            REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE. 120-2083      *
2896+*                                                                     *
2897+***********************************************************************
2898+*STATUS -                                                             *
2899+*   VERSION 1 MODIFICATION 0                                          *
2900+*                                                                     *
2901+*FUNCTION                                                             *
2902+*   * DL2ICS CONVERTS A RELATIVE DISK ADDRESS TO A PHYSICAL DISK      *
2903+*     ADDRESS AND COMBINES IT WITH A BASE ADDRESS PLACED IN DL2RAD    *
2904+*     BY THE CALLER.                                                  *
2905+*   * THE RELATIVE DISK ADDRESS IS A TWO BYTE CYLINDER SECTOR COUNT   *
2906+*     IN THE CALLERS DISK PARAMETER LIST (DPL).                       *
2907+*   * THE COUNT IS A CYLINDER SECTOR DISPLACEMENT FROM THE BASE       *
2908+*     ADDRESS PLACED IN DL2RAD                                        *
2909+*   * DL2ICS IS USED TO PROCESS DATA ON THE FIXED OR REMOVABLE DISK   *
2910+*     ON EITHER DRIVE AND PROVIDES THE INTERFACE TO $DISKN.           *
2911+*   * THE PHYSICAL DISK ADDRESS IS PLACED IN A COPY OF THE USERS DPL  *
2912+*     IN DL2ICS AND A CALL IS MADE TO $DISKN TO PERFORM THE REQUESTED *
2913+*     OPERATION.                                                      *
2914+*                                                                     *
2915+*ENTRY POINTS                                                         *
2916+*   * THE ENTRY IS DL2ICS. THE BASE REGISTER IS SAVED AND RESTORED    *
2917+*     ON RETURN. THE INDEX REGISTER IS NOT USED.                      *
2918+*   * THE FORMAT OF THE CALLING SEQUENCE IS AS FOLLOWS:               *
2919+*        B   DL2ICS                                                   *
2920+*        DC  AL2(PARMLT)                                              *
2921+*     WHERE PARMLT IS THE ADDR OF THE PARAMETER LIST TO BE PROCESSED. *
2922+*                                                                     *
2923+*INPUT                                                                *
2924+*   * THE INPUT IS A TWO BYTE BASE DISK ADDRESS PLACED IN             *
2925+*     DL2RAD AND A SIX BYTE DPL. THE SAME FORMAT AS THE DPL FOR       *
2926+*     $DISKN EXCEPT FOR THE DISK ADDRESS WHICH IS A RELATIVE CYLINDER *
2927+*     AND SECTOR DISPLACEMENT FROM THE BASE ADDRESS IN DL2RAD.        *
2928+*                                                                     *
2929+*OUTPUT                                                               *
2930+*     NONE.                                                           *
2931+*                                                                     *
2932+*EXTERNAL REFERENCES                                                  *
2933+*     $DISKN - ENTRY TO PHYSICAL DISK ROUTINE IS THE SYSTEM NUCLEUS.  *
2934+*                                                                     *
2935+*EXITS, NORMAL                                                        *
2936+*     NORMAL - EXIT IS TO THE FIRST INSTRUCTION FOLLOWING THE POINTER *
2937+*     TO THE DPL. THE BASE REGISTER IS RESTORED. THE RETURN ADDRESS   *
2938+*     IS THE ADDRESS RECALL REGISTER (ARR) +2.                        *
2939+*                                                                     *
2940+*EXITS, ERROR                                                         *
2941+*     NONE                                                            *
2942+*                                                                     *
2943+*TABLES/WORK AREAS                                                    *
2944+*   * THE CONSTANTS AND WORK AREAS RESIDE AT THE END OF THE EXECUTABLE*
2945+*     CODE AND ARE REFERENCED BY A DISPLACEMENT RELATIVE TO THE VALUE *
2946+*     IN INDEX REGISTER 1 (@BR).                                      *
2947+*   * DL2SEC AND DL2SAD ARE EQUATED TO OPERAND LOCATIONS IN THE       *
2948+*     EXECUTABLE CODE TO ELIMINATE EXCESS WORKING STORAGE.            *
```

```
                               2949+*                                                                     *
                               2950+*ATTRIBUTES                                                           *
                               2951+*   * DL2ICS IS REUSABLE                                              *
                               2952+*                                                                     *
                               2953+*CHARACTER CODE DEPENDENCY                                            *
                               2954+*     THE OPERATION OF THIS MODULE DOES NOT DEPEND UPON A PARTICULAR  *
                               2955+*     INTERNAL REPRESENTATION OF THE EXTERNAL CHARACTER SET.          *
                               2956+*                                                                     *
                               2957+*NOTES                                                                *
                               2958+*   ERROR PROCEDURES                                                  *
                               2959+*      NONE                                                           *
                               2960+*                                                                     *
                               2961+*   REGISTER USAGE                                                    *
                               2962+*      INDEX REGISTER 1 (@BR) IS SAVED AND RESTORED. THIS REGISTER IS *
                               2963+*      USED DURING EXECUTION. REGISTER 2 (@BR) IS NOT USED.           *
                               2964+*                                                                     *
                               2965+*   SAVED/RESTORED AREAS                                              *
                               2966+*      NONE                                                           *
                               2967+*                                                                     *
                               2968+*   MODIFICATION CONSIDERATIONS                                       *
                               2969+*      NONE                                                           *
                               2970+*                                                                     *
                               2971+*   REQUIRED MODULES                                                  *
                               2972+*     @SYSEQ - COMMON SYSTEM EQUATES.                                 *
                               2973+*     @FXDEQ - SYSTEM NUCLEUS ADDRESSES AND INDICATORS VALUES EQUATES *
                               2974+*                                                                     *
                               2975+*   OTHER                                                             *
                               2976+*      DL2ICS MAY BE USED TO CONVERT THE DISK ADDRESS ONLY AND NOT TO *
                               2977+*      CALL $DISKN IF THE USER MOVES A UCB CODE TO DL2SWH.            *
                               2978+*      THIS OPTION IS NOT STANDARD USAGE.                             *
                               2979+***********************************************************************
                          126C 2980+       USING DL2000,@BR                ESTABLISH ADDRESSABILITY
                               2981+*
                          0001 2982+DL2E01 EQU   X'01'                     FIELD LENGTH OF 1
                          0002 2983+DL2E02 EQU   X'02'                     FIELD LENGTH OF 2
                          0018 2984+DL2E18 EQU   X'18'                     HEX TRACK SECTOR COUNT
                          0060 2985+DL2E60 EQU   X'60'                     PHYSICAL SECTOR COUNT
                          0083 2986+DL2TSD EQU   X'83'                     MASK OFF TRACK SPINDLE DISK
                          007C 2987+DL2E7C EQU   X'7C'                     MASK OUT SECTOR COUNT
                          1268 2988+DL2ICS EQU   *                         ENTRY POINT
    1268 34 01 12E9            2989+       ST    DL2900+@OP1,@BR           SAVE OLD BASE
                          126C 2990+DL2000 EQU   *                         START PROCESSING
    126C C2 01 126C            2991+       LA    DL2000,@BR                SET BASE ADORESS
    1270 76 08 8A              2992+       A     DL2C01(,@BR),@ARR         BUMP TO RIGHT BYTE OF ADDR
    1273 74 08 14              2993+       ST    DL2001+@DOP2(,@BR),@ARR   ADDR OF PARAM
    1276 76 08 8A              2994+       A     DL2C01(,@BR),@ARR         BUMP TO RETURN ADDR
    1279 74 08 81              2995+       ST    DL2910+@OP1(,@BR),@ARR    SAVE RETURN ADDR
                               2996+*
    127C 4C 01 1D 0000         2997+DL2001 MVC   DL2002+@DOP2(@DADDR,@BR),*-* SETUP ADDR OF DPL
    1281 5E 01 1D 8C           2998+       ALC   DL2002+@DOP2(@CADDR,@BR),DL2C05(,@BR) DUMP TO RIGHT END
    1285 4C 05 92 0000         2999+DL2002 MVC   DL2DPL(@DPLNG,@BR),*-*    MOVE USER DPL TO WORK AREA
    128A 5F 00 8F 86           3000+DL2005 SLC   DL2LST+@DSAD(DL2E01,@BR),DL2C48(,@BR) ADJUST SCTR/CYL
    128E F2 82 07              3001+       JM    DL2006                    GO TO RESTORE TO CONTINUE
    1291 5E 00 8E 8A           3002+       ALC   DL2LST+@DCYL(DL2E01,@BR),DL2C01(,@BR) BUMP CYLINDER COUNT
    1295 D0 87 1E              3003+       B     DL2005(,@BR)              BACK FOR NEXT CYLINDER
    1298 5E 00 8F 86           3004+DL2006 ALC   DL2LST+@DSAD(DL2E01,@BR),DL2C48(,@BR) RESTORE POSITIVE
```

```
                               3005+*
                               3006+*      GET THE LOGICAL SECTOR FROM THE DPL. THE NUMBER IS LEFT ADJUSTED
                               3007+*      TO COMAE IT MTN THE POINTER ESTABLISHED PRIOR TO AN ENTRY.
    129C 5C 00 1D 8F           3008+       MVC   DL2SEC(DL2E01,@BR),DL2LST+@DSAD(,@BR) GET SECTOR NUMBER
    12A0 7C 00 8F              3009+       MVI   DL2LST+@DSAD(,@BR),@ZERO  CLEAR SECTOR BYTE
                               3010+*
                               3011+*      MOVE THE RELATIVE START TO THE DFL
                               3012+*
    12A3 5E 01 8F 94           3013+       ALC   DL2LST+@DSAD(DL2E02,@BR),DL2RAD(,@BR) DL2RAD TO DPL
    12A7 7D 18 1D              3014+       CLI   DL2SEC(,@BR),DL2E18       IS COUNT OVER A TRACK
    12AA F2 82 08              3015+       JL    DL2008                    NO GO CHANGE A PHYSICAL ADOR
    12AD 5E 01 8F 85           3016+       ALC   DL2LST+@DSAD(DL2E02,@BR),DL2K80(,@BR) BUMP TRACK VALUE
    12B1 5F 00 1D 88           3017+       SLC   DL2SEC(1,@BR),DL2K18(,@BR) DECR BY TRACK VALUE
    12B5 5E 00 1D 1D           3018+DL2008 ALC   DL2SEC(1,@BR),DL2SEC(,@BR) SHIFT LEFT 1
    12B9 5E 00 1D 1D           3019+       ALC   DL2SEC(1,@BR),DL2SEC(,@BR) SHIFT LEFT
    12BD 5C 00 14 8F           3020+       MVC   DL2SAD(DL2E01,@BR),DL2LST+@DSAD(,@BR) GET SECTOR ADDRESS
                               3021+*
                               3022+*      ZERO OUT THE SECTOR COUNT AND LEAVE THE DISK. SPINDLE AND
                               3023+*      TRACK BITS AS IS TO BE RE INSERTED AFTER THE SECTOR HAS BEEN
                               3024+*      LOCATES.
                               3025+*
    12C1 7B 7C 8F              3026+       SBF   DL2LST+@DSAD(,@BR),DL2E7C TURN OFF
    12C4 7B 83 14              3027+       SBF   DL2SAD(,@BR),DL2TSD       OFF TRACK SPINDLE DISK
    12C7 5E 00 14 1D           3028+       ALC   DL2SAD(DL2E01,@BR),DL2SEC(,@BR) COMBINE SECTOR COUNTS
    12CB 7D 60 14              3029+DL2010 CLI   DL2SAD(,@BR),DL2E60       TEST IF TRACK CROSSED
    12CE F2 82 08              3030+       JL    DL2100
                               3031+*
                               3032+*      INCREMENT TRACK BIT. OVERFLOW INTO THE CYLINDER COUNT.
                               3033+*
    12D1 5E 01 8F 85           3034+       ALC   DL2LST+@DSAD(DL2E02,@BR),DL2K80(,@BR)
    12D5 5F 00 14 83           3035+       SLC   DL2SAD(1,@BR),DL2K60(,@BR) DECR BY TRACK VALUE
                               3036+*
    12D9 5E 00 8F 14           3037+DL2100 ALC   DL2LST+@DSAD(1,@BR),DL2SAD(,@BR) INSERT SECTOR COUNT
                               3038+*
    12DD F2 80 06              3039+DL2110 JC    DL2900,@NOP               CONVERSION SWITCH
                          12DE 3040+DL2SWH EQU   DL2110+@Q                 ADDR OF Q CODE FOR SWITCH
    12E0 C0 87 0025            3041+       B     $DISKN                    GO PROCESS I/O
    12E4 12F9             12E5 3042+       DC    AL2(DL2LST)               ADDRESS OF DPL
    12E6 C2 01 0000            3043+DL2900 LA    *-*,@BR                   RESTORE CALLERS BASE
    12EA C0 87 0000            3044+DL2910 B     *-*
                               3045+**********************************************************************
                               3046+*      CONSTANTS
                               3047+**********************************************************************
    12EE 0060             12EF 3048+DL2K60 DC    XL2'0060'                 SECTOR COUNT OF 24 LEFT ADJUSTD
    12F0 0080             12F1 3049+DL2K80 DC    XL2'0080'                 BIT FOR INCREMENTING TRACK
    12F2 30               12F2 3050+DL2C48 DC    IL1'48'                   CYLINDER VALUE FOR 1 DISK
    12F3 0018             12F4 3051+DL2K18 DC    XL2'18'                   HEX SECTORS PER TRACK
    12F5 0001             12F6 3052+DL2C01 DC    IL2'1'                    CONSTANT FOR REGISTER MODE
    12F7 0005             12F8 3053+DL2C05 DC    IL2'5'                    DISP TO RIGHT END OF DPL
                               3054+**********************************************************************
                               3055+*      WORK AREA
                               3056+**********************************************************************
                          12F9 3057+DL2LST EQU   *                         LIST HIGH END
    12F9                  12FE 3058+DL2DPL DS    CL(@DPLNG)                WORKING DPL
                          12FB 3059+DL2PHY EQU   DL2LST+@DSAD              POINTER TO PHYSICAL DADDR
                          1280 3060+DL2SAD EQU   DL2001+@DOP2              SAVE SECTOR BYTE FROM DPI
                          1289 3061+DL2SEC EQU   DL2002+@DOP2              WORKING SECTOR ADDRESS FIELD
    12FF                  1300 3062+DL2RAD DS    CL(@DADDR)                USER RELATIVE STARTING ADDR.
                          1301 3063+DL2END EQU   *                         END OF DL2ICS
                               3064+***                       END OF DL2ICS                            ***
                          1301 3065 KSAIOB EQU   DL2END                    BUFFER CADDR
                               3066 *      $RCHF
```

```
3068+***********************************************************************
3069+*  5703-XM1 COPYRIGHT IBM CORP. 1970                                  *
3070+*           REFER TO INSTRUCTIONS ON COPY RIGHT NOTICE, 120-2083      *
3071+*                                                                     *
3072+***********************************************************************
3073+*STATUS                                                               *
3074+*   VERSION 1 MODIFICATION 0                                          *
3075+*                                                                     *
3076+*FUNCTION                                                             *
3077+*   * SRCHFN SEARCHES A USER DIRECTORY FOR A SPECIFIED FILENAME. IT   *
3078+*     IS ASSUMED THAT THE DIRECTORY TO BE SEARCHED HAS BEEN READ INTO *
3079+*     CORE AT SMUDBI IN TSMLES.  IF THE DIRECTORY IS LINKED TO AN     *
3080+*     ADDITIONAL BLOCK IT IS READ IN TO THE SECONDARY BUFFER WHILE    *
3081+*     THE PRIMARY BLOCK IS SEARCHED.                                  *
3082+*   * THE ADDRESS OF THE ENTRY OR THE ADDRESS FOR A NEW ENTRY IS      *
3083+*     PLACED IN SMUDEA. THE ADDRESS OF THE ACTIVE DIRECTORY IS PLACED *
3084+*     IN SMUDBA.  IF THE NAME WAS NOT FOUND SMIFNE IS SET TO 1 IN     *
3085+*     SMIND1.  IF THE NAME WAS FOUND THE INDICATOR IS SET TO 0.       *
3086+*                                                                     *
3087+*ENTRY POINTS                                                         *
3088+*   SRCHFN - ENTRY TO SEARCH FOR A FILENAME. THE CALLING SEQUENCE     *
3089+*            IS AS FOLLOWS:                                           *
3090+*               B    SRCHFN                                           *
3091+*                                                                     *
3092+*INPUT                                                                *
3093+*   THE USER DIRECTORY BLOCK MUST BE READ INTO SMUDB1 IN TSMLES.      *
3094+*   THE NAME OF THE ENTRY TO SEARCH FOR MUST BE IN SMFNAM IN TSMLES   *
3095+*                                                                     *
3096+*OUTPUT                                                               *
3097+*   * IF THE FILE NAME IS FOUND THE ADDRESS OF THE ENTRY IS SET IN    *
3098+*     SMUDEA. THE ADDRESS OF THE BUFFER CONTAINING THE ENTRY IS IN    *
3099+*     SMUDBA, AND THE INDICATOR BIT SMIFNE IN SMIND1 IS SET TO 0.     *
3100+*   * IF THE FILE NAME WAS NOT FOUND SMUDEA CONTAINS THE ADDRESS OF   *
3101+*     WHERE THE NEXT ENTRY MAY BE MADE IN THE DIRECTORY. SMUDBA       *
3102+*     CONTAINS THE ADDRESS OF THE BUFFER CONTAINING THE LAST BLOCK,   *
3103+*     AND SMIFNE IS SET TO 1 IN SMIND1.                               *
3104+*   * SMUDEA CONTAINS THE ADDRESS OF THE LEFTMOST BYTE OF THE FIELD,  *
3105+*   * THE ERROR CODE FOR FILE NOT FOUND IS ALWAYS MOVED TO $CAERR,    *
3106+*                                                                     *
3107+*EXTERNAL REFERENCES                                                  *
3108+*        $CAERR - LOCATION OF ERROR CODE INDICATOR.                   *
3109+*        $DISKN - ENTRY TO DISK IOCS.                                 *
3110+*        $WAITF - ADDRESS OF COMMON I/O WAIT FUNCTION.                *
3111+*        DL2ICS - ENTRY TO DISK LOGICAL IOCS.                         *
3112+*        SMFNAM - ADDRESS OF FILENAME SAVE AREA                       *
3113+*        SMUDEA - ADDRESS OF USER DIRECTORY ENTRY ADDRESS.            *
3114+*        SMUDBA - ADDRESS OF USER DIRECTORY BUFFER ADDRESS.           *
3115+*        SMDAAD - LOCATION OF RELATIVE DISK ADDRESS OF ACTIVE BUFFER. *
3116+*        SMIFNE - VALUE OF NOT FOUND INDICATOR.                       *
3117+*        SMIND1 - LOCATION INDICATOR 1.                               *
3118+*        SMUDB1 - ADDRESS OF DIRECTORY BLOCK BUFFER.                  *
3119+*        SMUDB2 - ADDRESS OF DIRECTORY BLOCK BUFFER.                  *
3120+*                                                                     *
3121+*EXITS, NORMAL                                                        *
3122+*   THE REGISTER @BR @XR ARE RESTORED AND THE EXIT IS TO THE          *
3123+*   ADDRESS SAVED FROM THE @ARR REGISTER.                             *
```

```
                               3124+*                                                                     *
                               3125+*EXITS, ERROR                                                         *
                               3126+*   NONE.                                                             *
                               3127+*                                                                     *
                               3128+*TABLES/WORKAREAS                                                     *
                               3129+*   NONE                                                              *
                               3130+*                                                                     *
                               3131+*ATTRIBUTES                                                           *
                               3132+*   RELOCATABLE                                                       *
                               3133+*                                                                     *
                               3134+*CHARACTER CODE DEPENDENCY                                            *
                               3135+*   CHARACTER CODE DEPENDENCY CLASS - C                               *
                               3136+*   THE OPERATION OF THIS MODULE DEPENDS UPON AN INTERNAL REPRESENTA- *
                               3137+*   TION OF THE EXTERNAL CHARACTER SET WHICH IS EQUIVALENT TO THE ONE *
                               3138+*   USED AT ASSEMBLY TIME.  THE CODING HAS BEEN ARRANGED SO THAT RE-  *
                               3139+*   DEFINITION OF CHARACTER CONSTANTS, BY REASSEMBLY, WILL RESULT IN  *
                               3140+*   A CORRECT MODULE FOR THE NEW DEFINITIONS.                         *
                               3141+*                                                                     *
                               3142+*NOTES                                                                *
                               3143+*   ERROR PROCEDURES                                                  *
                               3144+*      NONE                                                           *
                               3145+*                                                                     *
                               3146+*   REGISTER USAGE                                                    *
                               3147+*      @BR AND @XR ARE SAVED ON ENTRY AND RESTORED AT EXIT.           *
                               3148+*      @ARR IS USED AS THE RETURN ADDRESS.                            *
                               3149+*                                                                     *
                               3150+*   SAVED/RESTORED AREAS                                              *
                               3151+*      NONE                                                           *
                               3152+*                                                                     *
                               3153+*   MODIFICATION CONSIDERATIONS                                       *
                               3154+*      NONE                                                           *
                               3155+*                                                                     *
                               3156+*   REQUIRED MODULES                                                  *
                               3157+*      @SYSEQ - SYSTEM SOFTWARE EQUATES.                              *
                               3158+*      @DIREQ - LIBRARY DIRECTORY EQUATES.                            *
                               3159+*      @FXDEQ - SYSTEM NUCLEUS EQUATES.                               *
                               3160+*      DL2ICS - LOGICAL DISK IOCS.                                    *
                               3161+*      TSMLES - DATA MANAGEMENT COMMUNICATIONS AREA.                  *
                               3162+*                                                                     *
                               3163+*   OTHER                                                             *
                               3164+*      NONE                                                           *
                               3165+***********************************************************************
```

```
                          1301 3167+SRCHFN EQU   *                         ENTRY TO SEARCH FILENAME
    1301 34 01 138B            3168+       ST    SRC900+@OP1,@BR           SAVE BASE REGISTER
                          1305 3169+       USING SRC010,@BR
    1305 C2 01 1305            3170+SRC010 LA    SRC010,@BR                SET BASE ADDR
    1309 74 02 8A              3171+       ST    SRC910+@OP1(,@BR),@XR     SAVE INDEX REG
    130C 74 08 8E              3172+       ST    SRC920+@OP1(,@BR),@ARR    SAVE RETURN ADDR
    130F 3C 24 03CD            3173+       MVI   $CAERR,@@E211             FILE NOT FOUND
    1313 5C 01 9B A1           3174+       MVC   SRCBF1(@CADDR,@BR),SRCBA1(,@BR) INITIALIZE OLF POINTER
    1317 5C 01 9D A3           3175+       MVC   SRCBF2(@CADDR,@BR),SRCBA2(,@BR) ALTERNATE BUFFER
    131B 5C 01 9F 9B           3176+       MVC   SRCACT(@CADDR,@BR),SRCBF1(,@BR) SET ACTIVE BUFFER

    131F C0 87 0025            3178+SRC020 B     $DISKN                    WAIT FOR USER BLOCK
    1323 057F             1324 3179+       DC    AL2($WAITF)               WAIT OP DPL
                               3180+*
    1325 7C 87 5E              3181+       MVI   SRC055+@Q(,@BR),@UCB      RESET NOP FOR LINKED DIRCTY
    1328 75 02 9F              3182+       L     SRCACT(,@BR),@XR          PICKUP POINTER TO ACTIVE BUFFER
                               3183+*
                               3184+*            BLOCK LINK SHOULD ALWAYS BE GREATER THAN 1 IF IT IS
                               3185+*            PRESENT. IF NOT THE LINK BYTE SHOULD BE ZERO.
                               3186+*
    132B 9D 01 03 A6           3187+       CLC   ##DUHB(@DADDR,@XR),SRCC01(,@BR) TEST LIVE FIELD
    132F F2 82 11              3188+       JL    SRC030                    JUMP NOT LINKED
    1332 5C 01 AC 9D           3189+       MVC   SRCBFR(@DADDR,@BR),SRCBF2(,@BR) GET ALTERNATE BUFFER ADDR
    1336 6C 01 A9 03           3190+       MVC   SRCDAD(@DADDR,@BR),##DUHB(,@XR) SET LINK TO MEXT BLOCK
    133A C0 87 1268            3191+       B     DL2ICS                    READ NEXT BLOCK
    133E 13AC             133F 3192+       DC    AL2(SRCDPL)               POINTER TO DPL
                               3193+*
    1340 7C 80 5E              3194+       MVI   SRC055+@Q(,@BR),@NOP      SET SWITCH FOR LINKED BLOCK
    1343 6C 00 A4 04           3195+SRC030 MVC   SRCCNT(1,@BR),##DUHC(,@XR) GET ENTRY COUNT
    1347 E2 02 0C              3196+       LA    ##DUEI(,@XR),@XR          BUMP TO FIRST ENTRY
    134A 7D 00 A4              3197+       CLI   SRCCNT(,@BR),@ZERO        IS STARTING COUNT ZERO ?
    134D D0 81 5D              3198+       BE    SRC055(,@BR)              YES, RETURN NOT FOUND
    1350 8D 07 07 1B54         3199+SRC035 CLC   ##DUEN(##LUEN,@XR),SMFNAM LOOK AT ENTRY
    1355 F2 81 1C              3200+       JE    SRC040                    JUMP IF THE NAME IS FOUND
    1358 E2 02 32              3201+       LA    ##LUE(,@XR),@XR           BUMP THE POINTER FOR NEXT ENTRY
    135B 5F 00 A4 A6           3202+       SLC   SRCCNT(1,@BR),SRCC01(,@BR) DECR ENTRY COUNTER
    135F D0 01 4B              3203+       BNE   SRC035(,@BR)              BACK TO TEXT NEXT ENTRY
    1362 F2 00 2F              3204+SRC055 JC    SRC060,*-*                LINK SWITCH
    1365 5C 01 9B 9D           3205+       MVC   SRCBF1(@CADDR,@BR),SRCBF2(,@BR) SWITCH BUFFERS
    1369 5C 01 9D 9F           3206+       MVC   SRCBF2(@CADDR,@BR),SRCACT(,@BR) *
    136D 5C 01 9F 9B           3207+       MVC   SRCACT(@CADDR,@BR),SRCBF1(,@BR) SET ACTIVE BUFFER
    1371 D0 87 1A              3208+       B     SRC020(,@BR)              GO BACK TO NEXT BUFFER
                               3209+*
                               3210+*            FILENAME HAS BEEN FOUND.
                               3211+*
    1374 34 02 1B56            3212+SRC040 ST    SMUDEA,@XR                SAVE ENTRY ADDR
    1378 3B 80 1B3E            3213+       SBF   SMIND1,SM1FNE             TURN OFF NOT FOUND INDICATOR
    137C 75 02 9F              3214+SRC050 L     SRCACT(,@BR),@XR          GET CADDR OF ACTIVE BUFFER
    137F 34 02 1B5A            3215+       ST    SMUDBA,@XR                SAVE CADDR IN SMALES
    1383 2C 01 1B6C 01         3216+       MVC   SMDAAD,##DUHA(@DADDR,@XR) SAVE RDADDR OF ACTIVE DIRCTY
    1388 C2 01 0000            3217+SRC900 LA    *-*,@BR                   RESTORE CALLERS BASE
    138C C2 02 0000            3218+SRC910 LA    *-*,@XR                   RESTORE INDEX
    1390 C0 87 0000            3219+SRC920 B     *-*                       RETURN

                               3221+*
                               3222+*            FILENAME WAS NOT FOUND. SAVE ADDR FOR NEXT ENTRY AND
                               3223+*            SET THE INDICATOR.
                               3224+*
    1394 34 02 1B56            3225+SRC060 ST    SMUDEA,@XR                SAVE ADDR FOR NEXT ENTRY
    1398 3A 80 1B3E            3226+       SBN   SMIND1,SM1FNE             TURN ON NOT FOUND INDICATOR
    139C D0 87 77              3227+       B     SRC050(,@BR)              GO TO RETURN
```

```
                               3229+*
                               3230+*            CONSTANTS AND WORK AREA
                               3231+*
    139F                  13A0 3232+SRCBF1 DS    CL(@CADDR)                WORK AREA PRIMARY BUFFER ADDR
    13A1                  13A2 3233+SRCBF2 DS    CL(@CADDR)                WORK AREA SECONDARY BUFFER ADDR
    13A3                  13A4 3234+SRCACT DS    CL(@CADDR)                SAVE AREA FOR ACTIVE BUFFER
    13A5 1B6D             13A6 3235+SRCBA1 DC    AL2(SMUDB1)               ADDRESS OF USED DIRCTY BLUFFER 1
    13A7 1D6D             13A8 3236+SRCBA2 DC    AL2(SMUDB2)               ADDRESS OF DIRCTY BUFFER 2
    13A9                  13A9 3237+SRCCNT DS    CL1                       WORK AREA FOR ENTRY COUNT
    13AA 0001             13AB 3238+SRCC01 DC    IL2'1'                    CONSTANT TO DECR ENTRY COUNT
                          13AC 3239+SRCDPL EQU   *                         DEFINE LEFT END OF DPL
    13AC 01               13AC 3240+SRCGET DC    AL1(@DGET)                READ OP CODE
    13AD                  13AE 3241+SRCDAD DS    CL(@DADDR)                RELATIVE ADDR OF BLOCK
    13AF 02               13AF 3242+SRCSCT DC    AL1(##LU)                 SECTOR COUNT FOR BLOCK
    13B0                  13B1 3243+SRCBFR DS    CL(@CADDR)                BUFFER ADDR OF BLOCK
                               3244+***                       END OF SRCHFN                             ***
                               3245 *      $TORI
                               3247+***********************************************************************
                               3248+*  5703-XM1 COPYRIGHT IBM CORP, 1970                                  *
                               3249+*           REFER TO INSTRUCTIONS ON COPY RIGHT NOTICE, 120-2083     *
                               3250+*                                                                     *
                               3251+***********************************************************************
                               3252+*STATUS                                                               *
                               3253+*   VERSION 1 MODIFICATION 0                                          *
                               3254+*                                                                     *
                               3255+*FUNCTION                                                             *
                               3256+*   * STORIN WILL INSERT AN ENTRY IN THE NULL DIRECTORY. IF THE ENTRY *
                               3257+*     IS CONTIGUOUS WITH ANY OTHER ENTRY ALREADY IN THE DIRECTORY, IT *
                               3258+*     IS COMBINED WITH THAT ENTRY, IF THE ENTRY IS CONTIGUOUS TO TWO  *
                               3259+*     ENTRIES THE THREE ENTRIES ARE COMBINED INTO ONE AND THE         *
                               3260+*     DIRECTORY IS COMPRESSED,                                        *
                               3261+*   * IF THE ENTRY IS NOT CONTIGUOUS TO ANY OTHER ENTRY IT IS ADDED   *
                               3262+*     TO THE END OF THE DIRECTORY OR INSERTED IN SEQUENCE.            *
                               3263+*   * IF THE DIRECTORY IS FULL THE INDICATOR IN SMIND1 IS SET AND     *
                               3264+*     THE RETURN TAKEN,                                               *
                               3265+*                                                                     *
                               3266+*ENTRY POINTS                                                         *
                               3267+*     STORIN - ENTRY TO STORE A NULL ENTRY IN THE DIRECTORY. @BR      *
                               3268+*              AND @XR ARE SAVED AND RESTORED ON RETURN.  THE         *
                               3269+*              CALLING SEQUENCE IS AS FOLLOWS:                        *
                               3270+*                    B STORIN                                         *
                               3271+*              RETURN IS TO THE FIRST INSTRUCTION FOLLOWING THE       *
                               3272+*              BRANCH TO STORIN,                                      *
                               3273+*                                                                     *
                               3274+*INPUT                                                                *
                               3275+*   * THE ADDRESS OF THE LEFT BYTE OF THE ENTRY TO BE MADE IN THE     *
                               3276+*     DIRECTORY MUST BE IN SMNETD,                                    *
                               3277+*   * THE ADDRESS OF THE NULL DIRECTORY MUST BE IN SMNDBA.            *
                               3278+*                                                                     *
                               3279+*OUTPUT                                                               *
                               3280+*   NONE.                                                             *
                               3281+*                                                                     *
                               3282+*EXTERNAL REFERENCES                                                  *
                               3283+*      SMNETD - LOCATION OF THE ADDRESS OF THE ENTRY                  *
                               3284+*      SMNDBA - LOCATION OF NULL DIRECTORY BUFFER ADDRESS.            *
                               3285+*      SMIND1 - LOCATION OF INDICATOR BYTE IN TSMLES.                 *
                               3286+*      SM1STN - VALUE OF FULL DIRECTORY INDICATOR.                    *
                               3287+*                                                                     *
                               3288+*EXITS, NORMAL                                                        *
                               3289+*   RETURN IS TO THE LOCATION POINTED TO BY THE @ARR. IF THE ENTRY   *
                               3290+*   WAS MADE SM1STN IS 0, IF THE DIRECTORY IS FULL AND THE ENTRY      *
                               3291+*   CAN NOT BE MADE SM1STN IN SMIND1 IS SET TO 1.                     *
                               3292+*                                                                     *
                               3293+*EXITS, ERROR                                                         *
                               3294+*   NONE                                                              *
                               3295+*                                                                     *
                               3296+*TABLES/NORKAREAS                                                     *
                               3297+*   NONE                                                              *
                               3298+*                                                                     *
                               3299+*ATTRIBUTES                                                           *
                               3300+*   RELOCATABLE, REUSABLE                                             *
                               3301+*                                                                     *
                               3302+*CHARACTER CODE DEPENDENCY                                            *
                               3303+*   THE OPERATION OF THIS MODULE DEPENDS UPON AS INTERNAL             *
                               3304+*   REPRESENTATION OF THE EXTERNAL CHARACTER SET WHICH IS EQUIVALENT  *
                               3305+*   TO THE USED AT ASSEMBLY TIME, THE CODING HAS BEEN ARRANGED SO     *
                               3306+*   THAT REDEFINITION OF THE CHARACTER CONSTANTS, BY REASSEMBLY, WILL *
                               3307+*   RESULT IN A CORRECT MODULE FOR THE NEW DEFINITIONS.               *
                               3308+*                                                                     *
                               3309+*NOTES                                                                *
                               3310+*   ERROR PROCEDURES                                                  *
                               3311+*      NONE                                                           *
                               3312+*                                                                     *
                               3313+*   REGISTER USAGE                                                    *
                               3314+*      * @BR AND @XR ARE SAVED AND RESTORED ON RETURN.                *
                               3315+*        @ARR IS SAVED IN THE BRANCH TO RETURN.                       *
                               3316+*      * @BR IS USED AS A BASE REGISTER DURING EXECUTION.             *
                               3317+*        @XR IS USED A POINTER TO THE NULL DIRECTORY.                 *
                               3318+*                                                                     *
                               3319+*   SAVED/RESTORED AREAS                                              *
                               3320+*      NONE                                                           *
                               3321+*                                                                     *
                               3322+*   MODIFICATION CONSIDERATIONS                                       *
                               3323+*      TO CALCULATE THE END OF THE NULL DIRCTY STORIN MULTIPLIES THE  *
                               3324+*      NUMBER OF ENTRIES BY SIX,  IF THE LENGTH OF THE NULL ENTRY IS  *
                               3325+*      CHANGED, THIS CODING MUST BE UPDATED.                          *
                               3326+*                                                                     *
                               3327+*   REQUIRED MODULES                                                  *
                               3328+*        @SYSEQ - SYSTEM SOFTWARE EQUATES                             *
                               3329+*        @DIREQ - LIBRARY DIRECTORY EQUATES                           *
                               3330+*        TSMLES - DATA MANAGEMENT COMMUNICATIONS AREA                 *
                               3331+*                                                                     *
                               3332+*   OTHER                                                             *
                               3333+*      NONE                                                           *
                               3334+***********************************************************************
```

```
                          13B2 3336+STORIN EQU   *                         ENTRY TO STORE IN NULL DIRCTY
                          143B 3337+       USING STOR30,@BR                BASE
    13B2 34 01 14C1            3338+       ST    STOR90+@OP1,@BR           SAVE BASE
                          0001 3339+STORE1 EQU   1                         Q CODE VALUE
    13B6 C2 01 143B            3340+       LA    STOR30,@BR                LOAD BASE ADDR
    13BA 74 02 8A              3341+       ST    STOR95+@OP1(,@BR),@XR     SAVE INDEX
    13BD 74 08 8E              3342+       ST    STOR99+@OP1(,@BR),@ARR    SAVE RETURN
                               3343+*
                               3344+*            INITALIZE POINTERS AND COUNTERS
                               3345+*
    13C0 7C 80 01              3346+       MVI   STOR31(,@BR),@NOP         NO PREVIOUS ENTRY SWITCH
    13C3 5C 01 06 A5           3347+       MVC   STOR35+@OP1(@CADDR,@BR),STO70A(,@BR) NEW ENTRY SWITCH
    13C7 35 02 1B62            3348+       L     SMNETD,@XR                GET NEW ENTRY ADDR
    13CB 6C 03 92 03           3349+       MVC   STORWE(STOENL,@BR),##DNEF(,@XR) MOVE INTO WORKAREA
    13CF 35 02 1B6A            3350+       L     SMNDBA,@XR                PICKUP POINTER TO BUFFER AREA
    13D3 74 02 35              3351+       ST    STOR45(,@BR),@XR          SAVE BUFFER ADDR
    13D6 74 02 73              3352+       ST    STO048+@OP1(,@BR),@XR     SAVE BUFFER POINTER
    13D9 6C 00 A3 00           3353+       MVC   STOENC(##LAHC,@BR),##DPHC(,@XR) COUNT TO NEW ENTRY
    13DD 6C 00 9D 00           3354+       MVC   STORWC(##LAHC,@BR),##DPHC(,@XR) PICKUP ENTRY COUNT
                               3355+*
                               3356+*            TEST ENTRY COUNT FOR MAX ENTRIES OR ZERO ENTRIES
                               3357+*
    13E1 7D 2A 9D              3358+       CLI   STORWC(,@BR),##MNHM       TEST MAX ENTRY COUNT
    13E4 F2 81 13              3359+       JE    STOR10                    GO SET SWITCH NO NEW ENTRIES
    13E7 7C 80 EA              3360+       MVI   STOR70+@Q(,@BR),@NOP      SET SWITCH TO ALLOW ENTRY
    13EA 7D 00 9D              3361+       CLI   STORWC(,@BR),@ZERO        TEST IF DIRCTY EMPTY
    13ED F2 01 0D              3362+       JNE   STOR14                    GO COMPARE ENTRIES
                               3363+*
    13F0 BC 01 00              3364+       MVI   ##DNHC(,@XR),STORE1       INITIALIZE COUNTER
                               3365+*
                               3366+*            MOVE THE ENTRY INTO THE DIRCTY
                               3367+*
    13F3 9C 05 09 94           3368+       MVC   ##LNH+##DNER(##LNE,@XR),STORWE+##LNEZ(,@BR)
    13F7 D0 87 83              3369+       B     STOR90(,@BR)              GO RETURN
                               3370+*
    13FA 7C 87 EA              3371+STOR10 MVI   STOR70+@Q(,@BR),@UCB      SWITCH NO NEW ENTRIES
    13FD E2 02 04              3372+STOR14 LA    ##DNE1(,@XR),@XR          BUMP TO FIRST ENTRY
    1400 6D 01 90 01           3373+STOR15 CLC   STORWE-##LNEF(@DADDR,@BR),##DNEA(,@XR) COMPARE NEW/DIRCTY
    1404 F2 82 14              3374+       JL    STOR20                    NEW LOWER
    1407 74 02 30              3375+       ST    STORPA(,@BR),@XR          SAVE PREVIOUS ENTRY ADDR
    140A 5F 00 9D 9F           3376+       SLC   STORWC(1,@BR),STORC1(,@BR) DECK ENTRY COUNT
    140E F2 81 7F              3377+       JE    STOR47                    GO SETUP TO CALC CURRENT HIGH
    1411 7C 87 01              3378+       MVI   STOR31(,@BR),@UCB         SET PREVIOUS ENTRY SWITCH
    1414 E2 02 06              3379+       LA    ##LNE(,@XR),@XR           BUMP POINTER TO NEXT ENTRY
    1417 C0 87 1400            3380+       B     STOR15                    BACK FOR NEXT ENTRY
                               3381+*
                               3382+*            FOUND POSSIBLE POSITION FOR NEW ENTRY
                               3383+*
    141B 5C 03 B2 92           3384+STOR20 MVC   STORWK(STOENL,@BR),STORWE(,@BR)
    141F D0 87 C5              3385+       B     STOR60(,@BR)              GO CALC HIGH END
                               3386+*
                               3387+*            TEST IF ADDR OF HIGH END IS CONTIGUOUS TO NEXT ENTRY
                               3388+*
    1422 7C 80 1F              3389+       MVI   STO39A+@Q(,@BR),@NOP      DONT COMBINE
    1425 6D 01 B0 01           3390+       CLC   STORCS(@CADDR,@BR),##DNEA(,@XR) COMPARE ADDR
    1429 F2 01 0F              3391+       JNE   STOR30                    JUMP NOT CONTIGUOUS
    142C 9E 01 03 92               3392+        ALC   ##DNEF(##LNEF,@XR),STORWE(,@BR) ADD NEW COUNT TO ENTRY
    1430 9C 01 01 90               3393+        MVC   ##DNEA(@DADDR,@XR),STORWA(,@BR) MOVE IN NEW ENTRY
                               3394+*
    1434 5C 01 06 54               3395+        MVC   STOR35+@OP1(@CADDR,@BR),STORET(,@BR) GET RETURN CADDR
    1438 7C 87 1F                  3396+        MVI   STO39A+@Q(,@BR),@UCB      ALLOW ENTRIES BE COMBINED
                               3397+*
                               3398+*            TEST IF PREVIOUS ENTRY THAT MAY BE CONTIGUOUS.  TEST IF
                               3399+*            SWITCH ON OR OFF, @UCB IS ON. @NOP IS NO PREVIOUS ENTRY.
                               3400+*
    143B F2 00 04                  3401+STOR30 JC    STOR38,*-*                PREVIOUS ENTRY SWITCH
                          143C 3402+STOR31 EQU   STOR30+@Q
                               3403+*
                               3404+*            IF NEW ENTRY TO BE ADDED GO TO STOR70.  IF NO NEW ENTRY
                               3405+*            GO TO RETURN ROUTINE.
                               3406+*
    143E C0 87 0000                3407+STOR35 B     *-*                       RETURN OR GO MAKE ENTRY
                               3408+*
                               3409+*            SET UP TO CALCULATE HIGH END ADDR OF PREVIOUS ENTRY
                               3410+*            PICK UP THE DISPLACEMENT TO THE PREVIOUS ENTRY
                               3411+*
    1442 5E 01 30 94               3412+STOR38 ALC   STORPA(@CADDR,@BR),STOCLN(,@BR) BUMP TO RIGHT END
    1446 5C 01 13 30               3413+        MVC   STOR39+@DOP2(@CADDR,@BR),STORPA(,@BR)
    144A 4C 03 B2 0000             3414+STOR39 MVC   STORWK(STOENL,@BR),*-*    MOVE PREVIOUS ENTRY TO WORKAREA
                               3415+*
    144F D0 87 C5                  3416+        B     STOR60(,@BR)              CALC HIGH END
                               3417+*
    1452 5D 01 90 B0               3418+        CLC   STORCW(@DADDR,@BR),STORCS(,@BR)
    1456 D0 01 03                  3419+        BNE   STOR35(,@BR)              GO RETURN OR MAKE NEW ENTRY
    1459 F2 00 0C                  3420+STO39A JC    STOR40,*-*                SWITCH FOR COMBINING ENTRIES
                               3421+*
                               3422+*            NEW ENTRY IS CONTIGUOUS TO PREVIOUS ENTRY BUR NOT NEXT
                               3423+*
    145C 5C 01 28 30               3424+        MVC   STO39B+@OP1(@CADDR,@BR),STORPA(,@BR)
    1460 1E 01 0000 92             3425+STO39B ALC   *-*,STORWE(##LNEF,@BR)    NEW COUNT TO PREVIOUS ENTRY
    1465 F2 87 56                  3426+        J     STOR90                    GO RETURN
                               3427+*
                               3428+*            NEW ENTRY HAS FILED A SPACE BETWEEN TO FORMER ENTRIES.
                               3429+*            COMBINE THE THREE ENTRIES INTO THE FIRST ENTRY.
                               3430+*
    1468 2E 01 0000 03             3431+STOR40 ALC   *-*,##DNEF(##LNEF,@XR)    ADD COUNT FIELDS
                          146B 3432+STORPA EQU   STOR40+@OP1
                               3433+*
                               3434+*            PICK UP POINTER TO START OF BUFFER TO DECR COUNT
                               3435+*
    146D 1F 00 0000 9F             3436+STOR46 SLC   *-*,STORC1(##LAHC,@BR)    DECR HEADER ENTRY COUNT
                          1470 3437+STOR45 EQU   STOR46+@OP1               ADDR OF DIRCTY ADDR
    1472 E2 02 05                  3438+        LA    ##DNER(,@XR),@XR          BUMP TO RIGHT END
                               3439+*
    1475 5F 00 9D 9F               3440+        SLC   STORWC(1,@BR),STORC1(,@BR) DECR WORK ENTRY COUNT
    1479 F2 81 42                  3441+        JE    STOR90                    IF LAST ENTRY RETURN
                               3442+*
                               3443+*            SET UP POINTERS TO SQUEEZE UP THE DIRCTY. THE CURRENT
                               3444+*            ENTRY IS DELETED BY OVERLAYING IT WITH THE REMAINDER OF
                               3445+*            THE DIRCTY.
                               3446+*
    147C 74 02 B0                  3447+        ST    STOR52(,@BR),@XR          SAVE THE TO ADDR
    147F E2 02 06              3448+       LA    ##LNE(,@XR),@XR           BUMP TO NEXT ENTRY
    1482 74 02 B2              3449+       ST    STOR53(,@BR),@XR          SET THE FROM CADDR
    1485 5C 01 A1 98           3450+       MVC   STORAM(@CADDR,@BR),STOREL(,@BR) POSITIVE MODIFIER
    1489 D0 87 AA              3451+       B     STOR50(,@BR)              GO MOVE ENTRY
    148C C0 87 14BE            3452+STO047 B     STOR90                    GO RETURN
                          148F 3453+STORET EQU   STO047+@OP1               POINTER TO RETURN ACTION
                               3454+*
                               3455+*            ALL ENTRIES TESTED, CURRENT ENTRY LAST ONE.  CALCULATE
                               3456+*            HIGH END ADDR TO CHECK IF NEW ENTRY IS CONTIGUOUS.
                               3457+*
    1490 6C 03 B2 03           3458+STOR47 MVC   STORWK(STOENL,@BR),##DNEF(,@XR)
                               3459+*
    1494 D0 87 C5              3460+       B     STOR60(,@BR)              GO CALC HIGH END
                               3461+*
                               3462+*            TEST IF HIGH ADDR EQUAL TO START OF NEW ENTRY, IF NO GO
                               3463+*            MAKE NEW ENTRY, IF YES ADD NEW COUNT TO CURRENT ENTRY.
                               3464+*
    1497 5D 01 90 B0           3465+       CLC   STORCW(@CADDR,@BR),STORCS(,@BR) TEST HIGH AND NEW ENTRY
    149B F2 81 15              3466+       JE    STOR48                    JUMP IF CONTIGUOUS
    149E 7D 87 EA              3467+       CLI   STOR70+@Q(,@BR),@UCB      TEST IF NEW ENTRY IS ALLOWED
    14A1 F2 81 16              3468+       JE    STOR80                    ERROR EXIT
    14A4 E2 02 06              3469+       LA    ##LNE(,@XR),@XR           BUMP TO NEXT ENTRY
    14A7 9C 05 05 94           3470+       MVC   ##DNER(##LNE,@XR),STORWE+##LNEZ(,@BR) MOVE IN ENTRY
    14AB 1E 00 0000 9F         3471+STO048 ALC   *-*(##LAHC),STORC1(,@BR)  BUMP ENTRY COUNT
    14B0 F2 87 0B              3472+       J     STOR90                    GO RETURN
                               3473+*
    14B3 9E 01 03 92           3474+STOR48 ALC   ##DNEF(##LNEF,@XR),STOREC(,@BR) NEW COUNT TO CURRENT
    14B7 F2 87 04              3475+       J     STOR90                    GO RETURN
                               3476+*
                               3477+*            ERROR RETURN ACTION IF NO ENTRIES CAN BE MADE
                               3478+*
    14BA 3A 20 1B3E            3479+STOR80 SBN   SMIND1,SM1STN             TURN ON INDICATOR
                               3480+*
                               3481+*            ALL ACTION COMPLETE GO BACK TO CALLER
                               3482+*
    14BE C2 01 0000            3483+STOR90 LA    *-*,@BR                   RESTORE CALLERS REGS
    14C2 C2 02 0000            3484+STOR95 LA    *-*,@XR
    14C6 C0 87 0000            3485+STOR99 B     *-*                       RETURN TO THE CALLER
```

```
                               3487+***********************************************************************
                               3488+*            CONSTANTS AND WORK AREA                                  *
                               3489+***********************************************************************
    14CA                  14CD 3490+STORWE DS    IL(@DADDR+##LNEF)         WORK AREA FOR NEW ENTRY
                          14CB 3491+STORCW EQU   STORWE-##LNEF             NEW ENTRY ADDR
                          14CD 3492+STOREC EQU   STORWE                    POINTER TO ENTRY COUNT
                          14CB 3493+STORWA EQU   STORWE-##LNEF             ENTRY ADDR
    14CE 0003             14CF 3494+STOCLN DC    AL2(##DNEF)               DISPLACEMENT TO RIGHT END
    14D0 FFFE             14D1 3495+STORDR DC    AL2(@ZERO-##LNEZ)
    14D2 0006             14D3 3496+STOREL DC    AL2(##LNE)                INCR POINTERS
    14D4 0030             14D5 3497+STOC48 DC    IL2'48'                   INCR POINTERS
    14D6 FFFA             14D7 3498+STORNM DC    AL2(@ZERO-##LNE)          NEGATIVE MODIFIER
    14D8                  14D8 3499+STORWC DS    AL(##LAHC)
    14D9 0001             14DA 3500+STORC1 DC    IL2'1'                    INCR VALUE FOR COUNTERS
                          0002 3501+STORE2 EQU   2                         FIELD LENGTH FOR ADD AND SUBTR
    14DB                  14DC 3502+STORAM DS    IL(@CADDR)                ADDR MODIFIER FOR MOVE ROUTINE
                          0004 3503+STOENL EQU   @DADDR+##LNEF             LENGTH OF ADDR AND SECTOR COUNT
    14DD                  14DE 3504+STOENC DS    IL2                       ENTRY COUNT
    14DF 1524             14E0 3505+STO70A DC    AL2(STOR70)               CADDR OF INSERT NEW ENTRY
    14E1 0004             14E2 3506+STORHL DC    AL2(##LNH)                LENGTH OF HEADER
    14E3 14CD             14E4 3507+STOENA DC    AL2(STORWE)               ADDR OF ENTRY
```

```
                               3509+*
                               3510+*            THE FOLLOWING SUBROUTINE WILL MOVE THE END PORTION OF
                               3511+*            THE DIRCTY FORWARD OR BACKWARD DEPEND1NG ON THE VALUE
                               3512+*            OF THE MODIFIER PLUGGED IN BY THE CALLING ROUTINE.
                               3513+*            THE TO AND FROM ADDR FOR THE MOVE ARE ALSO PLUGGED IN
                               3514+*            BY THE CALLING ROUTINE,
                               3515+*
    14E5 74 08 E8              3516+STOR50 ST    STO067+@OP1(,@BR),@ARR    SAVE RETURN
                               3517+*
    14E8 0C 05 0000 0000       3518+STOR51 MVC   *-*(##LNE),*-*            MOVE ENTRY AS SPECIFIED
                          14EB 3519+STOR52 EQU   STOR51+@OP1               LOCATION OF TO ADDR
                          14ED 3520+STOR53 EQU   STOR51+@OP2               LOCATION OF FROM ADDR
    14EE 5F 00 9D 9F           3521+STOR55 SLC   STORWC(1,@BR),STORC1(,@BR) DECR WORK COUNT
    14F2 F2 81 2B              3522+       JE    STO067                    ZERO COUNT RETURN
    14F5 5E 01 B0 A1           3523+       ALC   STOR52(@CADDR,@BR),STORAM(,@BR) MODIFY THE TO ADDR
    14F9 5E 01 B2 A1           3524+       ALC   STOR53(@CADDR,@BR),STORAM(,@BR) MODIFY FROM ADDR
    14FD D0 87 AD              3525+       B     STOR51(,@BR)              GO MOVE NEXT ENTRY



                               3527+*
                               3528+*            THE FOLLOWING ROUTINE WILL CALCULATE THE HIGH END ADDR
                               3529+*            OF THE SPECIFIED ENTRY.
                               3530+*
    1500 74 08 E8              3531+STOR60 ST    STO067+@OP1(,@BR),@ARR    SAVE RETURN
                          14ED 3532+STORWK EQU   STOR53                    WORK AREA TO CALC HIGH ADOR
                          14EA 3533+STORC0 EQU   STORWK-##DNEF             POINTER TO LEFT BYTE
                          14EB 3534+STORCS EQU   STORWK-##LNEF             ENTRY DADDR
                               3535+*
    1503 5C 00 E3 AF           3536+       MVC   STO064+@Q(,@BR),STORC0(1,@BR) GET CYL BYTE
    1507 7C 00 AF              3537+       MVI   STORC0(,@BR),@ZERO        CLEAR HIGH ORDER CYL BYTE
    150A 5E 01 B0 B2           3538+       ALC   STORCS(##LNEF,@BR),STORWK(,@BR) ADD IN LENGTH
                               3539+*
    150E 5F 01 B0 9A           3540+STOR65 SLC   STORCS(STORE2,@BR),STOC48(,@BR) DECR IT CYL VALUE
    1512 F2 82 04              3541+       JL    STOR66                    GO RESTORE
                               3542+*
    1515 5E 00 E3 9F           3543+       ALC   STO064+@Q(1,@BR),STORC1(,@BR) BUMP CYL
                               3544+*      B     STOR65(,@BR)             BACK TO DECK AGAIN
    1519 5E 01 B0 9A           3545+STOR66 ALC   STORCS(STORE2,@BR),STOC48(,@BR) RESTORE REMAINDER
    151D 7C 00 AF              3546+STO064 MVI   STORC0(,@BR),*-*          MORE CYL COUNT
    1520 C0 87 0000            3547+STO067 B     *-*                       RETURN



                               3549+*
                               3550+*            THE FOLLOWING ROUTINE WILL INSERT A NEW ENTRY INTO THE
                               3551+*            DIRCTY IF THE SWITCH HAS BEEN SET TO ALLEW ENTRIES, IF
                               3552+*            A NEW ENTRY MUST BE MADE AND THE DIRCTY IS FULL THE ERROR
                               3553+*            EXIT IS TAKEN AND AN INDICATOR IS SET TO NOTE THE LIERART
                               3554+*            AREA SHOULD BE PACKED.
                               3555+*            NOTE - THIS ROUTINE DEPENDS ON THE NULL ENTRY BEING SIX
                               3556+*            BYTES.
                               3557+*
    1524 C0 00 14BA            3558+STOR70 BC    STOR80,*-*                BRANCH IF FULL SWITCH SET
                               3559+*
                          1523 3560+STOSAV EQU   STO067+@OP1               TEMP WORK AREA
    1528 7C 00 A2              3561+       MVI   STOENC-1(,@BR),@ZERO      CLEAR HIGH ORDER BYTE
    152B 7C 00 E7              3562+       MVI   STOSAV-1(,@BR),@ZERO      CLEAR HIGH ORDER BYTE
    152E 5E 01 A3 A3           3563+       ALC   STOENC(##LNEF,@BR),STOENC(,@BR) DOUBLE COUNT
    1532 5C 00 E8 A3           3564+       MVC   STOSAV(1,@BR),STOENC(,@BR) SAVE COUNT*2
    1536 5E 01 A3 E8           3565+       ALC   STOENC(##LNEF,@BR),STOSAV(,@BR) *4
    153A 5E 01 A3 E8           3566+       ALC   STOENC(##LNEF,@BR),STOSAV(,@BR) *6
    153E 5E 01 A3 A7           3567+       ALC   STOENC(STORE2,@BR),STORHL(,@BR) ADD HDR LENGTH
    1542 4E 01 A3 1B6A         3568+       ALC   STOENC(@CADDR,@BR),SMNDBA ADD START BUFFER
    1547 5F 01 A3 9F           3569+       SLC   STOENC(@CADDR,@BR),STORC1(,@BR) BACK TO RIGHT END
    154B 5C 01 B2 A3           3570+       MVC   STOR53(@CADDR,@BR),STOENC(,@BR) FROM ADDR
    154F 5E 01 A3 98           3571+       ALC   STOENC(@CADDR,@BR),STOREL(,@BR) BUMP TO RIGHT NEXT ENT
    1553 5C 01 B0 A3           3572+       MVC   STOR52(@CADDR,@BR),STOENC(,@BR) TO ADOR
    1557 5C 01 A1 9C           3573+       MVC   STORAM(@CADDR,@BR),STORNM(,@BR) NEGATIVE MODIFIER
    155B C0 87 14E5            3574+       B     STOR50                    BRANCH TO MOVER ROUTINE
                               3575+*
                               3576+*            MOVE THE NEW ENTRY INTO THE VACATED CURRENT ENTRY LOCATION
                               3577+*
    155F 9C 05 05 94           3578+       MVC   ##DNER(##LNE,@XR),STORWE+##LNEZ(,@BR) MOVE THE ENTRY IN
    1563 D0 87 70              3579+       B     STO048(,@BR)              GO RETURN
                               3580+***                     END OF STORIN                             ***
                               3581 *      $TUFI
                               3583+***********************************************************************
                               3584+*  5703-XM1 COPYRIGHT IBM CORP, 1970                                  *
                               3585+*           REFER TO INSTRUCTIONS ON COPY RIGHT NOTICE, 120-2083      *
                               3586+*                                                                     *
                               3587+***********************************************************************
                               3588+*STATUS                                                               *
                               3589+*   VERSION 1 MODIFICATION 0                                          *
                               3590+*                                                                     *
                               3591+*FUNCTION                                                             *
                               3592+*   STUFID INSERTS AN ENTRY IN A USER DIRECTORY BLOCK.  IF THE        *
                               3593+*   CURRENT DIRECTORY IS FULL STUFID WILL CREATE AN ADDITIONAL        *
                               3594+*   DIRECTORY AND LINK IT TO THE OLD BLOCK.                           *
                               3595+*                                                                     *
                               3596+*ENTRY POINTS                                                         *
                               3597+*   STUFID - ENTRY TO INSERT FILENAME ENTRY IN DIRECTORY BLOCK.       *
                               3598+*          THE CALLING SEQUENCE IS AS FOLLOWS:                        *
                               3599+*          B     STUFID                                               *
                               3600+*                                                                     *
                               3601+*INPUT                                                                *
                               3602+*   * SMUDEN MUST CONTAIN THE ADDRESS OF THE LEFT BYTE OF THE ENTRY   *
                               3603+*     TO BE INSERTED,                                                 *
                               3604+*   * SMUDBA MUST CONTAIN THE ADDRESS OF THE USER DIRECTORY BUFFER.   *
                               3605+*                                                                     *
                               3606+*OUTPUT                                                               *
                               3607+*   * THE ENTRY IS INSERTED INTO THE DIRECTORY WHICH IS THEN WRITTEN  *
                               3608+*     BACK TO THE DISK,                                               *
                               3609+*   * IF THE DIRECTORY IS FULL ANOTHER DIRECTORY IS CREATED.  THE NEW *
                               3610+*     BLOCK IS LINKED TO THE PREVIOUS DIRECTORY WHICH IS THE WRITTEN  *
                               3611+*     BACK TO DISK,  THE ENTRY IS MADE IN THE NEW BLOCK AND THEN      *
                               3612+*     WRITTEN BACK TO DISK,                                           *
                               3613+*                                                                     *
                               3614+*EXTERNAL REFERENCES                                                  *
                               3615+*     SMUPEN - CONTAINS THE ADDRESS OF THE NEW ENTRY.                 *
                               3616+*     SMUDBA - CONTAINS THE ADDRESS OF THE USER DIRECTORY.            *
                               3617+*     DL2ICS - DISK LOGICAL IOCS,                                     *
                               3618+*     SMNSCT - LOCATION OF REQUIRED NULL SECTOR COUNT.                *
                               3619+*     SURCHN - ENTRY TO SEARCH NULL DIRECTORY ROUTINE.                *
                               3620+*     SMNDEA - CONTAINS RELATIVE DISK ADDRESS OF NULL AREA.           *
                               3621+*     STUERR - ERROR RETURN TO USER,                                  *
                               3622+*                                                                     *
                               3623+*EXITS, NORNAL                                                        *
                               3624+*   NORMAL RETURN IS TO THE FIRST INSTRUCTION FILLOWING THE BRANCH    *
                               3625+*   TO STUFID,                                                        *
                               3626+*                                                                     *
                               3627+*EXITS, ERROR                                                         *
                               3628+*   IF AN ADDITIONAL DIRECTORY BLOCK RUST IT CREATED NO TWO SECTORS   *
                               3629+*   ARE NOT AVAILAILE,  A RETURN IS MARE TO STUD, IN THE CALLERS      *
                               3630+*   PROGRAM, @BR AND @XR ARE NOT RESTORED.                            *
                               3631+*                                                                     *
                               3632+*TABLES/WORKEARES                                                     *
                               3633+*   NONE                                                              *
                               3634+*                                                                     *
                               3635+*ATTRIBUTES                                                           *
                               3636+*   RELOCATAILE, REUSABLE                                             *
                               3637+*                                                                     *
                               3638+*CHARACTER CODE DEPENDENCY                                            *
                               3639+*   THE OPERATION OF THIS MODULE DOES NOT DEPEND UPON A PARTICULAR    *
                               3640+*   INTERNAL REPRESENTATION OF THE EXTERNAL CHARACTER SET.            *
                               3641+*                                                                     *
                               3642+*NOTES                                                                *
                               3643+*   ERROR PROCEDURES                                                  *
                               3644+*      A BRANCH IS TAKEN TO STUERR IN THE CALLERS PGM IF 2 SECTORS    *
                               3645+*      ARE NOT AVAILABLE TO CREATE A NEW USER DIRECTORY BLOCK.        *
                               3646+*                                                                     *
                               3647+*   REGISTER USAGE                                                    *
                               3648+*      * @BR AND @XR ARE SAVED AND RESTORED ON EXIT.  @ARR IS STORED  *
                               3649+*        IN THE BRANCH INSTRUCTION FOR RETURN.                        *
                               3650+*      * DURING EXECUTION @BR IS USED AS A BASE REGISTER AND @XR IS   *
                               3651+*        USED AS A GENERAL WORK REGISTER IN THE DIRECTORY.            *
                               3652+*                                                                     *
                               3653+*   SAVED/RESTORED AREAS                                              *
                               3654+*      NONE                                                           *
                               3655+*                                                                     *
                               3656+*   MODIFICATION CONSIDERATIONS                                       *
                               3657+*      N/A                                                            *
                               3658+*                                                                     *
                               3659+*   REQUIRED MODULES                                                  *
                               3660+*      @SYSEQ - SYSTEM SOFTWARE EQUATES                               *
                               3661+*      @DIREQ - LIBRARY DIRECTORY EQUATES                             *
                               3662+*      TSMLES - DATA MANAGEMENT COMMUNICATIONS AREA                   *
                               3663+*      DL2ICS - DISK IOCS ROUTINE                                     *
                               3664+*      SURCHN - SEARCH NULL DIRECTORY ROUTINE                         *
                               3665+*                                                                     *
                               3666+*   OTHER                                                             *
                               3667+*      N/A                                                            *
                               3668+***********************************************************************
```

```
                          1566 3670+STUFID EQU   *                         ENTRY TO STUFID
                          0001 3671+STUE01 EQU   1                         VALUE TO INITIALIZE COUNTER
                          0002 3672+STUE02 EQU   2                         VALUE FOR Q CODE
    1566 34 01 15F8            3673+       ST    STU900+@OP1,@BR           SAVE BASE REGISTER
                          156A 3674+       USING STU000,@BR
    156A C2 01 156A            3675+STU000 LA    STU000,@BR                SET UP BASE REGISTER
    156E 74 08 96              3676+       ST    STU920+@OP1(,@BR),@ARR    SAVE RETURN ADDR
    1571 74 02 92              3677+       ST    STU910+@OP1(,@BR),@XR     SAVE INDEX REGISTER
                               3678+*
    1574 35 02 1B64            3679+       L     SMUPEN,@XR                GET CADDR OF NEW ENTRY
    1578 E2 02 31              3680+       LA    ##DUER(,@XR),@XR          BUMP TO RIGHT END
    157B 74 02 43              3681+       ST    STU020+@DOP2(,@BR),@XR    SET IN MOVE TO DIRCTY
    157E 74 02 79              3682+       ST    STU060+@DOP2(,@BR),@XR    SET IN MOVE TO NEW DIRCTY
    1581 35 02 1B5A            3683+       L     SMUDBA,@XR                ACTIVE BUFFER ADDR
    1585 74 02 9C              3684+       ST    STULST+@DBFR2(,@BR),@XR   ACTIVE BUFFER ADDR
    1588 6C 01 99 01           3685+       MVC   STULST+@DSAD(@DADDR,@BR),##DUHA(,@XR) DADDR OF BLOCK
    158C BD 0A 04              3686+       CLI   ##DUHC(,@XR),##MUHM       TEST FOR MAX COUNT
    158F F2 02 2C              3687+       JNL   STU040                    GO SEARCH FOR NEW BLOCK
                               3688+*
    1592 6C 00 A6 04           3689+       MVC   STUCNT(1,@BR),##DUHC(,@XR) PICK UP COUNT FOR WORK
                               3690+*
    1596 E2 02 0C              3691+       LA    ##DUE1(,@XR),@XR          BUMP PAST HEADER
                               3692+*
    1599 7D 00 A6              3693+STU010 CLI   STUCNT(,@BR),@ZERO        TEST IF COUNT EXHAUSTED
    159C F2 81 0A              3694+       JE    STU020                    ZERO IS END OF DIRCTY
    159F 5F 00 A6 E4           3695+       SLC   STUCNT(1,@BR),STUC01(,@BR) DECR ENTRY COUNT
    15A3 E2 02 32              3696+       LA    ##LUE(,@XR),@XR           NEXT ENTRY
    15A6 D0 87 2F              3697+       B     STU010(,@BR)              BACK TO BUMP TO NEXT ENTRY
                               3698+*
    15A9 8C 31 31 0000         3699+STU020 MVC   ##LUE-1(##LUE,@XR),*-*    MOVE NEW ENTRY INTO DIRCTY
    15AE 75 02 9C              3700+       L     STULST+@DBFR2(,@BR),@XR   RESTORE ACTIVE BUFFER POINTER
    15B1 9E 00 04 E4           3701+       ALC   ##DUHC(1,@XR),STUC01(,@BR) BUMP DIRCTY ENTRY COUNT
                               3702+*
    15B5 C0 87 1268            3703+       B     DL2ICS                    REPLACE DIRCTY ON DISK
    15B9 1601             15BA 3704+       DC    AL2(STULST)               ADDR OF DPL
                               3705+*
    15BB F2 87 37              3706+       J     STU900                    GO TO RETURN
                               3707+*
                               3708+*             OLD BLOCK IS FULL, GO LOOK FOR 2 SECTORS TO BUILD A NEW
                               3709+*             USER DIRECTORY.
                               3710+*
    15BE 1C 01 1B60 A5         3711+STU040 MVC   SMNSCT,STUCLU(STUE02,@BR) REQUIRED SECTOR COUNT
    15C3 C0 87 0A50            3712+       B     SURCHN                    SEARCH NULL DIRCTY FOR A SPACE
    15C7 1D 01 1B5E A4         3713+       CLC   SMNDEA(@DADDR),STUC00(,@BR) TEST IF SPACE FOUND
    15CC C0 81 0CF7            3714+       BE    STUERR                    GO TAKE ERROR RUTURN
                               3715+*
    15D0 8C 01 03 1B5E         3716+STU050 MVC   ##DUHB(,@XR),SMNDEA(@DADDR) SET LINK IN OLD BLK HEADER
                               3717+*
    15D5 C0 87 1268            3718+       B     DL2ICS                    WRITE OLD BLOCK BACK TO DISK
    15D9 1601             15DA 3719+       DC    AL2(STULST)               POINTER TO OLD DPL
                               3720+*
    15DB 5F 0B B1 B1           3721+       SLC   STUNHD(,@BR),STUNHD(,@BR) CLEAR HEADER AREA
    15DF 4C 31 E3 0000         3722+STU060 MVC   STUNNT(##LUE,@BR),*-*     MOVE NEW ENTRY NEXT TO NEW HDR
                               3723+*
                               3724+*             NOW IN ENTRIES TO FORM NEN DIRCTY BLOCK HEADER
                               3725+*
    15E4 7C 01 AA              3726+       MVI   STUHDR+##DUHC(,@BR),STUE01 INITIAL COUNT
    15E7 6C 01 A7 03           3727+       MVC   STUHDR+##DUHA(@DADDR,@BR),##DUHB(,@XR) NEW BLK ADDR
    15EB 6C 01 9F 03           3728+       MVC   STUDPL+@DSAD(@DADDR,@BR),##DUHB(,@XR) NEW BLK ADDR
    15EF C0 87 1268            3729+       B     DL2ICS                    WRITE THE NEW BLOCK
    15F3 1607             15F4 3730+       DC    AL2(STUDPL)               POINTER TO DPL
                               3731+*
    15F5 C2 01 0000            3732+STU900 LA    *-*,@BR                   RESTORE BASE
    15F9 C2 02 0000            3733+STU910 LA    *-*,@XR                   RESTORE INDEX
    15FD C0 87 0000            3734+STU920 B     *-*                       RETURN
```

```
                               3736+*
                               3737+*            CONSTANTS AND WORKAREA
                               3738+*
                               3739+*                                      START OF DPL FOR OLD BLOCK
    1601 02               1601 3740+STULST DC    AL1(@DPUT)                READ OP CODE
    1602                  1603 3741+       DS    CL(@DADDR)                DISK ADDR SPACE
    1604 02               1604 3742+       DC    AL1(##LU)                 SECTOR COUNT OF DIRCTY
    1605                  1606 3743+       DS    CL(@CADDR)                BUFFER ADDR
    1607 02               1607 3744+STUDPL DC    AL1(@DPUT)                START OF DPL FOR NEW BLOCK
    1608                  1609 3745+       DS    CL(@DADDR)                NEW DISK ADDR
    160A 02               160A 3746+       DC    AL1(##LU)                 SECTOR COUNT OF DIRCTY
    160B 1610             160C 3747+       DC    AL2(STUHDR)               NEW BLOCK HEADER ADDR
    160D 0000             160E 3748+STUC00 DC    IL2'0'                    TEST VALUE FOR SPACE FOUND
    160F 02               160F 3749+STUCLU DC    AL1(##LU)                 SECTOR COUNT FOR USER DIRCTY
                               3750+*
                               3751+*         FOLLOWING IS THE NEW HEADER TO BE WRITTEN IF A NEW USER
                               3752+*         DIRECTRY BLOCK IS CREATED.
                               3753+*
                          1610 3754+STUHDR EQU   *                         START OF HEADER
    1610                  161B 3755+STUNHD DS    IL(##LUH)                 SAVE AREA FOR NEW ENTRY
                          1610 3756+STUCNT EQU   STUHDR                    WORK AREA FOR COUNTER
    161C                  164D 3757+STUNNT DS    IL(##LUE)                 SAVE AREA FOR NEW ENTRY
                               3758+*
    164E 01               164E 3759+STUC01 DC    IL1'1'                    CONSTANT 1 TO DECR ENTRY COUNT
                               3760+***                       END OF STUFID                             ***
                               3761 *      $FIND
                               3763+***********************************************************************
                               3764+*  5703-XM1    COPYRIGHT IBM CORP. 1970                               *
                               3765+*              REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083   *
                               3766+*                                                                     *
                               3767+***********************************************************************
                               3768+*STATUS                                                               *
                               3769+*   VERSION 1 MODIFICATION 0                                          *
                               3770+*                                                                     *
                               3771+*FUNCTION                                                             *
                               3772+*   * SFINDF IS A CONTROL MODULE USED TO LOCATE A SPECIFIED PASSWORD  *
                               3773+*     AND/OR FILENAME.                                                *
                               3774+*   * IF THE FILENAME, PASSWORD, AND VOLUME-ID ARE ALL EXPLICITLY     *
                               3775+*     SPECIFIED. A CALL IS ISSUED TO SVOLID, SGETDB AND SRCHFN TO     *
                               3776+*     SEARCH FOR THE REQUIRED FILE IN THE FILE LIBRARY SPECIFIED.     *
                               3777+*     IF THE PASSWORD OR VOLUME-ID IS NOT EXPLICITLY DEFINED, SFINDF  *
                               3778+*     WILL DEFAULT TO THE CURRENT USER SPECIFICATIONS, IF THEY EXIST, *
                               3779+*     FOR THE MISSING PARAMETERS AND THEN ISSUE THE REQUIRED CALLS    *
                               3780+*     TO SGETDS AND/OR SRCHFN TO LOCATE THE FILE.                     *
                               3781+*   * IF A ONE OR TWO-STAR FILENAME IS SPECIFIED, THE SPECIFIED DISK, *
                               3782+*     OR ALL DISKS ON THE SYSTEM WILL BE SEARCHED IN AN ATTEMPT TO    *
                               3783+*     LOCATE THE FILE. THE CALLER MAY SET AN INDICATOR TO TERMINATE   *
                               3784+*     THE SEARCH AFTER A GIVEN NUMBER OF DISKS HAVE BEEN SEARCHED.    *
                               3785+*                                                                     *
                               3786+*ENTRY POINTS                                                         *
                               3787+*   THE ENTRY POINT IS SFINDF.                                        *
                               3788+*   THE CALLING SEQUENCE IS AS FOLLOWS:                               *
                               3789+*        B     SFINDF                                                 *
                               3790+*                                                                     *
                               3791+*INPUT                                                                *
                               3792+*   * THE FOLLOWING INFORMATION MUST BE SET UP IN TSMLES BEFORE       *
                               3793+*     CALLING SFINDF.                                                 *
                               3794+*        * SMPSWD  MUST CONTAIN SPECIFIED PASSWORD                    *
                               3795+*        * SMVOID  MUST CONTAIN SPECIFIED VOLUME                      *
                               3796+*        * SMFNAM  MUST CONTAIN SPECIFIED FILENAME                    *
                               3797+*   * THE FOLLOWING SWITCHES ARE PROVIDED TO HANDLE ONE OR TWO-STAR   *
                               3798+*     FILES:                                                          *
                               3799+*        * SFIVOL - IF @NOP IS SET SVOLID WILL NOT BE CALLED. SVOLID  *
                               3800+*                   IS NOT REUSABLE AND THIS SWITCH MUST BE SET BEFORE*
                               3801+*                   SFINDF IS CALLED A SECOND TIME.                   *
                               3802+*        * SFISTR - IF @NOP IS SET ONLY 1 DISK WILL BE SEARCHED       *
                               3803+*        * SFIFND - IF @NOP SET WITH SFIVOL ONLY THE NUMBER OF DISKS  *
                               3804+*                   SPECIFIED IN SFINTR WILL BE SEARCHED.             *
                               3805+*                                                                     *
                               3806+*OUTPUT                                                               *
                               3807+*   * THE OUTPUT FROM SFINDF IS SET IN TSMLES, THE POINTERS AND USER  *
                               3808+*     DIRECTORIES REQUIRED ARE INITALIZED.                            *
                               3809+*                                                                     *
                               3810+*EXTERNAL REFERENCES                                                  *
                               3811+*        TSMLES - (SMALES) DATA MANAGEMENT SAVE AREAS AND BUFFERS.    *
                               3812+*        $VOLID - CORE RESIDENT VOLID TABLE.                          *
                               3813+*        $USRDR - DISPLACEMENT TO CURRENT USER DIRECTORY.             *
                               3814+*        $FILIB - CURRENT USER FILE LIBRARY DISK ADDRESS.             *
                               3815+*        DL2ICS - TWO TRACK LOGICAL IOCS.                             *
                               3816+*        SRCHFN - SEARCH USER DIRCTY BLOCK.                           *
                               3817+*        SGETDB - SEARCH PASSWORD DIRCTY.                             *
                               3818+*        SVOLID - SEARCH VOL-ID TABLE.                                *
                               3819+*        $CAERR - SAVE AREA FOR SYSTEM ERROR MESSAGT CODE.            *
                               3820+*                                                                     *
                               3821+*EXITS, NORMAL                                                        *
                               3822+*   * NORMAL RETURN IS TO THE CALLER FOLLOWING THE BRANCH TO SFINDF.  *
                               3823+*                                                                     *
                               3824+*EXITS, ERROR                                                         *
                               3825+*   * THE ERROR RETURN IS TO SFIERR WHICH MUST BE DEFINED BY THE      *
                               3826+*     CALLER.                                                         *
                               3827+*                                                                     *
                               3828+*TABLES/WORKAREAS                                                     *
                               3829+*   * N/A                                                             *
                               3830+*                                                                     *
                               3831+*ATTRIBUTES                                                           *
                               3832+*   * RELOCATABLE                                                     *
                               3833+*   * RE-USABLE                                                       *
                               3834+*                                                                     *
                               3835+*CHARACTER CODE DEPENDENCY                                            *
                               3836+*   * THE OPERATION OF THIS MODULE DOES NOT DEPEND UPON A PARTICULAR  *
                               3837+*     INTERNAL REPRESENTATION OF THE EXTERNAL CHARACTER SET.          *
                               3838+*                                                                     *
                               3839+*NOTES                                                                *
                               3840+*   ERROR PROCEDURES                                                  *
                               3841+*      IF A FILE-SPEC WAS NOT ENTERED AND A CURRENT USER IS NOT IN    *
                               3842+*      AFFECT.  THE ERROR EXIT TO SFIERR IS TAKEN.                    *
                               3843+*                                                                     *
                               3844+*   REGISTER USAGE                                                    *
                               3845+*      @BR AND @XR ARE SAVED AND RESTORED. DURING EXECUTION @BR IS    *
                               3846+*      USED AS A BASE REGISTER AND @XR IS USED TO POINT TO $NUCBS.    *
                               3847+*                                                                     *
                               3848+*   SAVED/RESTORED AREAS                                              *
                               3849+*      NONE                                                           *
                               3850+*                                                                     *
                               3851+*   MODIFICATION CONSIDERATIONS                                       *
                               3852+*      NONE                                                           *
                               3853+*                                                                     *
                               3854+*   REQUIRED MODULES                                                  *
                               3855+*        @SYSEQ - SYSTEM SOFTWARE EQUATES.                            *
                               3856+*        @FXDEQ - SYSTEM NUCLEUS ADDRESSES AND INDICATOR VALUES.      *
                               3857+*        TSMLES - DATA MANAGEMENT SAVE AREAS AND BUFFERS.             *
                               3858+*        $VOLID - SEARCH VOLUME-ID SUBROUTINE.                        *
                               3859+*        SRCHFN - SEARCH FOR FILENAME SUBROUTINES.                    *
                               3860+*        SGETDB - SEARCH PASSWORD DIRECTORY SUBROUTINE.               *
                               3861+*        DL2ICS - TWO TRACK DISK LOGICAL IOCS.                        *
                               3862+*                                                                     *
                               3863+*   OTHER                                                             *
                               3864+*      NONE                                                           *
                               3865+***********************************************************************


                               3867+*
                               3868+*                  EQUATES USED IN THIS SUBROUTINE
                               3869+*
                          164F 3870+SFINDF EQU   *                         START OF MODULE
    164F 34 01 175C            3871+       ST    SFISBR,@BR                SAVE @BR
    1653 C2 01 168D            3872+       LA    SFIBSE,@BR                SET LOCAL BASE
                          168D 3873+       USING SFIBSE,@BR                *
    1657 74 08 D3              3874+       ST    SFIEXT(,@BR),@ARR         SAVE RETURN ADDR
    165A 74 02 CB              3875+       ST    SFISXR(,@BR),@XR          SAVE @XR
    165D C2 02 03C0            3876+       LA    $NUCBS,@XR                SET NUCLEUS BASE
                          03C0 3877+       USING $NUCBS,@XR                *
    1661 3D 40 1B45            3878+       CLI   SMPSWD-##LPEN+@B1,@BLANK  WAS A PASSWD SPECIFIED ?
    1665 F2 81 98              3879+       JE    SFI500                    NO, GO CHECK LOGON STATUS
    1668 3D 40 174F            3880+       CLI   SMVOID-$VOLID+@B1,@BLANK  WAS A VOL-ID SPECIFIED ?
    166C F2 81 07              3881+       JE    SFI100                    NO, GO CHECK LOGON STATUS
    166F C0 87 03F6            3882+SFI050 B     $VOLID                    RESOLVE SPECIFIED VOL-ID
                          1670 3883+SFIVOL EQU   SFI050+@Q                 SET TO A NOP FOR SUCCESSIVE USE
    1673 F2 87 75              3884+       J     SFI350                    GO TO GET DIRECTORY
                               3885+*
                               3886+*            PASSWORD WAS SPECIFIED, BUT VOL-ID WAS NOT
                               3887+*
    1676 3D 5C 1B45            3888+SFI100 CLI   SMPSWD-##LPEN+@B1,SFIAST  IS PASSWORD AN '*' ?
    167A F2 01 63              3889+       JNE   SFI320                    NO, GO CHK FOR FILE LIBR DADDR
    167D 7C 00 D4              3890+       MVI   SFICTR(,@BR),@ZERO        YES, INITLZ LOOP CTR TO ZERO
    1680 7C 00 DB              3891+       MVI   SFITTC(,@BR),@ZERO        INITLZ THIS TIME COUNTER
    1683 BD 00 19              3892+       CLI   $FILIB-@B1(,@XR),@ZERO    CURRENT USER IN FORCE ?
    1686 F2 01 5D              3893+       JNE   SFI340                    YES, GO TRY THAT FIRST
    1689 3A 08 1B3E            3894+       SBN   SMIND1,SM1PNF             SET PASSWORD NOT FOUND INDR.
                               3895+*
                               3896+*            THE FOLLOWING ROUTINE WILL SEARCH ALL DISKS ON THE
                               3897+*            SYSTEM FOR THE SPECIFIED ONE OR TWO STAR FILE
                               3898+*
    168D 7D 01 D4              3899+SFI200 CLI   SFICTR(,@BR),@B1          CHECK THE DISK POINTER
    1690 F2 82 1A              3900+       JL    SFI220                    GO CHECK F1
    1693 F2 81 28              3901+       JE    SFI230                    GO CHECK F2
    1696 7D 03 D4              3902+       CLI   SFICTR(,@BR),SFIE03
    1699 F2 82 33              3903+       JL    SFI240                    GO CHECK R1
                               3904+*
    169C BD 00 4C              3905+SFI210 CLI   $VOLR2+SFIE06(,@XR),@ZERO DOES R2 CONTAIN A FILE LIBR
    169F F2 81 AC              3906+       JE    SFI545                    NO, NO MORE TO CHK, GO RETURN
    16A2 2C 01 1B58 4D         3907+       MVC   SMBFDA(@DADDR),$VOLR2+SFIE07(,@XR) SET LIBR DADDR FOR
    16A7 7C FE D4              3908+       MVI   SFICTR(,@BR),SFIEFE       * SEARCH AND INCR DISK POINTER
    16AA F2 87 3E              3909+       J     SFI350                    GO TO SEARCH
                               3910+*
    16AD BD 00 44              3911+SFI220 CLI   $VOLF1+SFIE06(,@XR),@ZERO DOES F1 CONTAIN A FILE LIBR
    16B0 F2 81 0B              3912+       JE    SFI230                    NO, GO CHECK F2
    16B3 2C 01 1B58 45         3913+       MVC   SMBFDA,$VOLF1+SFIE07(@DADDR,@XR) SET LIBR DADDR FOR SEWN
    16B8 7C 01 D4              3914+       MVI   SFICTR(,@BR),@B1          INCR DISK POINTER
    16BB F2 87 2D              3915+       J     SFI350                    SO TO SEARCH
                               3916+*
    16BE BD 00 54              3917+SFI230 CLI   $VOLF2+SFIE06(,@XR),@ZERO DOES F2 CONTAIN A FILE LIBR
    16C1 F2 81 0B              3918+       JE    SFI240                    NO, SO CHECK R1
    16C4 2C 01 1B58 55         3919+       MVC   SMBFDA,$VOLF2+SFIE07(@DADDR,@XR) SET LIBR DADDR FOR SEACH
    16C9 7C 02 D4              3920+       MVI   SFICTR(,@BR),SFIE02       INCR DISK POINTER
    16CC F2 87 1C              3921+       J     SFI350                    GO TO SEARCH
                               3922+*
    16CF BD 00 3C              3923+SFI240 CLI   $VOLR1+SFIE06(,@XR),@ZERO DOES R1 CONTAIN A FILE LIBR
    16D2 D0 81 0F              3924+       BE    SFI210(,@BR)              NO, GO CHECK R2
    16D5 2C 01 1B58 3D         3925+       MVC   SMBFDA,$VOLR1+SFIE07(@DADDR,@XR) SET LIBR DADDR FOR SEARC
    16DA 7C 03 D4              3926+       MVI   SFICTR(,@BR),SFIE03       INCR DISK POINTER
    16DD F2 87 0B              3927+       J     SFI350                    GO TO SEARCH
                               3928+*
                               3929+*            PASSWORD SPECIFIED, BUT VOLUME ID WAS NOT.
                               3930+*                  CHECK FOR CURRENT USER
                               3931+*
    16E0 BD 00 19              3932+SFI320 CLI   $FILIB-@B1(,@XR),@ZERO    CURRENT USER SPEC IN FORCE
    16E3 F2 81 20              3933+       JE    SFI505                    NO, GO TO ERR ROUTINE
    16E6 2C 01 1B58 1A         3934+SFI340 MVC   SMBFDA(@DADDR),$FILIB(,@XR) YES, SET TO USER LIBR
                               3935+*
                               3936+*                  SO SEARCH FOR SPECIFIED PASSWORD
                               3937+*
    16EB C0 87 1A0A            3938+SFI350 B     SGETDB                    SEARCH FOR PASSWORD
    16EF 38 08 1B3E            3939+       TBN   SMIND1,SM1PNF             WAS PASSWORD FOUND
    16F3 F2 10 3B              3940+       JT    SFI540                    NO, GO TEST STAR COUNTER
    16F6 38 10 1B3E            3941+       TBN   SMIND1,SM1PDS             PASSWORD DIRCTY ONLY REQ'SED
    16FA F2 10 58              3942+       JT    SFI550                    YES, GO RETURN TO USER
    16FD F2 87 26              3943+       J     SFI520                    NO, GO SEARCH FOR FILENAME
                               3944+*
                               3945+*                  ONLY FILENAME SPECIFIED, CHECK FOR CURRENT USER
                               3946+*
    1700 BD 00 19              3947+SFI500 CLI   $FILIB-@B1(,@XR),@ZERO    CURRENT USER SPEC IN FORCE
    1703 F2 01 07              3948+       JNE   SFI510                    YES, BYPASS ERROR MESSAGE
    1706 BC 21 0D              3949+SFI505 MVI   $CAERR(,@XR),@@E200       SET NO CURRENT USER ERROR CODE
    1709 C0 87 0CF7            3950+       B     SFIERR                    GO TO ERROR RETURN
                               3951+*
                               3952+*                  GET FIRST USER DIRECTORY BLOCK
                               3953+*
    170D 2C 01 1300 1A         3954+SFI510 MVC   DL2RAD,$FILIB(@DADDR,@XR) SET DL2ICS BASE DADDR
    1712 2C 01 1B58 1A         3955+       MVC   SMBFDA,$FILIB(@DADDR,@XR) SET LIBR DADDR TO COMMON AREA
    1717 6C 01 D7 1C           3956+       MVC   SFIRDA(,@BR),$USRDR(@DADDR,@XR) SET DL2ICS RELATIVE DADDR
    171B C0 87 1268            3957+       B     DL2ICS                    GO READ USER DIRECTORY BLOCK
    171F 1762             1720 3958+       DC    AL2(SFIDPL)               * CADDR OF DPL
    1721 2C 01 1B68 1C         3959+       MVC   SMFUDA,$USRDR(@DADDR,@XR) PRESERVE 1ST BLOCK REL. DADDR
                               3960+*
                               3961+*                  SEARCH USER DIRECTORY BLOCK FOR FILENAME
                               3962+*
    1726 C0 87 1301            3963+SFI520 B     SRCHFN                    GO TO SEARCH ROUTINE
    172A 38 80 1B3E            3964+       TBN   SMIND1,SM1FNE             WAS NAME FOUND
    172E F2 10 24              3965+       JT    SFI550                    YES, SO RETURN
                               3966+*
                               3967+*                  PASSWORD OR FILENAME NOT FOUND
                               3968+*
    1731 7D FE D4              3969+SFI540 CLI   SFICTR(,@BR),SFIEFE       ONE OR TWO STAR FILE WITH MORE
    1734 F2 84 1E              3970+       JH    SFI550                    * DISKS TO SEARCH ? NO, GET OUT
    1737 D0 82 00              3971+SFI542 BC    SFI200(,@BR),@BL          * YES, GO SEARCH
                          1738 3972+SFISTR EQU   SFI542+@Q                 * NOP FOR 1ST * OR ** SEARCHED
    173A F2 87 11              3973+SFI543 JC    SFI545,@UCB               BYPASS TRY CONTROL UNLESS
                          173B 3974+SFIFND EQU   SFI543+@Q                 * Q-CODE CHANGED TO A NOP
    173D 7D 06 DC              3975+       CLI   SFINTR(,@BR),SFIETD       IS TRY COUNTER AT MAX ?
    1740 F2 02 0B              3976+       JNL   SFI545                    YES, SO SET ERROR CODE
    1743 5E 00 DB DD           3977+       ALC   SFITTC(,@BR),SFIONE(,@BR) INCR THIS TRY COUNTER
    1747 5D 00 DB DC           3978+       CLC   SFITTC(,@BR),SFINTR(1,@BR) THIS TRY = TRYS REQUIRED ?
    174B D0 01 00              3979+       BNE   SFI200(,@BR)              NO, GO TRY THE NEXT DISK
    174E BC 26 0D              3980+SFI545 MVI   $CAERR(,@XR),@@E213       SET * OR ** NOT FOUND CODE
    1751 3A 80 1B3E            3981+       SBN   SMIND1,SM1FNE             SET ON FILE NOT FOUND INDR.
                               3982+*
                               3983+*                  RETURN TO USER
                               3984+*
    1755 C2 02 0000            3985+SFI550 LA    *-*,@XR                   RELOAD @XR
```

```
                          1758 3986+SFISXR EQU   SFI550+@OP1                 *
    1759 C2 01 0000            3987+SFI560 LA    *-*,@BR                     RELOAD @BR
                          175C 3988+SFISBR EQU   SFI560+@OP1                 *
    175D C0 87 0000            3989+SFI570 B     *-*                         RETURN TO THE USER
                          1760 3990+SFIEXT EQU   SFI570+@OP1                 *
                               3991+*
                               3992+*                  CONSTANTS AND SAVE AREAS
                               3993+*
    1761                  1761 3994+SFICTR DS    XL1                         COUNTER USED TO CONTROL THE
    1761                       3995+       ORG   *-1                         * SEARCH FOR A STAR FILE
    1761 FF               1761 3996+       DC    AL1(SFIEFF)                 INITLZ'D FOR NO SEARCH
    1762 01               1762 3997+SFIDPL DC    AL1(@DGET)                  DPL TO READ USER DIRCTY BLOCK 1
    1763                  1764 3998+SFIRDA DS    XL2                         * RELATIVE DISK ADDRESS
    1765 02               1765 3999+       DC    XL1'02'                     * SECTOR COUNT
    1766 1B6D             1767 4000+       DC    AL2(SMUDB1)                 * CORE BUFFER ADDRESS
    1768                  1768 4001+SFITTC DS    CL1                         THIS TRY COUNTER
    1769                  1769 4002+SFINTR DS    CL1                         NUMBER OF TRYS REQUIRED COUNTER
    1769                       4003+       ORG   SFINTR                      INITLZ NUMBER CF TRYS REQUIRED
    1769 00               1769 4004+       DC    XL1'0'                      * COUNTER TO ZERO
    176A 01               176A 4005+SFIONE DC    XL1'1'                      COUNTER INCREMENT
                               4006+*
                               4007+*                  EQUATES
                               4008+*
                          0CF7 4009+SVOERR EQU   SFIERR                      SVOLID ERROR RETURN ADDRESS
                          005C 4010+SFIAST EQU   C'*'                        STAR LIBR TEST CHARACTER
                          0002 4011+SFIE02 EQU   X'02'                       STAR COUNTER TEST R1 CODE
                          0003 4012+SFIE03 EQU   X'03'                       STAR COUNTER TEST R2 CODE
                          00FE 4013+SFIEFE EQU   X'FE'                       STAR COUNTER COMPLETE CODE
                          00FF 4014+SFIEFF EQU   X'FF'                       NOT A * OR ** FILE COUNTER CODE
                          0006 4015+SFIE06 EQU   X'06'                       DISP TO LIBR DADDR BYTE 0
                          0007 4016+SFIE07 EQU   X'07'                       DISP TO LIBR DADDR BYTE 1
                          168D 4017+SFIBSE EQU   SFI200                      LOCAL BASE ADDRESS
                          176A 4018+SFIEND EQU   *-1                         LAST BYTE OF SFINDF
                          0006 4019+SFIETD EQU   6                           MAX TRY REQUIRED COUNTER VALUE
                          0001 4020+       DROP  @BR
                          0002 4021+       DROP  @XR
                               4022+***                       END OF SFINDF                           ***
                               4023 *      $VOLI
```

```
4025+***********************************************************************
4026+* 5703-XM1  COPYRIGHT IBM CORP. 1970                                  *
4027+*           REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE  120-2083       *
4028+*                                                                     *
4029+***********************************************************************
4030+*STATUS                                                               *
4031+*   VERSION 1 MODIFICATION 0                                          *
4032+*                                                                     *
4033+*FUNCTION                                                             *
4034+*   THE FUNCTION OF SVOLID IS TO SEARCH THE CORE RESIDENT TABLE OF    *
4035+*   VOLUME ID'S ON THE SYSTEM FOR A SPECIFIED VOLUME ID. IF THE       *
4036+*   VOLUME IS NOT FOUND, AN ERROR CODE WILL BE PUT IN $CAERR AND AN   *
4037+*   EXIT TO $VOERR IN THE CALLING ROUTINE WILL BE TAKEN. IF MORE      *
4038+*   THAN ONE VOLUME WITH THE SAME VOL-ID IS FOUND ON THE SYSTEM, THE  *
4039+*   USER OF THE SYSTEM IS REQUESTED TO INDICATE WHICH DRIVE AND DISK  *
4040+*   IS TO BE USED. IF THE USER IS UNABLE TO RESOLVE THE CONFLICT,     *
4041+*   THE COMMAND IS REJECTED. IF THE INPUT SOURCE IS NOT THE KEYBOARD, *
4042+*   THE COMMAND IS REJECTED. OTHERWISE THE FILE LIBRARY ADDRESS OF    *
4043+*   THE RESOLVED VOLUME IS PLACED IN SMBFDA IN THE TSMLES COMMUNICA-  *
4044+*   TIONS REGION, AND A NORMAL RETURN IS TAKEN.                       *
4045+*                                                                     *
4046+*ENTRY POINTS                                                         *
4047+*   $VOLID - THE FIRST EXECUTABLE INSTRUCTION. IT IS ASSUMED THAT     *
4048+*   SMVOID IN TSMLES HAS BEEN PRIMER. ALSO, IF THE VM OPTION OF       *
4049+*   SVOLID HAS BEEN ASSEMBLED FOR EXECUTION TIME USAGE.               *
4050+*   THE FIELDS SVOIOF AND SVODSK SHOULD BE PRIMED WITH THE GET/PUT    *
4051+*   GET/PUT FILENAME AND DISK FILENAME, RESPETIVELY.                  *
4052+*                                                                     *
4053+*INPUT                                                                *
4054+*   INPUT TO SVOLID IS THE SPECIFIED VOL-ID IN THE TSMLES REGION -    *
4055+*   SMVOID.                                                           *
4056+*                                                                     *
4057+*OUTPUT                                                               *
4058+*   OUTPUT FROM SVOLID IS THE FILE LIBRARY ADDRESS OF THE RESOLVED    *
4059+*   SPECIFIED VOL-ID - PLACED IN SMBFDA.                              *
4060+*                                                                     *
4061+*EXTERNAL REFERENCES                                                  *
4062+*       SVOBUF - TEMPORARY SECTOR BUFFER SAVE AREA - USER SUPPLIED    *
4063+*       SVOERR - ERROR EXIT ADDR FROM SVOLID                          *
4064+*       TSMLES - DATA MANAGEMENT COMMUNICATIONS REGION                *
4065+*       $$ILHD - FIRST BYTE OF INPUT LINE HEADER                      *
4066+*       $$XIND - EXECUTION INDR PASS AREA                             *
4067+*       $$INND - LAST CHARACTER OF INPUT LINE BUFFER                  *
4068+*       $$INLN - FIRST CHARACTER OF INPUT LINE BUFFER                 *
4069+*       $$PRES - ENTRY TO ENABLE KEYBOARD                             *
4070+*       $VOLID - ADDR IN SYSTEM NUCLEUS - VOLUME ID TABLE             *
4071+*       $CAERR - ADDR IN SYSTEM NUCLEUS - ERROR CODE SAVE AREA        *
4072+*       $KEYCD - INDR BYTE CONTAINING KEYBOARD INDR IN SYSTEM NUCLEUS *
4073+*       $CARDI - MASK IN SKEYCD - CARD INPUT MODE                     *
4074+*       $SPRNT - ADDR IN SYSTEM NUCLEUS-SYSTEM PRINTER IOCR INTERFACE *
4075+*       $CIMSK - ADDR IN SYSTEM NUCLEUS-IR MASK ROUTINE INDR          *
4076+*       $WAITF - ADDR IN SYSTEM NUCLEUS-DISK WAITS DPL                *
4077+*       $KYBSY - MASK IN $KEYCD - KEYBOARD BUSY                       *
4078+*       $TRUNK - MASK IN $KEYCD - TRUNCATED LINE INDR                 *
4079+*       $UNHSK - ADDR IN SYSTEM NUCLEUS-ENTRY TO UNMASK IR            *
4080+*                                                                     *
```

```
                               4081+*EXITS, NORMAL                                                        *
                               4082+*   NEXT SEQUENTIAL INSTRUCTION IN CALL ROUTINE.                      *
                               4083+*                                                                     *
                               4084+*EXITS, ERROR                                                         *
                               4085+*   $VOERR - ERROR EXIT ROUTINE IN CALL ROUTINE.                      *
                               4086+*   (NOTE: ERROR PROCEDURES).                                         *
                               4087+*                                                                     *
                               4088+*TABLES/WORK AREAS                                                    *
                               4089+*   CONSTANTS, PPL'S. AND WORK AREAS WHICH ARE ADDRESSED BY THE BASE  *
                               4090+*   REGISTER (@BR) ARE LOCATED TO BE REFERENCED AS SUCH. THOSE        *
                               4091+*   WHICH ARE NOT ADDRESSED BY A BASE REGISTER ARE LOCATED AT THE     *
                               4092+*   END OF THE MODULE.                                                *
                               4093+*                                                                     *
                               4094+*ATTRIBUTES                                                           *
                               4095+*   RELOCATABLE, CONDITIONALLY REUSABLE (SEE OTHER).                  *
                               4096+*                                                                     *
                               4097+*CHARACTER CODE DEPENDENCY                                            *
                               4098+*   CHARACTER CODE DEPENDENCY CLASS - C                               *
                               4099+*   THE OPERATION OF THIS MODULE DEPENDS UPON AN INTERNAL REPRESENTA- *
                               4100+*   TION OF THE EXTERNAL CHARACTER SET WHICH IS EQUIVALENT TO THE ONE *
                               4101+*   USED AT ASSEMBLY TIME. THE CODING HAS BEEN ARRANGED SO THAT RE    *
                               4102+*   DEFINITION OF CHARACTER CONSTANTS, BY REASSEMBLY, WILL RESULT IN  *
                               4103+*   A CORRECT MODULE FOR THE NEW DEFINITIONS. THE FOLLOWING ARE THE   *
                               4104+*   SPECIAL CONSIDERATIONS FOR THIS MODULE:                           *
                               4105+*   * CHARACTER CONSTANT FOR DECIMAL L(ONE)   INTERNAL EQUATE         *
                               4106+*   * CHARACTER CONSTANT FOR DECIMAL 2(TWO)   INTERNAL EQUATE         *
                               4107+*   * @BLANK - PART OF @SYSEQ - FOR SYNTAX CHECK                      *
                               4108+*   * @CHARR - PART OF @SYSEQ - FOR SYNTAX CHECK                      *
                               4109+*   * @CHARF - PART OF @SYSEQ - FOR SYNTAX CHECK                      *
                               4110+*   * @EOS   - PART OF @SYSEQ - FOR SYNTAX CHECK                      *
                               4111+*                                                                     *
                               4112+*NOTES                                                                *
                               4113+*   ERROR PROCEDURES                                                  *
                               4114+*     THE FOLLOWING CONDITIONS WILL CAUSE AN ERROR CODE TO BE PLACED  *
                               4115+*     IN SCAERR AND AN EXIT BRANCH TO BE TAKEN TO SVOERR:             *
                               4116+*     * THE SPECIFIED VOLUME ID IS NOT ON THE SYSTEM.                 *
                               4117+*     * DUPLICATE VOLUME ID'S ARE RTLADO. AND INPUT IS NOT FROM       *
                               4118+*       THE KEYBOARD.                                                 *
                               4119+*     * THE SPECIFIED PHYSICAL ID FROM THE KEYBOARD DOES NOT CONTAIN  *
                               4120+*       ONE OF THE MULTIPLY DEFINED VOLUME ID'S.                      *
                               4121+*     * THE SPECIFIEC OR RESOLVED VOLUME DOES NOT CONTAIN A LIBRARY   *
                               4122+*       AREA.                                                         *
                               4123+*                                                                     *
                               4124+*   REGISTER USAGE                                                    *
                               4125+*     INDEX REGISTER 1 (@BR) IS USED PRIMARILY AS A BASE REGISTER     *
                               4126+*     AND SECONDLY AS AN INDEX IN THE VOL ID TABLE.                   *
                               4127+*     INDEX REGISTER 2 (@XR) IS USED PRIMARILY AS AN INDEX REGISTER   *
                               4128+*     IN THE VOL-ID TABLE AND SECONDLY AS AN INDEX TO SYNTAX CHECK    *
                               4129+*     KEYBOARD INPUT WHEN VOLUMES ARE MULTIPLY DEFINED.               *
                               4130+*                                                                     *
                               4131+*   SAVED/RESTORED AREAS                                              *
                               4132+*     N/A                                                             *
                               4133+*                                                                     *
                               4134+*   MODIFICATION CONSIDERATIONS                                       *
                               4135+*     VOLID'S SEARCH OF THE VOL-ID TABLE (SVOLID) IS TOTALLY          *
                               4136+*     DEPENDENT ON THE FORMAT OF THE TABLE AS IT EXISTS; ESPECIALLY   *
                               4137+*     THE NUMBER OF ENTRIES WHICH NOW EXIST (IE. FOUR).              *
                               4138+*                                                                     *
                               4139+*   REQUIRED MODULES                                                  *
                               4140+*       @CANEQ - COMMON CORE LOCATIONS OUTSIDE SYSTEM NUCLEUS         *
                               4141+*       @DIREQ - SYSTEM LIBRARY DIRECTORY EQUATES                     *
                               4142+*       @ERMEQ - ERROR MESSAGE EQUATES                                *
                               4143+*       @FXDEQ - COMMON CORE LOCATIONS WITHIN THE SYSTEM NUCLEUS      *
                               4144+*       @SYSEQ - COMMON SYSTEM SOFTWARE EQUATES                       *
                               4145+*       TSMLES - DATA MANAGEMENT COMMUNICATION REGIONS                *
                               4146+*                                                                     *
                               4147+*   OTHER                                                             *
                               4148+*     SVOLID MAY BE RE-USED IF THE CALL ROUTINE WILL PRIME 'SVOCT1'   *
                               4149+*     WITH A '4', AND 'SVOCT2' WITH A '0' BEFORE EACH RE-ENTRY.       *
                               4150+*     BOTH OF THESE FIELDS ARE 1 BYTE LONG AND CONTIGUOUS, RESPEC-    *
                               4151+*     TIVELY. (IE. CAN BE INITIALIZED WITH 'MVC' OF X'0400').         *
                               4152+***********************************************************************
                               4154+************************************************************************
                               4155+*                                                                      *
                               4156+*            SVOLID MODULE EQUATES                                     *
                               4157+*                                                                      *
                               4158+************************************************************************
                               4159+*
                          0001 4160+SVOLN1 EQU   1                         LENGTH CODE OF ONE
                          00F1 4161+SVO001 EQU   X'F1'                     CONSTANT OF 1 FOR COMPARE
                          00F2 4162+SVO002 EQU   X'F2'                     CONSTANT OF 2 FOR COMPARE
                          0100 4163+SVOINP EQU   $$XIND-$$ILHD+@B1         LENGTH INPUT BUFFER
                          00FF 4164+SVOEND EQU   $$XIND-$$ILHD             DISP TO END OF SVOBUF

                               4166+************************************************************************
                               4167+*                                                                      *
                               4168+*            INITIALIZATION OF MODULE                                  *
                               4169+*                                                                      *
                               4170+************************************************************************
                               4171+*
                          176B 4172+SVOLID EQU   *                         ENTRY POINT
                          177D 4173+       USING SVOBSE,@BR                BASE ADDRESS
    176B 34 01 17B7            4174+       ST    SVO274+@OP1,@BR           SAVE BASE CONTENTS
    176F C2 01 177D            4175+       LA    SVOBSE,@BR                LOAD BASE ADDRESS
    1773 74 02 3E              4176+       ST    SVO276+@OP1(,@BR),@XR     SAVE INDEX REGISTER
    1776 74 08 46              4177+       ST    SVO290+@OP1(,@BR),@ARR    SAVE RETURN ADDR

                               4179+************************************************************************
                               4180+*                                                                      *
                               4181+*            SEARCH VOL-ID TABLE                                       *
                               4182+*                                                                      *
                               4183+************************************************************************
                               4184+*
    1779 C2 02 1770            4185+       LA    SVOLID+@VOLID-@B1,@XR     LOAD XR AS POINTER INTO NUCLEUS
                          177D 4186+SVOBSE EQU   *
    177D 8D 05 00 1B44         4187+SVO100 CLC   @ZERO(@VOLID,@XR),SMVOID  IS THIS THE VOL-ID ?
    1782 D0 01 11              4188+       BNE   SVO200(,@BR)              NO, CHECK NEXT ENTRY
    1785 1C 01 1B58 02         4189+       MVC   SMBFDA(@DADDR),@DADDR(,@BR) SAVE DADDR-DUPLICATE CHECK
    178A 5E 00 48 49           4190+       ALC   SVOCT2(SVOLN1,@BR),SVOONE(,@BR) INCREMENT COUNT
    178E E2 02 08              4191+SVO200 LA    @VOLID+@DADDR(,@XR),@XR   INCREMENT XR
    1791 5F 00 47 49           4192+       SLC   SVOCT1(SVOLN1,@BR),SVOONE(,@BR) IS THE LAST ENTRY ?
    1795 D0 01 00              4193+       BNZ   SVO100(,@BR)              NO, CHECK NEXT ONE
```

```
                               4195+***********************************************************************
                               4196+*                                                                     *
                               4197+*            PROCESS ENTRY IF FOUND                                   *
                               4198+*                                                                     *
                               4199+***********************************************************************
                               4200+*
    1798 7D 02 48              4201+       CLI   SVOCT2(,@BR),@D1          WAS AN ID FOUND ?
    179B 3C 29 03CD            4202+       MVI   $CAERR,@@E217             ERROR - NO ID FOUND
    179F D0 82 33              4203+       BL    SVO270(,@BR)              NO, ERROR EXIT
    17A2 D0 84 4A              4204+       BH    SVO300(,@BR)              MORE THAN 1 ID

                               4206+***********************************************************************
                               4207+*                                                                     *
                               4208+*            CHECK DISK ADDR OF LIBRARY                               *
                               4209+*                                                                     *
                               4210+***********************************************************************
                               4211+*
    17A5 3D 00 1B57            4212+SVO260 CLI   SMBFDA-@B1,@ZERO          IS THERE A LIBRARY ?
    17A9 F2 01 08              4213+       JNE   SVO274                    YES, RETURN
    17AC 3C 54 03CD            4214+       MVI   $CAERR,@@E351             ERROR - NO LIBRARY
    17B0 3C 87 17BD            4215+SVO270 MVI   SVO280+@Q,@UCB            SET ERROR EXIT

                               4217+***********************************************************************
                               4218+*                                                                     *
                               4219+*            END OF MODULE PROCESSING                                 *
                               4220+*                                                                     *
                               4221+***********************************************************************
                               4222+*
    17B4 C2 01 0000            4223+SVO274 LA    *-*,@BR                   RESTORE BASE REGISTER
    17B8 C2 02 0000            4224+SVO276 LA    *-*,@XR                   RESTORE INDEX REGISTER
                               4225+*
    17BC C0 80 0CF7            4226+SVO280 BC    SVOERR,@NOP               ERROR EXIT
    17C0 C0 87 0000            4227+SVO290 B     *-*                       RETURN
```

```
                               4229+***********************************************************************
                               4230+*                                                                     *
                               4231+*            DATA CONSTANTS, BUFFERS, WORK AREAS AND SAVE AREAS      *
                               4232+*                                                                     *
                               4233+***********************************************************************
                               4234+*
    17C4                  17C4 4235+SVOCT1 DS    CL1                        COUNTER - NUMBER OF DISKS - 4
    17C4                       4236+       ORG   SVOCT1                     RESET FOR INITIALIZATION
    17C4 04               17C4 4237+       DC    XL1'04'                    INITIALIZED TO 4
    17C5                  17C5 4238+SVOCT2 DS    CL1                        COUNTER - DUPLICATE DISK LABELS
    17C5                       4239+       ORG   SVOCT2                     RESET FOR INITIALIZATION
    17C5 00               17C5 4240+       DC    XL1'00'                    INITIALIZED TO ZERO
    17C6 01               17C6 4241+SVOONE DC    XL1'01'                    INITIALIZED TO 1 FOR COUNTER

                               4243+***********************************************************************
                               4244+*                                                                     *
                               4245+*            PROCESS MULTIPLE ENTRIES                                 *
                               4246+*                                                                     *
                               4247+***********************************************************************
                               4248+*
    17C7 38 01 03C3            4249+SVO300 TBN   $KEYCD,$CARDI              IS KEYBOARD INPUT MODE ?
    17CB 3C 25 03CD            4250+SVO310 MVI   $CAERR,@@E212              KEYBOARD NOT INPUT MODE
    17CF D0 10 33              4251+SVO315 BT    SVO270(,@BR)               NO ERROR EXIT
```

```
                               4253+***********************************************************************
                               4254+*                                                                     *
                               4255+*            ASK USER FOR DRIVE CLARIFICATION                         *
                               4256+*                                                                     *
                               4257+***********************************************************************
                               4258+*
                          17D2 4259+SVO320 EQU   *                         PRINT MESSAGES
    17D2 C0 87 0465            4260+       B     $SPRNT                    PRINT MESSAGE
    17D6 0C0A             17D7 4261+       DC    AL2(@@M300)               ERROR MESSAGE PPL
                               4262+*
    17D8 0C 00 17FB 0476       4263+       MVC   SVO335+@VQ(@B1),$CIMSK    OBTAIN CURRENT MASK STATUS
    17DE C0 87 0465            4264+       B     $SPRNT                    WAIT FOR PRINT
    17E2 057F             17E3 4265+       DC    AL2($WAITF)               ADDR OF PPL

                               4267+***********************************************************************
                               4268+*                                                                     *
                               4269+*            MODIFY INPUT BUFFER FOR ACCEPTANCE OF INPUT ANSWER       *
                               4270+*                                                                     *
                               4271+***********************************************************************
                               4272+*
                          17E4 4273+SVO330 EQU   *                         ENABLE INPUT ROUTINE
                               4274+*  SET FOR JUMP AFTER INITIAL SAVE OF INPUT BUFFER
    17E4 F2 80 09              4275+       JC    SVO333,@NOP               SAVE SWITCH
    17E7 0C FF 1C6C 06FF       4276+       MVC   SVOBUF+SVOEND(SVOINP),$$XIND SAVE INPUT BUFFER
    17ED 7C 87 68              4277+       MVI   SVO330+@Q(,@BR),@UCB      SET SWITCH TO BYPASS SAVE
    17F0 3C 40 06FA            4278+SVO333 MVI   $$INND,@BLANK             CLEAR INPUT BUFFER
    17F4 0C F2 06F9 06FA       4279+       MVC   $$INND-@B1($$INND-$$INLN),$$INND
    17FA C0 01 048D            4280+SVO335 BC    $UNMSK,@VQ                BRANCH IF UNMASKED
    17FE C0 87 0890            4281+       B     $$PRES                    GET USER'S RESRONSE
    1802 38 10 03C3            4282+SVO350 TBN   $KEYCD,$KYBSY             IS KEYBOARD BUSY ?
    1806 C0 10 1802            4283+       BT    SVO350                    YES, WAIT
    180A C0 87 0465            4284+       B     $SPRNT                    WAIT FOR PRINTER RETURN
    180E 057F             180F 4285+       DC    AL2($WAITF)               ADDR OF PPL
```

```
                               4287+************************************************************************
                               4288+*                                                                      *
                               4289+*           VERIFY VOL-ID ON DRIVE SPECIFIED                          *
                               4290+*                                                                      *
                               4291+************************************************************************
                               4292+*
    1810 C2 02 0606            4293+       LA    $$INLN-@B1,@XR            ADDR FIRST RESPONSE BYTE
    1814 C2 01 1770            4294+       LA    SVOLID+@VOLID-@B1,@BR     REFERENCE POINT FOR THE VOLID
                               4295+*
    1818 E2 02 01              4296+SVO360 LA    @B1(,@XR),@XR             INDEX BY BLANK
    181B BD 40 00              4297+       CLI   @ZERO(,@XR),@BLANK        IS IT A BLANK ?
    181E C0 81 1818            4298+       BE    SVO360                    YES, CHECK NEXT BYTE
                               4299+*
    1822 BD F1 01              4300+       CLI   @B1(,@XR),SVO001          IS IT DRIVE 1 ?
    1825 F2 81 0A              4301+       JE    SVO400                    YES, CHECK DISK TYPE
    1828 BD F2 01              4302+       CLI   @B1(,@XR),SVO002          IS IT DRIVE 2 ?
    182B C0 01 17D2            4303+       BNE   SVO320                    NO, ASK USER AGAIN
    182F D2 01 10              4304+       LA    2*@VOLID+2*@DADDR(,@BR),@BR SET INDEX FOR DRIVE 2
    1832 BD D9 00              4305+SVO400 CLI   @ZERO(,@XR),@CHARR        IS IT REMOVABLE ?
    1835 F2 81 0A              4306+       JE    SVO440
    1838 BD C6 00              4307+       CLI   @ZERO(,@XR),@CHARF        IS IT FIXED ?
    183B C0 01 17D2            4308+       BNE   SVO320                    ASK AGAIN
    183F D2 01 08              4309+       LA    @VOLID+@DADDR(,@BR),@BR   SET INDEX FOR FIXED
    1842 E2 02 01              4310+SVO440 LA    @B1(,@XR),@XR             INCREMENT TO NEXT BYTE
    1845 E2 02 01              4311+SVO445 LA    @B1(,@XR),@XR             INCREMENT TO NEXT BYTE
    1848 BD 40 00              4312+       CLI   @ZERO(,@XR),@BLANK        IS IT A BLANK ?
    184B C0 81 1845            4313+       BE    SVO445                    YES, CHECK NEXT BYTE
                               4314+*
    184F BD 1E 00              4315+       CLI   @ZERO(,@XR),@EOS          AT EOS ?
    1852 C0 01 17D2            4316+       BNE   SVO320                    ASK AGAIN
                               4317+*
    1856 0C FF 06FF 1C6C       4318+       MVC   $$XIND(SVOINP),SVOBUF+SVOEND RESTORE INPUT
    185C 4D 05 00 1B44         4319+SVO450 CLC   @ZERO(@VOLID,@BR),SMVOID  IS IT THE VOLID ?
    1861 3C 28 03CD            4320+       MVI   $CAERR,@@E216             VOLUME NOT ON THAT DRIVE
    1865 C0 01 17B0            4321+       BNE   SVO270                    NO, ERROR EXIT
```

```
                               4323+***********************************************************************
                               4324+*                                                                     *
                               4325+*            SAVE VOL-ID LIBRARY ADDR                                 *
                               4326+*                                                                     *
                               4327+***********************************************************************
                               4328+*
    1869 1C 01 1B58 02         4329+       MVC   SMBFDA(@DADDR),@DADDR(,@BR) SAVE LIBRARY ADDR
    186E 3B 80 03C3            4330+       SBF   $KEYCD,$TRUNK             SET OFF RM EXCEEDED INDR
    1872 C0 87 17A5            4331+       B     SVO260                    NORMAL EXIT
                               4332+***                       END OF SVOLID                             ***
                               4333 *      $UFFE
                               4335+***********************************************************************
                               4336+* 5703-XM1  COPYRIGHT IBM CORP. 1970                                  *
                               4337+*          REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE  120-2083       *
                               4338+*                                                                     *
                               4339+***********************************************************************
                               4340+*STATUS                                                               *
                               4341+*   VERSION 1 MODIFICATION 0                                          *
                               4342+*                                                                     *
                               4343+*FUNCTION                                                             *
                               4344+*   THE FUNCTION OF SUFFER IS TO SYNTAX CHECK A FILE SPECIFICATION    *
                               4345+*   AND SCAN TO THE FIRST NON-DELIMITER FOLLOWING A VALID ONE.        *
                               4346+*   A SPECIFICATION CAN CONSIST OF ANY OF THE FOLLOWING:              *
                               4347+*   *   FILENAME / PASSWORD / VOL-D                                   *
                               4348+*   *   FILENAME / PASSWORD                                           *
                               4349+*   *   FILENAME                                                      *
                               4350+*   *   **FILENAME / VOL-ID                                           *
                               4351+*   *   **FILENAME                                                    *
                               4352+*   *   *FILENAME / VOL-ID                                            *
                               4353+*   *   *FILENAME                                                     *
                               4354+*                                                                     *
                               4355+*ENTRY POINTS                                                         *
                               4356+*   SUFFER - FIRST LOCATION IN PROGRAM. SUFFER EXPECTS INDEX          *
                               4357+*            REGISTER 2 (@XR) TO BE ADDRESSING THE LEFTMOST CHARACTER *
                               4358+*            OF THE FILE SPECIFICATION. THE CALLING SEQUENCE IS:      *
                               4359+*                    B     SUFFER                                     *
                               4360+*                                                                     *
                               4361+*INPUT                                                                *
                               4362+*   INPUT TO SUFFER IS INDE, REGISTER 2 (@XR) ADDRESSING THE LEFTMOST *
                               4363+*   CHARACTER OF THE FILE-SPECIFICATION TO BE SYNTAX CHECKED.         *
                               4364+*                                                                     *
                               4365+*OUTPUT                                                               *
                               4366+*   OUTPUT FROM SUFFER UPON NORMAL EXIT IS INDEX REGISTER 2 (@XR)     *
                               4367+*   ADDRESSING THE FIRST NON-DELIMITER FOLLOWING THE FILE SPECIFICA-  *
                               4368+*   TION. THE FILENAME WILL BE SAVED IN SMFNAM IN TSMLES. THE PASS-   *
                               4369+*   WORD IF SPECIFIED WILL BE SAVED IN SMPSWD 1N TSMLES, OTHERWISE IT *
                               4370+*   WILL BE BLANKS. (NOTE: ** OR * FILENAMES, WHEN SPECIFIED, WILL    *
                               4371+*   CAUSE THE *'S TO BE SAVED IN SMPSWD). THE VOL-ID, IF SPECIFIED,   *
                               4372+*   WILL BE SAVED IN SMVOID IN TSMLES, OTHERWISE A BLANK IS MOVED     *
                               4373+*   TO SMVOID AS AN INDICATOR.                                        *
                               4374+*   OUTPUT FROM SUFFER UPON ERROR EXIT IS INDEX REGISTER 2 (@XR)      *
                               4375+*   ADDRESSING THE INVALID CHARACTER (SEE EXITS,ERROR). THE PROGRAM   *
                               4376+*   STATUS REGISTER (@PSR) WILL CONTAIN A LOW CONDITION CODE.         *
                               4377+*                                                                     *
                               4378+*EXTERNAL REFERENCES                                                  *
                               4379+*        SALPHR - ADDR IN SALPHA - SYNTAX CHECKED PARAMETER           *
                               4380+*        SALPH6 - ENTRY TO SALPHA - SYNTAX CHECK VOL-ID               *
                               4381+*        SALPH8 - ENTRY TO SALPHA - SYNTAX CHECK PASSWORD; FILENAME   *
                               4382+*        SAL375 - SAVE AREA IN SALPHA - ERROR POINTER SAVE AREA       *
                               4383+*        SCANIT - DELIMITER SCAN MODULE                               *
                               4384+*        SCAMMA - SWITCH IN SCANIT - DELIMITER SCAN TYPE INDR         *
                               4385+*        SCACOF - MASK IN SCANIT TO BYPASS BLANKS ONLY                *
                               4386+*        SCACOM - MASK IN SCANIT - BYPASS 1 COMMA                     *
                               4387+*        SCACNT - COUNTER IN SCANIT - NUMBER OF SCANNED BLANKS        *
                               4388+*        TSMLES - DATA MANAGEMENT COMMUNICATIONS REGIONS              *
                               4389+*        $CAERR - ADDR IN SYSTEM NUCLEUS-ERROR CODE SAVE AREA         *
                               4390+*                                                                     *
                               4391+*EXITS, NORMAL                                                        *
                               4392+*   NEXT SEQUENTIAL INSTRUCTION IN CALL ROUTINE. INDEX REGISTER       *
                               4393+*   2 (@XR) WILL BE ADDRESSING THE FIRST NON-DELIMITER FOLLOWING      *
                               4394+*   THE FILE SPECIFICATION. THE PROGRAM STATUS REGISTER (@PSR)        *
                               4395+*   WILL CONTAIN A NON-LOW CONDITION CODE.                            *
                               4396+*                                                                     *
                               4397+*EXITS, ERROR                                                         *
                               4398+*   NEXT SEQUENTIAL INSTRUCTION IN CALL ROUTINE. INDEX REGISTER       *
                               4399+*   2 (@XR) WILL BE ADDRESSING THE LEFTMOST BYTE OF AN INVALID        *
                               4400+*   PARAMETER OR WILL BE ADDRESSING AN INVALID DELIMITER IN THE       *
                               4401+*   FILE SPECIFICATION. THE PROGRAM STATUS REGISTER (@PSR)            *
                               4402+*   WILL CONTAIN A LOW CONDITION CODE.                                *
                               4403+*   T                                                                 *
                               4404+*TABLES/WORK AREAS                                                    *
                               4405+*   SUFFER DOES NOT CONTAIN ANY TABLES OR WORK AREAS.                 *
                               4406+*                                                                     *
                               4407+*ATTRIBUTES                                                           *
                               4408+*   RELOCATABLE,REUSABLE                                              *
                               4409+*                                                                     *
                               4410+*CHARACTER CODE DEPENDENCY                                            *
                               4411+*   CHARACTER CODE DEPENDENCY CLASS - C                               *
                               4412+*   THE OPERATION OF THIS MODULE DEPENDS UPON AN INTERNAL REPRESENTA- *
                               4413+*   TION OF THE EXTERNAL CHARACTER SET WHICH IS EQUIVALENT TO THE     *
                               4414+*   USED AT ASSEMBLY TIME. THE CODING HAS BEEN ARRANGED SO  THAT RE-  *
                               4415+*   DEFINITION OF CHARACTER CONSTANTS, BY REASSEMBLY, WILL RESULT IN  *
                               4416+*   A CORRECT MODULE FOR THE NEW DEFINITIONS. THE FOLLOWING ARE THE   *
                               4417+*   SPECIAL CONSIDERATIONS FOR THIS MODULE:                           *
                               4418+*      * @ASTER - PART OF @SYSEQ                                      *
                               4419+*      * @SLASH - PART OF @SYSEQ                                      *
                               4420+*      * @COMMA - PART OF @SYSEQ                                      *
                               4421+*      * @EOS   - PART OF @SYSEQ                                      *
                               4422+*      * @BLANK - PART OF @SYSEQ                                      *
                               4423+*      * CHARACTER LEFT PARENTHESIS - C'('                            *
                               4424+*                                                                     *
                               4425+*NOTES                                                                *
                               4426+*   ERROR PROCEDURES                                                  *
                               4427+*      THE FOLLOWING ERROR CONDITIONS WILL CAUSE SUFFER TO RETURN A   *
                               4428+*      LOW CONDITION CODE TO THE CALL ROUTINE AND INDEX REGISTER 2    *
                               4429+*      (@XR) ADDRESSING THE ERROR:                                    *
                               4430+*         * ANY ERROR RETURNED FROM SALPHA (NOTE SALPHA ERRORS).      *
                               4431+*         * ANY ERROR RETURNED FROM SCANIT (NOTE SCANIT ERRORS).      *
                               4432+*         * ANY INVALID DELIMITER FOLLOWING THE SPECIFICATION         *
                               4433+*         * ANY INVALID PARAMETER WITHIN THE SPECIFICATION.           *
                               4434+*         NOTE MODIFICATION CONSIDERATIONS.                           *
                               4435+*                                                                     *
                               4436+*   REGISTER USAGE                                                    *
                               4437+*      INDEX RESISTER 1 (@BR) IS SAVED AND RESTORED FOR THE CALL      *
                               4438+*      ROUTINE AND USED AS A BASE FOR ADDRESSING WITHIN THE MODULE.   *
                               4439+*      INDEX REGISTER 2 (@XR) IS USED AS AN INDEX TO SCAN THE FILE    *
                               4440+*      SPECIFICATION.                                                 *
                               4441+*                                                                     *
                               4442+*   SAVED/RESTORED AREAS                                              *
                               4443+*      N/A                                                            *
                               4444+*                                                                     *
                               4445+*   MODIFICATION CONSIDERATIONS                                       *
                               4446+*      SUFFER'S NORMAL DELIMITER SCAN UPON EXIT ALLOWS ONLY BLANKS    *
                               4447+*      AND 1 COMMA FOLLOWING THE FILE-SPECIFICATION. AN EXCEPTION     *
                               4448+*      TO THIS USE (UTILIZED BY THE MODULE KALLOC) IS THE OPTION OF   *
                               4449+*      HAVING A LEFT PARENTHESIS IE. '(' FOLLOWING THE FILE SPECI-    *
                               4450+*      FICATION INSTEAD OF A COMMA. THIS USE IS EFFECTED BY           *
                               4451+*      MODIFYING THE Q-CODE OF THE INSTRUCTION LABELED SUF625 WITH A  *
                               4452+*      BRANCH EQUAL CONDITION CODE.                                   *
                               4453+*                                                                     *
                               4454+*   REQUIRED MODULES                                                  *
                               4455+*      SALPHA - FILENAME, PASSWORD, VOL-ID ALPHAMERIC SYNTAX CHECKER  *
                               4456+*      SCANIT - DELIMITER SCAN ROLTINE                                *
                               4457+*      TSMLES - DATA MANAGEMENT COMMUNICATION REGIONS                 *
                               4458+*      @DIREQ - SYSTEM LIBRARY DIRECTORY EQUATES                      *
                               4459+*      @ERMEQ - ERROR MESSAGE EQUATES                                 *
                               4460+*      @FXDEQ - COMMON CORE LOCATIONS WITHIN THE SYSTEM NUCLEUS       *
                               4461+*      @SYSEQ - COMMON SYSTEM SOFTWARE EQUATES                        *
                               4462+*                                                                     *
                               4463+*   OTHER                                                             *
                               4464+*      N/A                                                            *
                               4465+***********************************************************************
                               4467+***********************************************************************
                               4468+*
                               4469+*            INITIALIZATION OF MODULE
                               4470+*
                               4471+***********************************************************************
                               4472+*
                               4473+*SUFFER ENTER BASE-SUFBSE,EXIT-SUFND,@BR,,@ARR
                          18A9 4474+       USING SUFBSE,@BR                BASE ADDRESS SPECIFICATION
                          1876 4475+SUFFER EQU   *                         MODULE ENTRY POINT
    1876 34 01 193A            4476+       ST    SUFND0+@OP1,@BR           SAVE @BR
    187A C2 01 18A9            4477+       LA    SUFBSE,@BR                LOAD BASE REGISTER
    187E 74 08 95              4478+       ST    SUFND2+@OP1(,@BR),@ARR    SAVE RETURN ADDRESS
                               4479+*** END OF EXPANSION ***

                               4481+***********************************************************************
                               4482+*
                               4483+*            INITIALIZE FIELDS IN TSMLES
                               4484+*
                               4485+***********************************************************************
                               4486+*
    1881 3C 40 1B4C            4487+       MVI   SMPSWD,@BLANK             BLANK ALL OF PASSWORD FIELD
    1885 0C 06 1B4B 1B4C       4488+       MVC   SMPSWD-@B1(##LPEN-@B1),SMPSWD
    188B 3C 40 1B3F            4489+       MVI   SMVOID-@VOLID+@B1,@BLANK  BLANK FIRST BYTE OR VOL-1D

                               4491+***********************************************************************
                               4492+*
                               4493+*            CHECK FOR AND PROCESS POOLED AND IBM FILENAMES
                               4494+*
                               4495+***********************************************************************
                               4496+*
    188F BD 5C 00              4497+       CLI   @ZERO(,@XR),@ASTER        ASTERISK IN FILENAME ?
    1892 F2 01 14              4498+       JNE   SUF100                    NO, PROCESS FILENAME
    1895 3C 5C 1B45            4499+       MVI   SMPSWD-##DPEN,@ASTER      SAVE * IN SMPSWD
    1899 E2 02 01              4500+       LA    @B1(,@XR),@XR             INCREMENT XR BY ONE
    189C BD 5C 00              4501+       CLI   @ZERO(,@XR),@ASTER        ASTERISK IN FILENAME ?
    189F F2 01 07              4502+       JNE   SUF100                    NO, PROCESS FILENAME
    18A2 3C 5C 1B46            4503+       MVI   SMPSWD-##DPEN+@B1,@ASTER  SAVE * IN SMPSWD
    18A6 E2 02 01              4504+       LA    @B1(,@XR),@XR             INCREMENT XR BY ONE

                               4506+***********************************************************************
                               4507+*
                               4508+*            PROCESS FILENAME
                               4509+*
                               4510+***********************************************************************
                               4511+*
                          18A9 4512+SUFBSE EQU   *                         BASE ADDR IN MODULE
    18A9 3C 87 1C05            4513+SUF100 MVI   SCAMMA,SCACOF             PRIME SCANIT
    18AD C0 87 193F            4514+       B     SALPH8                    SYNTAX CHECK FILENAME
    18B1 D0 82 85              4515+       BL    SUF750(,@BR)              TAKE ERROR EXIT
    18B4 0C 07 1B54 1A05       4516+       MVC   SMFNAM(##LUEN),SALPHR+##DUEN SAVE FILENAME
    18BA BD 61 00              4517+       CLI   @ZERO(,@XR),@SLASH        IS A SLASH DELIMITER PRESENT ?
    18BD F2 01 35              4518+       JNE   SUF600                    NO, RETURN TO USER
    18C0 3D 5C 1B45            4519+       CLI   SMPSWD-##DPEN,@ASTER      SHOULD A PASSWORD BE CHECKED?
    18C4 F2 81 1A              4520+       JE    SUF200                    NO, CHECK VOL-ID

                               4522+***********************************************************************
                               4523+*
                               4524+*            PROCESS PASSWORD
                               4525+*
                               4526+***********************************************************************
                               4527+*
    18C7 E2 02 01              4528+       LA    @B1(,@XR),@XR             INCREMENT XR BY ONE
    18CA C0 87 1BE8            4529+       B     SCANIT                    BYPASS BLANKS
    18CE C0 87 193F            4530+       B     SALPH8                    SYNTAX CHECK PASSWORD
    18D2 D0 82 85              4531+       BL    SUF750(,@BR)              TAKE ERROR EXIT
    18D5 0C 07 1B4C 1A05       4532+       MVC   SMPSWD(##LPEN),SALPHR+##DPEN SAVE PASSWORD
    18DB BD 61 00              4533+       CLI   @ZERO(,@XR),@SLASH        IS SLASH DELIMITER PRESENT ?
    18DE F2 01 14              4534+       JNE   SUF600                    NO, RETURN TO USER

                               4536+***********************************************************************
                               4537+*
                               4538+*            PROCESS VOL-ID
                               4539+*
                               4540+***********************************************************************
                               4541+*
    18E1 E2 02 01              4542+SUF200 LA    @B1(,@XR),@XR             INCREMENT XR BY ONE
    18E4 C0 87 1BE8            4543+       B     SCANIT                    BYPASS BLANKS
    18E8 C0 87 1943            4544+       B     SALPH6                    SYNTAX CHECK VOL-ID
    18EC D0 82 85              4545+SUF400 BL    SUF750(,@BR)              TAKE ERROR EXIT
    18EF 0C 05 1B44 1A03       4546+       MVC   SMVOID(@VOLID),SALPHR+@VOLID-@B1 SAVE VALID
    18F5 BD 4D 00              4547+SUF600 CLI   @ZERO(,@XR),C'('          IS THIS '(' ?
    18F8 F2 80 39              4548+SUF625 JC    SUF800,@NOP               JUMP IF '(' VALID ADJACENT
    18FB 3D 00 1C28            4549+       CLI   SCACNT,@ZERO              ANY BLANKS SCANNED ?
    18FF F2 01 0C              4550+       JNE   SUF650                    YES, CONTINUE DELIMITER SCAN
    1902 BD 1E 00              4551+       CLI   @ZERO(,@XR),@EOS          IS IT EOS ?
    1905 F2 81 2C              4552+       JE    SUF800                    YES, RETURN
    1908 BD 6B 00              4553+       CLI   @ZERO(,@XR),@COMMA        IS IT A COMMA ?
    190B F2 01 18              4554+       JNE   SUF680                    NO, ERROR EXIT
                               4555+*
    190E 34 02 1993            4556+SUF650 ST    SAL375+@OP1,@XR           SAVE ERROR POINTER
    1912 3C 01 1C05            4557+       MVI   SCAMMA,SCACOM             MODIFY SCANIT TO BYPASS COMMA
    1916 C0 87 1BE8            4558+       B     SCANIT                    BYPASS DELIMITERS
    191A F2 82 11              4559+       JL    SUF750                    ERROR - RETURN

                               4561+***********************************************************************
                               4562+*
                               4563+*            MODIFY PSR FOR ERROR INDICATION
                               4564+*
                               4565+***********************************************************************
                               4566+*
    191D BD 4D 00              4567+       CLI   @ZERO(,@XR),C'('          IS IT '(' ?
    1920 F2 01 11              4568+       JNE   SUF800                    NO, RETURN
    1923 7C 18 7E              4569+       MVI   SUF680+@Q(,@BR),@@E139    INVALID DELIMITER
    1926 3C 00 03CD            4570+SUF680 MVI   $CAERR,*-*                ERROR CODE
    1926                       4571+       ORG   SUF680                    INITIALIZE INSTRUCTION
    1926 3C 11 03CD            4572+       MVI   $CAERR,@@E131             INVALID PARAMETER
                               4573+*
    192A 35 02 1993            4574+       L     SAL375+@OP1,@XR           RESTORE ERROR POINTER
    192E 75 04 44              4575+SUF750 L     SUF400+@Q(,@BR),@PSR      LOAD CONDITION LOW INTO PSR
    1931 F2 87 03              4576+SUF780 J     SUFND0                    ERROR EXIT

                               4578+***********************************************************************
                               4579+*
                               4580+*            END OF MODULE PROCESSING
                               4581+*
                               4582+***********************************************************************
                               4583+*
    1934 75 04 89              4584+SUF800 L     SUF780+@Q(,@BR),@PSR      LOAD CODE FOR NORMAL EXIT
                               4585+*SUFND  EXIT  @BR,,RETURN
    1937 C2 01 0000            4586+SUFND0 LA    *-*,@BR                   RESTORE @BR
    193B C0 87 0000            4587+SUFND2 B     *-*                       RETURN TO CALLING PROGRAM
                               4588+*** END OF EXPANSION ***
                               4589+***                       END OF SUFFER                             ***
                               4590 *      $ALPH
                               4592+***********************************************************************
                               4593+*  5703-XM1    COPYRIGHT IBM CORP. 1970                               *
                               4594+*              REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083    *
                               4595+*                                                                     *
                               4596+***********************************************************************
                               4597+*STATUS                                                               *
                               4598+*   VERSION 1 MODIFICATION 0                                          *
                               4599+*                                                                     *
                               4600+*FUNCTION                                                             *
                               4601+*   THE FUNCTION OF SALPHA IS TO SYNTAX CHECK AN 8 CHARACTER OR 6     *
                               4602+*   CHARACTER ALPHAMERIC PARAMETER DETERMINED BY THE ENTRY POINT,     *
                               4603+*   SALPH8 OR SALPH6 RESPECTIVELY. ENTRY AT SALPHA IMPLIES A REQUEST  *
                               4604+*   THAT THE FIRST CHARACTER BE ALPHABETIC. A SYNTACTICALLY CORRECT   *
                               4605+*   PARAMETER WILL BE SAVED AT SALPHR (LEFTMOST BYTE ADDRESS), THE    *
                               4606+*   COUNT OF THE NUMBER OF VALID CMARACTERS, IF NEEDED, IS FOOD IN    *
                               4607+*   SALCNT. UPON ENTRY, SALPHA REQUIRES INDEX RESISTER 2 (OM TO BE    *
                               4608+*   ADDRESSING THE FIRST CHARACTER 0, THE PARAMETER TO BE SYNTAX      *
                               4609+*   CHECKED. UPON NORMAL RETURN INDEX REGISTER 2 (@XR) WILL BE        *
                               4610+*   ADDRESSING THE FIRST NON-DELIMITER FOLLOWING THE PARAMETER (NOTE  *
                               4611+*   INPUT),                                                           *
                               4612+*                                                                     *
                               4613+*ENTRY POINTS                                                         *
                               4614+*   * SALPH8 - ENTRY POINT TO SYNTAX CHECK AN EIGHT CHARACTER         *
                               4615+*              ALPHAMERIC PARAMETER WHOSE FIRST CHARACTER MUST BE     *
                               4616+*              ALPHABETIC.                                            *
                               4617+*   * SALPH6 - ENTRY POINT TO SYNTAX CHECK A SIX CHARACTER            *
                               4618+*              ALPHAMERIC PARAMETER WHICH HAS NO RESTRICTIONS ON      *
                               4619+*              THE TYPE OF THE FIRST CHARACTER. (NOTE MODIFICA-       *
                               4620+*              TION CONSIDERATIONS)                                   *
                               4621+*                                                                     *
                               4622+*INPUT                                                                *
                               4623+*   UPON ENTRY TO SALPHA, AT EITHER ENTRY POINT, INDEX REGISTER 2     *
                               4624+*   (@XR) SHOULD BE ADDRESSING THE LEFTMOST CHARACTER OF THE PARAMETER*
                               4625+*   TO BE SYNTAX CHECKED. ALSO, THE SWITCH 'SCAMMA' IN SCANIT SHOULD  *
                               4626+*   BE SET FOR THE TYPE OF DELIMITER SCAN REQUESTED AFTER THE SYNTAX  *
                               4627+*   CHECK. (IE. BLANKS ONLY OR BLANKS WITH 1 COMMA).                  *
                               4628+*                                                                     *
                               4629+*OUTPUT                                                               *
                               4630+*   OUTPUT FROM SALPHA INCLUDES THE SYNTAX CHECKED PARAMETER AT SALPHR*
                               4631+*   (LEFTMOST BYTE OF SAVE AREA) AND THE COUNT OF VALID CHARACTERS    *
                               4632+*   IN SALCNT, AND INDEX REGISTER 2 (@XR) WILL BE POINTING AT THE     *
                               4633+*   FIRST NON-DELIMITER AFTER THE PARAMETER. THE ONLY EXCEPTION TO    *
                               4634+*   THIS IS UPON DETECTION OF AN ERROR (SEE ERROR EXITS AND PROC.)    *
                               4635+*                                                                     *
                               4636+*EXTERNAL REFERENCES                                                  *
                               4637+*        SCANIT - DELIMITER SCAN MODULE                               *
                               4638+*        $CAERR - ADDR IN SYSTEM NUCLEUS-ERROR CODE SAVE AREA         *
                               4639+*                                                                     *
                               4640+*EXITS, NORMAL                                                        *
                               4641+*   NEXT SEQUENTIAL INSTRUCTION IN CALL ROUTINE WITH INDEX            *
                               4642+*   REGISTER 2 (@XR) POINTING TO THE NEXT NON-DELIMITER               *
                               4643+*   FOLLOWING THE PARAMETER AND WITH A NON-LOW CONDITION CODE         *
                               4644+*   IN THE PROGRAM STATUS RESISTER (@PSR),                            *
                               4645+*                                                                     *
                               4646+*EXITS, ERROR                                                         *
                               4647+*   NEXT SEQUENTIAL INSTRUCTION IN CALL ROUTINE WILH INDEX            *
                               4648+*   REGISTER 2 (@XR) POINTING TO THE LEFTMOST CHARACTER OF THE        *
                               4649+*   INVALID PARAMETER AND WITH A LOW CONDITION CODE IN THE            *
                               4650+*   PROGRAM STATUS REGISTER (@PSR),                                   *
                               4651+*                                                                     *
                               4652+*TABLES/WORK AREAS                                                    *
                               4653+*   ALL OF THE CONSTANTS AND WORK AREAS IN SALPHA ARE LOCATED AT THE  *
                               4654+*   END OF THE MODULE AND ARE ADDRESSED BY INDEX REGISTER 1 (RBR).    *
                               4655+*                                                                     *
                               4656+*ATTRIBUTES                                                           *
                               4657+*   REUSABLE, RELOCATABLE                                             *
                               4658+*                                                                     *
                               4659+*CHARACTER CODE DEPENDENCY                                            *
                               4660+*   CHARACTER CODE DEPENDENCY CLASS - E                               *
                               4661+*   THE OPERATION OF THIS MODULE DEPENDS UPON THE FOLLOWING PROPERTIES*
                               4662+*   OF THE INTERNAL REPRESENTATION OF THE EXTERNAL CHARACTER SET:     *
                               4663+*   * THE FOLLOWING SPECIAL ALPHABETIC CHARACTERS ARE PART OF         *
                               4664+*     @SYSEQ AND ARE SPECIFICALLY COMPARED FOR:                       *
                               4665+*       * @DOLAR                                                      *
                               4666+*       * @NUMBR                                                      *
                               4667+*       * @ASIGN                                                      *
                               4668+*   * THE REMAINING-ALPHABETIC CHARACTERS ARE DEFINED TO BE           *
                               4669+*     INCLUSIVELY IN THE RANGE DEFINED BY THE FOLLOWING IN @SYSEQ:    *
                               4670+*       * @CHARA                                                      *
                               4671+*       * @CHARZ                                                      *
                               4672+*                                                                     *
                               4673+*   THE DECIMAL NUMBERS FALL INTO THE CATEGORY OF BEING GREATER       *
                               4674+*   THAN AN @CHARZ (IE. THIS IS DEFAULTED TO BY CHECKING METHOD)      *
                               4675+*   THE SPECIFIC INSTRUCTIONS WHICH REQUIRE MODIFICATION IF THESE     *
                               4676+*   PROPERTIES OF THE CHARACTER SET ARE CHANGED MAY BE IDENTIFIED BY: *
                               4677+*     * SAL200 - FOR THE THREE SPECIAL CHARACTERS                     *
                               4678+*     * SAL250 - FOR THE REMAINING ALPHABETIC RANGE                   *
                               4679+*     * SAL425 - BRANCHES 'TO' THIS LOCATION IMPLY DEFAULT TO NUMERIC *
                               4680+*                                                                     *
                               4681+*NOTES                                                                *
                               4682+*   ERROR PROCEDURES                                                  *
                               4683+*     THE FOLLOWING ERROR CONDITIONS WILL RESULT IN AN ERROR CODE     *
                               4684+*     BEING SET IN $CAERR AND AN ERROR EXIT BEING MADE (SEE EDITS,    *
                               4685+*     ERROR):                                                         *
                               4686+*     * A NON-ALPHABETIC FIRST CHARACTER WHEN ENTRY WAS AT            *
                               4687+*       SALPH8.                                                       *
                               4688+*     * A NON-ALPHAMERIC CHARACTER EMBEDDED IN A PARAMETER WHICH      *
                               4689+*       SALPH8 WAS CALLED TO CHECK.                                   *
                               4690+*     * A NON-ALPHAMERIC CHARACTER BEING FIRST OR EMBEDDED IN A       *
                               4691+*       PARAMETER WHICH SALPH6 WAS CALLED TO CHECK.                   *
                               4692+*     * A PARAMETER OF GREATER THAN EIGHT CHARACTERS WHEN ENTRY       *
                               4693+*       WAS AT SALPH8.                                                *
                               4694+*     * A PARAMETER OF GREATER THAN SIX CHARACTERS WHEN ENTRY         *
                               4695+*       WAS AT SALPH6.                                                *
                               4696+*                                                                     *
                               4697+*   REGISTER USAGE                                                    *
                               4698+*     INDEX REGISTER 1 (@BR) IS USED AS A BASE REGISTER THROUGHOUT    *
                               4699+*     THE EXEQUTION OF THE MODULE. IT IS SAVED FOR THE CALL PROGRAM   *
                               4700+*     UPON ENTRY AND RESTORED UPON EXIT.                              *
                               4701+*     INDEX REGISTER 2 (@XR) IS USED AS A PARAMETER PASSING REGISTER. *
                               4702+*     UPON ENTRY IT CONTAINS THE ADDRESS OF THE LEFTMOST BYTE OF      *
                               4703+*     PARAMETER TO BE SYNTAX CHECKED AND UPON EXIT IT CONTAINS THE    *
                               4704+*     ADDRESS OR THE FIRST NON-DELIMITER FOLLOWING THE PARAMETEP.     *
                               4705+*     (NOTE ERROR EXITS AND PROCEDURES),                              *
                               4706+*                                                                     *
                               4707+*   SAVED/RESTORED AREAS                                              *
                               4708+*     N/A                                                             *
                               4709+*                                                                     *
                               4710+*   MODIFICATION CONSIDERATIONS                                       *
                               4711+*     BECAUSE OF ITS CHARACTER CODE DEPENDENCY AND PARAMETER LENGTH   *
                               4712+*     QUALIFICATIONS, ONE MUST TAKE SPECIAL CARE IN MODIFYING SALPHA, *
                               4713+*     ESPECIALLY THE CONSTANTS AND WORK AREAS AND THEIR RE-INITIAL,   *
                               4714+*     IZATION. SALPHA IS MOST COMMONLY USED TO SYNTAX FILENAMES,      *
                               4715+*     PASSWORDS, AND VOL-IDS AND IS THEREFORE USED BY THE MODULE      *
                               4716+*     SUFFER (FILE SPECIFICATION SYNTAX CHECKER). THEREFORE, ANY      *
                               4717+*     SIGNIFICANT CHANGE IN SALPHA WILL REQUIRE AN INVESTIGATION INTO *
                               4718+*     ITS USE AND IMPACT ON SUFFER.                                   *
                               4719+*        SPECIAL NOTE: AN IRREGULAR USE OF SALPHA WHICH CAN BE        *
                               4720+*        EFFECTED IS THE SYNTAY CHECK OF A PARAMETER WITH A MAXIMUM   *
                               4721+*        OF 10 CHARACTERS. THIS IS DONE BY MODIFYING THE Q-CODE OF    *
                               4722+*        THE INSTRUCTION AT SAL450 PRIOR TO ENTRANCE AT SALPH6, WITH  *
                               4723+*        X'0A' OR ITS EQUIVALENT. (NOTE: ONE SUCH MODULE WHICH        *
                               4724+*        USES THIS OPTION IS UINITL)                                  *
                               4725+*                                                                     *
                               4726+*   REQUIRED MODULES                                                  *
                               4727+*     SCANIT - DELIMITER SCAN ROUTINE                                 *
                               4728+*     @DIREQ - SYSTEM LIBRARY DIRECTORY EQUATES                       *
                               4729+*     @ERMEQ - ERROR MESSAGE EQUATES                                  *
                               4730+*     @FXDEQ - COMMON CORE LOCATIONS WITHIN THE SYSTEM NUCLEUS        *
                               4731+*     @SYSEQ - COMMON SYSTEM SOFTWARE EQUATES                         *
                               4732+*                                                                     *
                               4733+*   OTHER                                                             *
                               4734+*     N/A                                                             *
                               4735+***********************************************************************

                               4737+***********************************************************************
                               4738+*
                               4739+*                  SALPNA MODULE EQUATES
                               4740+*
                               4741+***********************************************************************
                          0008 4742+SALCT8 EQU   ##LUEN                    COUNT COMPARE FIELD
                               4743+*
                          0006 4744+SALCT6 EQU   @VOLID                    COUNT COMPARE FIELD

                               4746+***********************************************************************
                               4747+*
                               4748+*                  INITIALIZATION OF MODULE
                               4749+*
                               4750+***********************************************************************

                               4752+*SALPH8 ENTER CHECK                     FILENAME OR PASSWORD
                          193F 4753+SALPH8 EQU   *                         MODULE ENTRY POINT
                               4754+*** END OF EXPANSION ***
    193F 3A 80 19FA            4755+       SBN   SALIDR,SAL008             SET ON SALPH8 INDR
                               4756+*
                               4757+*SALPH6 ENTER BASE-SALBSE,EXIT-SALND,@BR,,@ARR  VOL-ID CHECK
                          195F 4758+       USING SALBSE,@BR                BASE ADDRESS SPECIFICATION
                          1943 4759+SALPH6 EQU   *                         MODULE ENTRY POINT
    1943 34 01 19F5            4760+       ST    SALND0+@OP1,@BR           SAVE ABA
    1947 C2 01 195F            4761+       LA    SALBSE,@BR                LOAD BASE RESISTER
    194B 74 08 9A              4762+       ST    SALND2+@OP1(,@BR),@ARR    SAVE RETURN ADDRESS
                               4763+*** END OF EXPANSION ***
    194E 74 02 34              4764+       ST    SAL375+@OP1(,@BR),@XR     SAVE ERROR POINTER

                               4766+***********************************************************************
                               4767+*
                               4768+*                  INITIALIZE WORK AREAS AND VARIABLE INSTRUCTIONS
                               4769+*
                               4770+***********************************************************************
    1951 7C 40 A8              4771+SAL100 MVI   SALPR7(,@BR),@BLANK       BLANK OUT SALPAR FOR PROCESSING
    1954 5C 08 A7 A8           4772+       MVC   SALPR6(##LPEN+@B1,@BR),SALPR7(,@BR)
    1958 7C 00 9C              4773+       MVI   SALCNT(,@BR),@ZERO        ZERO OUT COUNTER
    195B 5C 01 63 AA           4774+       MVC   SAL525+@OP1(2,@BR),SALPHS(,@BR) MODIFY MOVE OF CHARACTER

                               4776+***********************************************************************
                               4777+*
                               4778+*                  CHECK EBCDIC CHARACTERS
                               4779+*
                               4780+***********************************************************************
                               4781+*
                          195F 4782+SALBSE EQU   *                         MODULE BASE ADDR
    195F BD 5B 00              4783+SAL200 CLI   @ZERO(,@XR),@DOLAR        IS IT A '$' ?
    1962 F2 81 32              4784+       JE    SAL400                    YES, PROCESS CHARACTER
    1965 BD 7B 00              4785+       CLI   @ZERO(,@XR),@NUMBR        IS IT A '#' ?
    1968 F2 81 2C              4786+       JE    SAL400                    YES, PROCESS CHARACTER
    196B BD 7C 00              4787+       CLI   @ZERO(,@XR),@ASIGN        IS IT A '@' ?
    196E F2 81 26              4788+       JE    SAL400                    YES, PROCESS CHARACTER
                               4789+*
    1971 BD C1 00              4790+       CLI   @ZERO(,@XR),@CHARA        IS IT AN ALPHA (A-Z) ?
    1974 F2 82 53              4791+SAL250 JL    SAL750                    NO, CHECK FOR DELIMITERS
    1977 BD E9 00              4792+       CLI   @ZERO(,@XR),@CHARZ        IS IT AN ALPHA (A-Z) ?
    197A F2 04 1A              4793+       JNH   SAL400                    YES, PROCESS CHARACTER
    197D 78 80 9B              4794+       TBN   SALIDR(,@BR),SAL008       ENTERED AT SALPH8 ?
    1980 F2 90 17              4795+       JF    SAL425                    NO, CHECK IF NUMERIC
                               4796+*
    1983 78 01 9B              4797+       TBN   SALIDR(,@BR),SALFST       WAS FIRST CHAR FOUND ALPHA ?
    1986 3C 00 03CD            4798+       MVI   $CAERR,@@E100             ALPHA CHAR REQUIRED--ERROR
    198A F2 10 0D              4799+       JT    SAL425                    YES, CONTINUE
    198D 75 04 16              4800+SAL350 L     SALERR(,@BR),@PSR         LOAD ERROR CODE - LOW
    1990 C2 02 0000            4801+SAL375 LA    *-*,@XR                   RESTORE ERROR POINTER
    1994 F2 87 58              4802+       J     SAL800                    TAKE ERROR FAIT

                               4804+***********************************************************************
                               4805+*
                               4806+*                  PROCESS ALPHAMERIC CHARACTER
                               4807+*
                               4808+***********************************************************************
    1997 7A 01 9B              4809+SAL400 SBN   SALIDR(,@BR),SALFST       SET ON ALPHA :NOR
                               4810+*
    199A 5E 00 9C 9E           4811+SAL425 ALC   SALCNT(1,@BR),SAL001(,@BR) ADD 1 TO CHARACTER COUNTER
    199E 78 80 9B              4812+       TBN   SALIDR(,@BR),SAL008       WAS ENTRY AT SALPH8 ?
    19A1 D0 90 52              4813+       BF    SAL450(,@BR)              NO, CHECK COUNT FOR VALUE OF SIX
    19A4 7D 08 9C              4814+       CLI   SALCNT(,@BR),##LPEN       HAS COUNT EXCEEDED 8 ?
    19A7 3C 02 03CD            4815+       MVI   $CAERR,@@E102             PASSWORD/FILENAME LENGTH ERROR
    19AB D0 84 2E              4816+       BH    SAL350(,@BR)              YES, TAKE ERROR EXIT
    19AE F2 87 0A              4817+       J     SAL500                    NO, CONTINUE PROCESSING
    19B1 7D 06 9C              4818+SAL450 CLI   SALCNT(,@BR),@VOLID       HAS COUNT EXCEEDED 6 ?
    19B4 3C 03 03CD            4819+       MVI   $CAERR,@@E103             INVALID VOL-ID LENGTH
    19B8 D0 84 2E              4820+       BH    SAL350(,@BR)              YES, TAKE ERROR EXIT

                               4822+*
                               4823+*                  MODIFY MOVE OF CHARACTER
                               4824+*
    19BB 5E 01 63 9E           4825+SAL500 ALC   SAL525+@OP1(2,@BR),SAL001(,@BR)
    19BF 2C 00 0000 00         4826+SAL525 MVC   *-*,@ZERO(1,@XR)          MOVE CHARACTER TO OUTPUT AREA
    19C4 E2 02 01              4827+       LA    @B1(,@XR),@XR             INCREMENT XR BY I
    19C7 D0 87 00              4828+       B     SAL200(,@BR)              CHECK NEXT CHARACTER

                               4830+***********************************************************************
                               4831+*
                               4832+*                  CHECK ERRORS AND BYPASS DELIMITERS
                               4833+*
                               4834+***********************************************************************
    19CA 7D 00 9C              4835+SAL750 CLI   SALCNT(,@BR),@ZERO        ANY VALID CHARACTERS ?
    19CD 3C 10 03CD            4836+SAL755 MVI   $CAERR,@@E130             REQUIRED PARAM MISSING
    19D1 F2 01 17              4837+       JNE   SAL775                    YES, BYPASS DELIMITERS, EYIT
    19D4 BD 1E 00              4838+       CLI   @ZERO(,@XR),@EOS          IS IT EOS ?
    19D7 F2 81 0E              4839+       JE    SAL760                    YES, ERROR EVIL
    19DA 78 80 9B              4840+       TBN   SALIDR(,@BR),SAL008       ENTERED AT SALPH8 ?
    19DD 3C 00 03CD            4841+       MVI   $CAERR,@@E100             ALPHABETIC CHAR REQUIRED
    19E1 F2 10 04              4842+       JT    SAL760                    ERROR EYIT
    19E4 3C 01 03CD            4843+       MVI   $CAERR,@@E101             ALPHAMERIC CHAR REQUIRED
    19E8 D0 87 2E              4844+SAL760 B     SAL350(,@BR)              ERROR EYIT
    19EB C0 87 1BE8            4845+SAL775 B     SCANIT                    BYPASS DELIMITERS

                               4847+***********************************************************************
                               4848+*
                               4849+*                  SET OFF INDICATORS FOR POSSIBLE SALDHA RE-ENTRY
                               4850+*
                               4851+***********************************************************************
    19EF 7C 00 9B              4852+SAL800 MVI   SALIDR(,@BR),@ZERO

                               4854+***********************************************************************
                               4855+*
                               4856+*                  END OF MODULE PROCESSING
                               4857+*
                               4858+***********************************************************************
                               4859+*SALND  EXIT  @BR,,RETURN              EXIT
    19F2 C2 01 0000            4860+SALND0 LA    *-*,@BR                   RESTORE @BR
    19F6 C0 87 0000            4861+SALND2 B     *-*                       RETURN TO CALLING PROGRAM
                               4862+*** END OF EXPANSION ***

                               4864+***********************************************************************
                               4865+*
                               4866+*                  DATA CONSTANTS, BUFFERS, AND WORK AREAS
                               4867+*
                               4868+***********************************************************************
    19FA                  19FA 4869+SALIDR DS    CL1                       1 BYTE OF FLAGS
    19FA                       4870+       ORG   *-1
```

```
ERR LOC  OBJECT CODE      ADDR STMT SOURCE STATEMENT
    19FA 00               19FA 4871+       DC    XL1'00'                   INITIALIZED TO ZERO

                          0080 4873+SAL008 EQU   X'80'                     ENTRY POINT INDICATOR
                               4874+*                                      * 0 - ENTERED AT SALPH6
                               4875+*                                      * 1 - ENTERED AT SALPH8
                          0001 4876+SALFST EQU   X'01'                     FIRST CHARACTER IS ALPHA / INDR
                               4877+*                                      * 0 - CHARACTER IS NOT ALPHA
                               4878+*                                      * 1 - CHARACTER IS ALPHA
    19FB                  19FB 4879+SALCNT DS    CL1                       BYTE CHARACTER COUNTER
    19FB                       4880+       ORG   *-1
    19FB 00               19FB 4881+       DC    XL1'00'                   INITIALIZED TO ZERO
    19FC 0001             19FD 4882+SAL001 DC    XL2'0001'                 COUNTER INCREMENT
                          19FE 4883+SALPHR EQU   *
    19FE                  1A07 4884+       DS    CL(##LUEN+2*@B1)          SYNTAX SAVE UNIT
    1A08 19FD             1A09 4885+SALPHS DC    AL2(SALPHR-1)             ADDR FOR MODIFYING MOVE
                          1A07 4886+SALPR7 EQU   SALPHR+##DPEN+2*@B1       ADDR IN SALPHR FOR CLANKINS
                          1A06 4887+SALPR6 EQU   SALPHR+##DPEN+@B1         * OUT THE FIELD
                          1975 4888+SALERR EQU   SAL250+@Q                 ADDR ERROR CODE FOR LOAD
                               4889+***                       END OF SALPHA                            ***
                               4890 *      $GETD
```

```
                                4892+***********************************************************************
                                4893+*  5703-XM1 COPYRIGHT IBM CORP. 1970                                  *
                                4894+*           REFER TO INSTRUCTIONS ON COPY RIGHT NOTICE, 120-2083      *
                                4895+*                                                                     *
                                4896+***********************************************************************
                                4897+*STATUS                                                               *
                                4898+*   VERSION 1 MODIFICATION 0                                          *
                                4899+*                                                                     *
                                4900+*FUNCTION                                                             *
                                4901+*   * SGETDB PROVIDES TWO PRIMARY FUNCTIONS. IT WILL SEARCH THE       *
                                4902+*     PASSWORD DIRECTORY FOR A SPECIFIED PASSWORD ONLY, OR IF         *
                                4903+*     INDICATED WILL GO AND READ IN THE FIRST USER BLOCK ASSOCIATED   *
                                4904+*     WITH THAT PASSWORD.                                             *
                                4905+*   * IF THE PASSWORD SEARCH ONLY IS REQUESTED A SWITCH IS SET TO     *
                                4906+*     INHIBIT READING THE DIRECTORY ON SUBSEQUENT ENTRIES.            *
                                4907+*   * THE ERROR CODE FOR PASSWORD NOT FOUND IS ALWAYS SET IN $CAERR.  *
                                4908+*     IF THE PASSWORD IS OR IS NOT FOUND THE INDICATOR IN SMIND1 IS   *
                                4909+*     SET APPROPRIATELY.                                              *
                                4910+*                                                                     *
                                4911+*ENTRY POINTS                                                         *
                                4912+*   SGETDB - ENTRY TO SEARCH PASSWORD DIRECTORY AND GET               *
                                4913+*            ASSOCIATED USER DIRECTORY. THE CALLING SEQUENCE IS       *
                                4914+*            AS FOLLOWS:                                              *
                                4915+*              B     SGETDB                                           *
                                4916+*                                                                     *
                                4917+*INPUT                                                                *
                                4918+*   * THE BASE ADDRESS OF THE LIBRARY MUST BE IN SM1FDA IN TSMLES.    *
                                4919+*   * THE PASSWORD MUST BE IN SMPSWD.                                 *
                                4920+*   * IF THE PASSWORD DIRECTORY IS TO BE SEARCHED ONLY, THEN SM1PDS   *
                                4921+*     IN SMIND1 MUST BE SET TO 1. IF THE FIRST USER DIRECTORY BLOCK   *
                                4922+*     ASSOCIATED WITH THE SPECIFIED PASSWORD IS TO BE READ IN THEN    *
                                4923+*     THEN SM1PDS MUST BE SET TO 0.                                   *
                                4924+*                                                                     *
                                4925+*OUTPUT                                                               *
                                4926+*   * IF THE SPECIFIED PASSWORD IS FOUND THE ADDRESS OF THE LEFT BYTE *
                                4927+*     OF THE ENTRY IS PLACED IN SMPEAD, SM1PNF IN SMIND1 IS SET TO 0. *
                                4928+*     AND THE USER DIRECTORY RDADDR IS PLACED IN SMFUDA.              *
                                4929+*   * IF THE USER DIRECTORY WAS REQUESTED, THE READ OPERATION IS      *
                                4930+*     STARTED BUT NO WAIT IS PERFORMED. THE USER DIRECTORIES OVERLAY  *
                                4931+*     THE PASSWORD DIRECTORIES IN CORE.                               *
                                4932+*   * IF THE SPECIFIED PASSWORD WAS NOT FOUND SM1PNF, IS SET TO 1 AND *
                                4933+*     THE ADDRESS FOR THE NEXT AVAILABLE ENTRY IS IN SMPEAD.          *
                                4934+*                                                                     *
                                4935+*EXTERNAL REFERENCES                                                  *
                                4936+*         $CAERR - LOCATION FOR SYSTEM ERROR CODE                     *
                                4937+*         SMIND1 - DATA MANAGEMENT INDICATOR                          *
                                4938+*         DL2RAD - LOCATION OF FILE PHYSICAL BASE ADDRESS             *
                                4939+*         SMBFDA - LOCATION OF LIBRARY BASE ADDRESS                   *
                                4940+*         DL2ICS - ENTRY TO DISK I/O ROUTINE                          *
                                4941+*         $DISKN - ENTRY TO SYSTEM DISK IOCS                          *
                                4942+*         $WAITF - LOCATION OF COMMON I/O WAIT FUNCTION               *
                                4943+*         SMPSWD - LOCATION PASSWORD ARGUMENT                         *
                                4944+*         SMPEAD - LOCATION OF PASSWORD ENTRY ADDRESS                 *
                                4945+*         SMFUDA - LOCATION OF USER DIRECTORY RDADDR                  *
                                4946+*                                                                     *
                                4947+*EXITS, NORMAL                                                        *
```

```
                               4948+*   NORMAL EXIT IS TO THE FIRST INSTRUCTION FOLLOWING THE BRANCH      *
                               4949+*   TO SGETDB                                                         *
                               4950+*                                                                     *
                               4951+*EXITS, ERROR                                                         *
                               4952+*   NONE                                                              *
                               4953+*                                                                     *
                               4954+*TABLES/WORKAREAS                                                     *
                               4955+*   NONE                                                              *
                               4956+*                                                                     *
                               4957+*ATTRIBUTES                                                           *
                               4958+*   RELOCATABLE                                                       *
                               4959+*   REUSABLE                                                          *
                               4960+*                                                                     *
                               4961+*CHARACTER CODE DEPENDENCY                                            *
                               4962+*   THE OPERATION OF THIS MODULE DOES NOT DEPEND UPON A PARTICULAR    *
                               4963+*   INTERNAL REPRESENTATION OF THE EXTERNAL CHARACTER SET.            *
                               4964+*                                                                     *
                               4965+*NOTES                                                                *
                               4966+*   ERROR PROCEDURES                                                  *
                               4967+*      THE ERROR CODE FOR PASSWORD NOT FOUND IS ALWAYS SET BUT SGETDB *
                               4968+*      DETECTS NO PARTICULAR ERROR. THE CONDITION AS TO IF THE        *
                               4969+*      PASSWORD WAS OR WAS NOT FOUND IS INDICATED HOWEVER.            *
                               4970+*                                                                     *
                               4971+*   REGISTER USAGE                                                    *
                               4972+*      @BR AND @XR1 ARS SAVED AND RESTORED. @BR IS USED AS A BASE     *
                               4973+*      REGISTER AND @XR IS USED AS AN INDEX TO THE PASSWORD DIRCTY.   *
                               4974+*      @ARR IS USED TO PROVIDE THE RETURN ADDRESS.                    *
                               4975+*                                                                     *
                               4976+*   SAVED/RESTORED AREAS                                              *
                               4977+*      NONE                                                           *
                               4978+*                                                                     *
                               4979+*   MODIFICATION CONSIDERATIONS                                       *
                               4980+*      IN USING SGETDB THE USER MUST TAKE INTO CONSIDERATION THAT     *
                               4981+*      SGETDB DOES NOT WAIT FOR THE USER DIRECTORY BLOCK TO BE IN     *
                               4982+*      CORE BEFORE RETURNING.                                         *
                               4983+*                                                                     *
                               4984+*   REQUIRED MODULES                                                  *
                               4985+*      @SYSEQ - SYSTEM SOFTWARE EQUATES                               *
                               4986+*      @FXDEQ - NUCLEUS EQUATES                                       *
                               4987+*      @DIREQ - LIBRARY DIRECTORY EQUATES                             *
                               4988+*      DL2ICS - DISK IOCS                                             *
                               4989+*      TSMLES - DATA MANAGEMENT COMMUNICATIONS AREA                   *
                               4990+*                                                                     *
                               4991+*   OTHER                                                             *
                               4992+*      NONE                                                           *
                               4993+***********************************************************************
                               4994+*SGETDB ENTER BASE,SGETDB,EXIT,SGE90,@BR,@XR,@ARR
                          1A0A 4995+       USING SGETDB,@BR                BASE ADDRESS SPECIFICATION
                          1A0A 4996+SGETDB EQU   *                         MODULE ENTRY POINT
    1A0A 34 01 1A82            4997+       ST    SGE900+@OP1,@BR           SAVE @BR
    1A0E C2 01 1A0A            4998+       LA    SGETDB,@BR                LOAD BASE REGISTER
    1A12 74 02 7C              4999+       ST    SGE901+@OP1(,@BR),@XR     SAVE @XR
    1A15 74 08 80              5000+       ST    SGE902+@OP1(,@BR),@ARR    SAVE RETURN ADDRESS
                               5001+*** END OF EXPANSION ***

    1A18 3C 23 03CD            5003+       MVI   $CAERR,@@E210             PASSWORD NOT ON DISK
    1A1C 3B 08 1B3E            5004+        SBF   SMIND1,SM1PNF             INITIALIZE INDICATOR TO FOUND
    1A20 F2 80 15              5005+SGE050  JC    SGE055,@NOP               SET SWITCH FOR 2ND ENTRY
    1A23 7C 87 17              5006+        MVI   SGE050+@Q(,@BR),@UCB      TURN SWITCH ON FOR NEXT ENTRY
    1A26 0C 01 1300 1B58       5007+        MVC   DL2RAD,SMBFDA             STUFF IN THE BASE ADDR
    1A2C C0 87 1268            5008+        B     DL2ICS                    CALL DISK I/O ROUTINE
    1A30 1A8B             1A31 5009+        DC    AL2(SGEDPL)               POINTER TO PARAMETER LIST
    1A32 C0 87 0025            5010+        B     $DISKN                    WAIT FOR DIRCTY TO LOAD
    1A36 057F             1A37 5011+        DC    AL2($WAITF)               WAIT FOR DIRCTY

    1A38 75 02 86              5013+SGE055  L     SGEDPL+@DBFR2(,@BR),@XR   PASSWORD BUFFER CADDR
    1A3B 6C 00 89 00           5014+        MVC   SGECNT(1,@BR),##DPHC(,@XR) ENTRY COUNT TO WORK
    1A3F E2 02 04              5015+        LA    ##DPE1(,@XR),@XR          BUMP TO FIRST PASSWORD
                               5016+*
    1A42 2D 07 1B4C 07         5017+SGE060  CLC   SMPSWD(##LPEN),##DPEN(,@XR) LOOK AT PSWD ENTRY
    1A47 F2 81 0E              5018+        JE    SGE070                    FOUND THE PSWD
    1A4A E2 02 0C              5019+        LA    ##LPE(,@XR),@XR           BUMP TO LOOK AT NEXT ENTRY
    1A4D 5F 00 89 8B           5020+        SLC   SGECNT(1,@BR),SGEC01(,@BR) DECR ENTRY COUNT
    1A51 D0 01 38              5021+        BNE   SGE060(,@BR)              BACK FOR LOOK AT ENTRY
    1A54 3A 08 1B3E            5022+        SBN   SMIND1,SM1PNF             NOT FOUND INDICATOR
                               5023+*
                               5024+*             THE PASSWORD OR THE END OF THE DIRCTY HAS BEEN FOUND,
                               5025+*             SAVE THE POINTERS.
                               5026+*
    1A58 34 02 1B66            5027+SGE070  ST    SMPEAD,@XR                SAVE ENTRY ADDRESS
    1A5C 2C 01 1B68 09         5028+        MVC   SMFUDA(@DADDR),##DPEA(,@XR) POSSIBLE USER DADDR OF BLK
    1A61 38 10 1B3E            5029+        TBN   SMIND1,SM1PDS             TEST SEARCH BIT ONLY ON
    1A65 F2 10 17              5030+        JT    SGE900                    SEARCH ONLY SO EXIT
    1A68 7D 00 89              5031+        CLI   SGECNT(,@BR),@ZERO        TEST COUNT IF ENTRY FOUND
    1A6B F2 81 11              5032+        JE    SGE900                    JUMP IF NOT FOUND
    1A6E 6C 01 83 09           5033+SGE080  MVC   SGEDPL+@DSAD(@DADDR,@BR),##DPEA(,@XR) BLK ADDR TO DPL
    1A72 C0 87 1268            5034+        B     DL2ICS                    CALL TO READ USER DIRCTY
    1A76 1A8B             1A77 5035+        DC    AL2(SGEDPL)               POINTER TO PARAMETER LIST
                               5036+*
    1A78 7C 80 17              5037+        MVI   SGE050+@Q(,@BR),@NOP      TURN OFF SKIP INSTR
    1A7B 5C 01 83 88           5038+        MVC   SGEDPL+@DSAD(@DADDR,@BR),SGERAD(,@BR) RESTORE DSAD PSWD
                               5039+*
                               5040+*SGE900 EXIT @BR,@XR,,RETURN
    1A7F C2 01 0000            5041+SGE900 LA    *-*,@BR                   RESTORE OBR
    1A83 C2 02 0000            5042+SGE901 LA    *-*,@XR                   RESTORE OXR
    1A87 C0 87 0000            5043+SGE902 B     *-*                       RETURN TO CALLING PROGRAM
                               5044+*** END OF EXPANSION ***
                               5045+*
                               5046+*             DPL TO READ IN THE PASSWORD DIRCTY
                               5047+*
                               5048+*SGEDPL $DPL  FUNC-@DGET,DADDR-##RP,CNT-##LP,CADDR-SMPDB1
                          1A8B 5049+SGEDPL EQU   *                         DISK PARAMETER
    1A8B 01               1A8B 5050+        DC    AL1(@DGET)                REQUESTED FUNCTION
    1A8C 0001             1A8D 5051+        DC    AL2(##RP)                 DISK ADDRESS
    1A8E 04               1A8E 5052+        DC    AL1(##LP)                 SECTOR COUNT
    1A8F 1B6D             1A90 5053+        DC    AL2(SMPDB1)               BUFFER ADDRESS
                               5054+*** END OF EXPANSION ***

    1A91 0001             1A92 5056+SGERAD DC    AL2(##RP)                 RELATIVE DADDR OF DIRCTY
    1A93                  1A93 5057+SGECNT DS    CL1                       SAVE AREA FOR ENTRY COUNT
    1A94 0001             1A95 5058+SGEC01 DC    IL2'1'                    CONSTANT 1 FOR ADDR MODIFCATION
```

```
                          1A96 5060+SGEEND EQU   *                         END ADDR OF SGETDB
                               5061+***                       END OF SGETDB                             ***
    1B3E                       5062        ORG   X'1B3E'
```

```
                               5064 ************************************************************************
                               5065 * SMALES- SYSTEM DATA MANAGEMENT COMMON SAVE AREAS AND EQUATES        *
                               5066 *         USED TO PROVIDE COMMUNICATION BETWEEN SUBROUTINES USED      *
                               5067 *         BY THE VARIOUS KEYWORDS INVOLVED WITH FILE MANIPULATION     *
                               5068 ************************************************************************
                               5069 *
                          1B3E 5070 SMALES EQU   *                         START OF MANAGEMENT AREA
                          1B3E 5071 SMIND1 EQU   SMALES                    INDICATOR BYTE 1
                          1B44 5072 SMVOID EQU   SMIND1+6                  SPECIFIED VOLUME ID SAVE AREA
                          1B4C 5073 SMPSWD EQU   SMVOID+8                  SPECIFIED PASSWORD SAVE AREA
                          1B54 5074 SMFNAM EQU   SMPSWD+8                  SPECIFIED FILENAME SAVE AREA
                          1B56 5075 SMUDEA EQU   SMFNAM+2                  FILENAME DIRCTY ENTRY ADDR
                          1B58 5076 SMBFDA EQU   SMUDEA+2                  DADDR OF FILE LIBRARY
                          1B5A 5077 SMUDBA EQU   SMBFDA+2                  CADDR OF ACTIVE BUFFER ADDR
                          1B5C 5078 SMNULT EQU   SMUDBA+2                  TOTAL OF NULL SECTORS AVAILABLE
                          1B5E 5079 SMNDEA EQU   SMNULT+2                  NULL DIRCTY ENTRY ERROR
                          1B60 5080 SMNSCT EQU   SMNDEA+2                  COUNT OF NULL SECTORS REQUIRED
                          1B62 5081 SMNETD EQU   SMNSCT+2                  CADDR NEW ENTRY TO NULL DIRCTY
                          1B64 5082 SMUPEN EQU   SMNETD+2                  CADDR NEW USER DIRCTY ENTRY
                          1B66 5083 SMPEAD EQU   SMUPEN+2                  CADDR PASSWORD ENTRY
                          1B68 5084 SMFUDA EQU   SMPEAD+2                  REL DADDR 1ST USER DIRCTY BLOCK
                          1B6A 5085 SMNDBA EQU   SMFUDA+2                  NULL DIRCTY BUFFER CORE ADDR
                          1B6C 5086 SMDAAD EQU   SMNDBA+2                  DADDR OF ACTIVE DIRCTY
                          0080 5087 SM1FNE EQU   X'80'                     SRCHFN INDR NAME NOT FOUND
                          0040 5088 SM1NPD EQU   X'40'                     PACK INDR NULL DIRCTY FULL
                          0020 5089 SM1STN EQU   X'20'                     STORIN PACK INDICATOR BIT
                          0010 5090 SM1PDS EQU   X'10'                     SGETDB SEARCH ONLY FLAG
                          0008 5091 SM1PNF EQU   X'08'                     SGETDB PASSWORD NOT FOUND
                          1B6D 5092 SMPDB1 EQU   SMDAAD+1                  PASSWORD DIRCTY BUFFER
                          1B6D 5093 SMPIBS EQU   SMPDB1                    SVOLID TEMP SAVE INPUT BUFFER
                          1B6D 5094 SMUDB1 EQU   SMPDB1                    USER DIRCTY BLOCK 1 BUFFER
                          1D6D 5095 SMUDB2 EQU   SMUDB1+512                USER DIRCTY BLOCK 2 BUFFER
                          1F6D 5096 SMAEND EQU   SMUDB2+512                END OF SMALES AREA
    1B3E                       5097        ORG   SMIND1                    BUFFER ADDR FOR SVOLID
    1B3E 00               1B3E 5098        DC    IL1'0'                    SET SMIND1 TO ZERO
    1B3F 4040404040404040 1B54 5099        DC    22CL1' '                  INITIALIZE TSMLES FIELDS
    1B6D                       5100        ORG   SMPDB1
                          1B6D 5101 SVOBUF EQU   SMPDB1                    TEMPORALY SVOID BUFFER
                               5102 *SCSTRG EQU    *                        ENTRY POINT
                               5103 *
                               5104 *      $CSTR
                               5106+***********************************************************************
                               5107+* 5703-XM1      COPYRIGHT IBM CORP. 1970                              *
                               5108+*               REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083   *
                               5109+*                                                                     *
                               5110+***********************************************************************
                               5111+*STATUS                                                               *
                               5112+*   VERSION 1 MODIFICATION 0                                          *
                               5113+*                                                                     *
                               5114+*FUNCTION                                                             *
                               5115+*   * SCSTRG PLACES THE SYNTACTIC UNIT <CHARACTER STRING> IN          *
                               5116+*     AN AREA DEFINED BY THE USER.  THIS ROUTINE WILL ALSO PLACE A    *
                               5117+*     NUMBER OF CHARACTERS IN THE CALLING PROGRAMS AREA.              *
                               5118+*   * A COUNT OF THE NUMBER OF CHARACTERS IN THE STRING IS MAINTAINED *
                               5119+*     BY SCSTRG.                                                      *
                               5120+*                                                                     *
                               5121+*ENTRY POINTS                                                         *
                               5122+*   THE ONLY ENTRY TO SCSTRG IS THE FIRST BYTE OF                     *
                               5123+*   THE ROUTINE. THE CALLING SEQUENCE IS:                             *
                               5124+*      B     SCSTRG                                                   *
                               5125+*      DC    AL2(AREA)                                                *
                               5126+*                                                                     *
                               5127+*   WHERE AREA POINTS TO THE LEFTMOST BYTE OF THE CALLING             *
                               5128+*   PROGRAMS OUTPUT AREA.                                             *
                               5129+*                                                                     *
                               5130+*INPUT                                                                *
                               5131+*   INDEX REGISTER TWO(2) SHOULD POINT TO THE LEFT QUOTE OF THE       *
                               5132+*   CHARACTER STRING.  THE CALLING PROGRAM MUST ALSO SET THE          *
                               5133+*   CHARACTER COUNT IN THE ONE BYTE FIELD SCSLNG.  A ZERO(0) LENGTH   *
                               5134+*   DENOTES THAT THE CALLING PROGRAM WANTS THE ENTIRE STRING.         *
                               5135+*                                                                     *
                               5136+*OUTPUT                                                               *
                               5137+*   THE CHARACTER STRING IS RETURNED TO THE ADDRESS GIVEN BY THE      *
                               5138+*   CALLING ROUTINE.  THE FIELD SCSCNT CONTAINS THE NUMBER OF         *
                               5139+*   CHARACTERS IN THE CHARACTER STRING.                               *
                               5140+*                                                                     *
                               5141+*EXTERNAL REFERENCES                                                  *
                               5142+*   NONE                                                              *
                               5143+*                                                                     *
                               5144+*EXITS, NORMAL                                                        *
                               5145+*   NORMAL EXIT IS TO THE FIRST BYTE FOLLOWING THE THE                *
                               5146+*   POINTER TO THE USERS STRING AREA.  THE BASE REGISTER              *
                               5147+*   IS RESTORED(XR1).  XR2 WILL POINT TO THE CHARACTER                *
                               5148+*   FOLLOWING THE ENDING QUOTE.  THE PSR WILL BE NOT LOW.             *
                               5149+*                                                                     *
                               5150+*EXITS,ERROR                                                          *
                               5151+*   SHOULD AN ERROR BE FOUND THE PSR IS FORCED LOW.  THE XR2          *
                               5152+*   WILL POINT TO THE POSITION WHERE THE ERROR WAS FOUND.             *
                               5153+*                                                                     *
                               5154+*TABLES/WORKAREAS                                                     *
                               5155+*   NONE                                                              *
                               5156+*                                                                     *
                               5157+*ATTRIBUTES                                                           *
                               5158+*   SCSTRG IS REUSABLE                                                *
                               5159+*                                                                     *
                               5160+*CHARACTER CODE DEPENDENCY                                            *
                               5161+*   THIS ROUTINE ASSUMES THE EBCDIC CODE OF X'7D' FOR A               *
                               5162+*   SINGLE QUOTE.                                                     *
                               5163+*                                                                     *
                               5164+*NOTES                                                                *
                               5165+*   ERROR PROCEDURES                                                  *
                               5166+*      N/A                                                            *
                               5167+*                                                                     *
                               5168+*   REGISTER USAGE                                                    *
                               5169+*      INDEX REGISTER 1 IS USED AS A POINTER TO THE CALLING PROGRAMS  *
                               5170+*      STRING AREA. INDEX REGISTER 2 POINTS TO THE CHARACTER STRING   *
                               5171+*      IN THE INPUT LINE. XR 1 IS SAVED AND RESTORED.                 *
                               5172+*                                                                     *
                               5173+*   REQUIRED MODULES                                                  *
                               5174+*      @SYSEQ - SYSTEM EQUATES                                        *
                               5175+*                                                                     *
                               5176+*   MODIFICATION CONSIDERATIONS                                       *
                               5177+*      NONE                                                           *
                               5178+*                                                                     *
                               5179+*   OTHER                                                             *
                               5180+*      NONE                                                           *
                               5181+***********************************************************************

                          1B6D 5183+SCSTRG EQU   *                         ENTRY POINT
    1B6D 34 01 1BDD            5184+       ST     SCS050+@OP1,@BR           SAVE BASE REGISTER
    1B71 34 08 1BE1            5185+       ST     SCS051+@OP1,@ARR          SAVE RETURN ADDRESS
    1B75 0E 00 1BE1 1BE5       5186+       ALC    SCS051+@OP1(@B1),SCSPL2   INCREMENT PAST PARAMETER
    1B7B 36 08 1BE4            5187+       A      SCSPL1,@ARR               POINT TO PARAMETER
    1B7F 34 08 1B8E            5188+       ST     SCS005+@OP1,@ARR          SAVE PARAMETER ADDRESS
    1B83 3C 00 1BE2            5189+       MVI    SCSCNT,@ZERO              CLEAR COUNTER
    1B87 3C 80 1BB4            5190+       MVI    SCS020+@Q,@NOP            SET SWITCH OFF
    1B8B 35 01 0000            5191+SCS005 L      *-*,@BR                   PICK UP OUTPUT ADDRESS
    1B8F BD 7D 00              5192+       CLI    @ZERO(,@XR),SCSQUO        CHECK QUOTES
    1B92 F2 01 37              5193+       JNE    SCS030                    ERROR -
                               5194+*
    1B95 E2 02 01              5195+SCS006 LA     @B1(,@XR),@XR             INCREMENT POINTER
    1B98 BD 7D 00              5196+       CLI    @ZERO(,@XR),SCSQUO        EMBEDDED QUOTES
    1B9B F2 01 09              5197+       JNE    SCS010                    NO GO CHECK FOR EOS
    1B9E E2 02 01              5198+       LA     @B1(,@XR),@XR             MOVE INPUT POINTER
    1BA1 BD 7D 00              5199+       CLI    @ZERO(,@XR),SCSQUO        DOUBLE QUOTE ?
    1BA4 F2 01 30              5200+       JNE    SCS040                    EXIT
    1BA7 BD 1E 00              5201+SCS010 CLI    @ZERO(,@XR),@EOS          END OF STATEMENT ?
    1BAA F2 81 1F              5202+       JE     SCS030                    YES - ERROR
    1BAD 0E 00 1BE2 1BE4       5203+       ALC    SCSCNT(@B1),SCSPL1        INCREMENT COUNT
                               5204+*
    1BB3 F2 00 12              5205+SCS020 JC     SCS029,*-*                SWITCH
    1BB6 6C 00 00 00           5206+       MVC    @ZERO(@B1,@BR),@ZERO(,@XR) MOVE CHARACTER
    1BBA D2 01 01              5207+       LA     @B1(,@BR),@BR             BUMP OUTPUT POINTER
                               5208+*
    1BBD 3D 00 1BE2            5209+SCS025 CLI    SCSCNT,*-*                CHECK CHARACTER COUNT
    1BC1 F2 01 04              5210+       JNE    SCS029                    NOT EXCEEDED CONTINUE
    1BC4 3C 87 1BB4            5211+       MVI    SCS020+@Q,@UCB            SET SWITCH ON
    1BC8 C0 87 1B95            5212+SCS029 B      SCS006                    RETURN TO MAINLINE
```

```
                               5214+*
                               5215+*                   ERROR SETTING
                               5216+*
                          1BCC 5217+SCS030 EQU   *
    1BCC 35 04 1BE7            5218+       L     SCSERR,@PSR               SET ERROR INDICATOR
    1BD0 3C 17 03CD            5219+       MVI   $CAERR,@@E138             INCOMPLETE CHARACTER CONSTANT
    1BD4 F2 87 03              5220+       J     SCS050                    RETURN
    1BD7 BD FF 00              5221+SCS040 CLI   0(,@XR),SCSFRC            FORCE PSR LOW
                               5222+*
                               5223+*                   RETURN
                               5224+*
    1BDA C2 01 0000            5225+SCS050 LA    *-*,@BR                   RESTORE BASE
    1BDE C0 87 0000            5226+SCS051 B     *-*                       RETURN
                               5227+*
                               5228+*                   CONSTANTS
                               5229+*
                          1BBE 5230+SCSLNG EQU   SCS025+@Q                 LENGTH REQUESTED
                          007D 5231+SCSQUO EQU   X'7D'                     QUOTE
                          00FF 5232+SCSFRC EQU   X'FF'                     FORCE PSR INDICATOR
                               5233+*
    1BE2                  1BE2 5234+SCSCNT DS    CL1                       CHARACTER COUNT
    1BE3 0001             1BE4 5235+SCSPL1 DC    IL2'1'                    PLUS ONE
    1BE5 02               1BE5 5236+SCSPL2 DC    IL1'2'                    PLUS TWO
    1BE6 0084             1BE7 5237+SCSERR DC    XL2'84'                   PSR CODE FOR ERROR
                               5238+***                       END OF SCSTRG                             ***
                               5239 *      $CANI
```

```
                              5241+*********************************************************************
                              5242+*  5703-XM1    COPYRIGHT IBM CORP. 1970                             *
                              5243+*              REFER TO INSTRUCTIONS ON COPYRIGHT NOTICE, 120-2083    *
                              5244+*                                                                     *
                              5245+*********************************************************************
                              5246+*STATUS                                                               *
                              5247+*   VERSION 1 MODIFICATION 0                                          *
                              5248+*                                                                     *
                              5249+*FUNCTION                                                             *
                              5250+*   THE FUNCTION OF SCANIT IS TO SCAN PAST VALID DELIMITERS AND       *
                              5251+*   RETURN A POINTER TO THE FIRST CHARACTER THAT'S NOT A DELIMITER.   *
                              5252+*                                                                     *
                              5253+*ENTRY POINTS                                                         *
                              5254+*   * THE ENTRY POINT IS SCANIT.                                      *
                              5255+*   * THE CALLING SEQUENCE IS AS FOLLOWS:                             *
                              5256+*            B     SCANIT                                             *
                              5257+*     WITH REGISTER 2 (@XR) POINTING TO THE FIRST CHARACTER TO BE     *
                              5258+*     EXAMINED.                                                       *
                              5259+*                                                                     *
                              5260+*INPUT                                                                *
                              5261+*   NONE                                                              *
                              5262+*                                                                     *
                              5263+*OUTPUT                                                               *
                              5264+*   NONE                                                              *
                              5265+*                                                                     *
                              5266+*EXTERNAL REFERENCES                                                  *
                              5267+*   $CAERR - ERROR CODE SAVE AREA                                     *
                              5268+*                                                                     *
                              5269+*EXITS, NORMAL                                                        *
                              5270+*   NORMAL EXIT FROM SCANIT IS TO THE BYTE FOLLOWING THE BRANCH TO    *
                              5271+*   SCANIT IN THE CALLING ROUTINE. THE PSR (REGISTER 4) WILL CONTAIN  *
                              5272+*   A ZERO IF NO DELIMITERS WERE FOUND OR A HIGH CONDITION IF ONE OR  *
                              5273+*   MORE DELIMITERS WERE SCANNED.                                     *
                              5274+*                                                                     *
                              5275+*EXITS, ERROR                                                         *
                              5276+*   ERROR EXIT FROM SCANIT IS TO THE BYTE FOLLOWING THE BRANCH TO     *
                              5277+*   SCANIT IN THE CALLING ROUTINE. THE PSR WILL CONTAIN A LOW         *
                              5278+*   CONDITION.                                                        *
                              5279+*                                                                     *
                              5280+*TABLES/WORKAREAS                                                     *
                              5281+*   * SCACNT - AREA CONTAINING NUMBERS OF DELIMITERS SCANNED          *
                              5282+*   * SCAMMA - LOC WHERE SCACOM MAY BE MOVED IF ONE COMMA IS ALSO     *
                              5283+*     TO BE CONSIDERED A DELIMITER. MOVING SCACOF BACK INTO SCAMMA    *
                              5284+*     INDICATES THAT ONLY BLANKS SHOULD BE CONSIDERED DELIMITERS.     *
                              5285+*                                                                     *
                              5286+*ATTRIBUTES                                                           *
                              5287+*   RELOCATABLE AND RE-USABLE                                         *
                              5288+*                                                                     *
                              5289+*CHARACTER CODE DEPENDENCY                                            *
                              5290+*   THE OPERATION OF THIS MODULE DOES NOT DEPEND UPON A PARTICULAR    *
                              5291+*   INTERNAL REPRESENTATION OF THE EXTERNAL CHARACTER SET.            *
                              5292+*                                                                     *
                              5293+*NOTES                                                                *
                              5294+*   ERROR PROCEDURES                                                  *
                              5295+*      THE ONLY ERROR CONDITION DETECTED BY SCANIT IS THE CASE WHERE  *
                              5296+*      A CARRIAGE-RETURN CODE FOLLOWS A COMMA.  UPON RETURN TO THE    *
```

```
                               5297+*      CALLING ROUTINE, @PSR WILL BE SET TO A LOW CONDITION, THE      *
                               5298+*      ERROR CODE IS SET IN $CAERR, AND MG WILU BE POINTING TO THE    *
                               5299+*      CARRIAGE-RETURN CHARACTER.                                     *
                               5300+*                                                                     *
                               5301+*   REGISTER USAGE                                                    *
                               5302+*      REGISTER 2 (@XR) IS USED AS A POINTER ACROSS THE AREA BEING    *
                               5303+*      SCANNED FOR DELIMETERS.                                        *
                               5304+*                                                                     *
                               5305+*   SAVED/RESTORED AREAS                                              *
                               5306+*      UPON ENTRY TO SCANIT, REGISTER 8 (@ARR) IS SAVED AND USED AS   *
                               5307+*      THE RETURN ADDRESS.                                            *
                               5308+*                                                                     *
                               5309+*   MODIFICATION CONSIDERATIONS                                       *
                               5310+*      NONE                                                           *
                               5311+*                                                                     *
                               5312+*   REQUIRED MODULES                                                  *
                               5313+*      * @SYSEQ - COMMON SYSTEM EQUATES                               *
                               5314+*      * @FXDEQ - FIXED NUCLEUS ADDRESSES EQUATES                     *
                               5315+*                                                                     *
                               5316+*   OTHER                                                             *
                               5317+*      SCANIT IS INITIALIZED TO BYPASS BLANKS ONLY. IF SCACOM IS      *
                               5318+*      MOVED TO SCAMMA, ONE COMMA WILL BE SCANNED ALONG WITH BLANKS.  *
                               5319+*      THE INSTRUCTION TO DO THIS IS AS FOLLOWS:                      *
                               5320+*            MVI   SCAMMA,SCACOM                                      *
                               5321+*                                                                     *
                               5322+*      TO DROP THE COMMA FROM ITS DELIMITER STATUS, SCACOF SHOULD BE  *
                               5323+*      MOVED TO SCAMMA, USING THE FOLLOWING INSTRUCTION:              *
                               5324+*            MVI   SCAMMA,SCACOF                                      *
                               5325+*                                                                     *
                               5326+***********************************************************************


                               5328+*
                               5329+*                  EQUATES USED IN THIS SUBROUTINE
                               5330+*
                          0001 5331+SCAINC EQU   1                         TO INCREMENT POINTER
                          0001 5332+SCACOM EQU   @BNE                      SWITCH TO ALLOW SCANNING COMMA
                          0087 5333+SCACOF EQU   @UCB                      SWITCH TO SET OFF THE INDICATON
                               5334+*                                      * FOR SCANNING A COMMA
                          1BE8 5335+SCANIT EQU   *                         ENTRY POINT TO THIS SUBROUTINE
    1BE8 34 08 1C24            5336+       ST    SCA500+@OP1,@ARR          SAVE RETURN ADDRESS
    1BEC 34 02 1C26            5337+       ST    SCASVE,@XR                SAVE POINTER VALUE
    1BF0 3C 04 03CD            5338+       MVI   $CAERR,@@E110             SET ERROR CODE
    1BF4 F2 87 03              5339+       J     SCA200                    GO TO PROCESS
    1BF7 E2 02 01              5340+SCA100 LA    SCAINC(,@XR),@XR          INCREMENT POINTER TO NEXT CHAR
    1BFA BD 40 00              5341+SCA200 CLI   0(,@XR),@BLANK            IS THIS CHAR BLANK ?
    1BFD C0 81 1BF7            5342+       BE    SCA100                    YES, FETCH NEXT ONE
    1C01 BD 6B 00              5343+       CLI   0(,@XR),@COMMA            IS IT A COMMA ?
    1C04 F2 87 10              5344+SCA250 JC    SCA400,@UCB               UCS TO RETURN -- OR NOP IF
                               5345+*                                      * SCAMMA IS ACTIVE AND CHAR
    1C07 E2 02 01              5346+SCA300 LA    SCAINC(,@XR),@XR          INCREMENT POINTER TO NEXT CHAR
    1C0A BD 40 00              5347+       CLI   0(,@XR),@BLANK            IS THIS CHAR A BLANK ?
    1C0D C0 81 1C07            5348+       BE    SCA300                    YES, FETCH NEXT ONE
    1C11 BD 1F 00              5349+       CLI   0(,@XR),@EOS+1            IS THIS EOS ?
    1C14 F2 82 0A              5350+       JL    SCA500                    IF NOT, SKIP ERROR ROUTINE
    1C17 34 02 1C28            5351+SCA400 ST    SCACNT,@XR                SAVE NEW POINTER VALUE
    1C1B 0F 01 1C28 1C26       5352+       SLC   SCACNT(2),SCASVE          SET PSR TO EQUAL IF POINTER
                               5353+*                                      * NOT ADVANCED
    1C21 C0 87 0000            5354+SCA500 B     *-*                       YES, RETURN
                          1C05 5355+SCAMMA EQU   SCA250+@Q                 TO SET SCAN COMMA INDICATOR
                               5356+*
                               5357+*                  SAVE AREA
                               5358+*
                          1C25 5359+SCASV1 EQU   *                         FIRST BYTE OF SCASVE
    1C25                  1C26 5360+SCASVE DS    CL2                       ORIGINAL POINTER VALUE SAVE
    1C27                  1C28 5361+SCACNT DS    CL2                       SAVE AREA FOR TOTAL CHAR SCAN
                               5362+***                       END OF SCANIT                            ***
                          FFFF 5363        END

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

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                          VER 15, MOD 00  03/02/22  PAGE  81
$$$$$$  001  0C00 2178
$$$CMD  001  0020 0659
$$$DAT  001  0040 0658
$$$EPL  001  0091 0655
$$$ERN  001  0080 0709
$$$FUN  001  0010 0660
$$$NLN  001  00A0 0705
$$$STD  001  0081 0654
$$$001  015  0C53 2217
$$BNLN  001  0605 0635   0637
$$CDBS  001  08C0 0685
$$CDND  001  0666 0644
$$CDRD  001  0890 0683   0685
$$CKEY  001  0603 0633
$$CKFF  001  0B3D 0665
$$COFF  001  0B44 0664
$$CSNS  001  209C 0694
$$DATB  001  0BBF 0666
$$EOSA  001  0AFE 0663
$$ERSK  001  1C00 0704
$$FITS  001  1D00 0712   2227
$$FLIB  001  06FF 0711   2557*
$$ILEN  001  0601 0629   0631  0635
$$ILHD  001  0600 0627   0629  4163  4164
$$INLN  001  0607 0642   0644  0646  4279  4293
$$INND  001  06FA 0646   4278* 4279  4279  4279*
$$KBDT  001  09E1 0653   0657
$$KBSN  001  09E2 0657   0662
$$KLD1  001  0600 0717
$$KLD2  001  0700 0719   2193  2194  2317
$$KLD3  001  0C00 0721
$$LPOS  001  09EB 0662
$$PCNT  001  07E9 0678
$$PLYN  001  2004 0692
$$PRES  001  0890 0651   0653  0663  0664  0665  0666  0683  4281
$$PRFL  001  2143 0696
$$PRNT  001  0707 0672   0673  0677  0678
$$PRTN  001  0782 0673
$$PSIO  001  07CE 0677
$$PYCD  001  2200 0698
$$PYMP  001  2000 0690   0692  0694  0696  0698
$$SLIB  001  1C00 0707
$$TPCD  001  0606 0637   0642
$$UPAR  001  0602 0631   0633
$$WSPB  001  1E00 0710
$$XIND  001  06FF 0708   0711  4163  4164  4276  4318*
$$ZERO  001  0000 0223   0224  0226  0227  0228  0232  0690
$ABORT  001  0010 0336
$BASIC  001  0080 0394   2437  2718
$BIGCD  001  0080 0470
$BLDPL  001  0579 0603   0605
$BLNOE  001  0569 0593
$BLOAD  001  0522 0584   0586  0589  0602  0603
$BLRTN  001  0550 0592   0593
$BRSAV  001  03C5 0281   0282
$BSADR  001  0587 0608   0610
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  03/02/22  PAGE  82
$BUFPT  001  03E3 0489   0490
$CABLD  001  04B4 0562   0563
$CAERK  001  0469 0539   0542  2276
$CAERR  001  03CD 0287   0289  2231* 2252* 2255* 2378* 2381* 2391* 2753* 2803* 2808* 2816* 3173*
                         3949* 3980* 4202* 4214* 4250* 4320* 4570* 4572* 4798* 4815* 4819* 4836*
                         4841* 4843* 5003* 5219* 5338*
$CAIPL  001  049D 0558   0560
$CALLI  001  0008 0479
$CARDI  001  0001 0250   4249
$CARPL  001  04A1 0560   0562  2700
$CIENT  001  0483 0549   0550
$CIEXT  001  0480 0548   0549
$CIMSK  001  0476 0545   0548  2359* 4263
$CISUS  001  0496 0553   0558
$CLBFR  001  0010 0437
$CMDKY  001  0008 0349
$CMODE  001  0002 0399
$CONFG  001  03DD 0462   0472
$CRPOS  001  03E2 0488   0489
$CRTAD  001  044D 0527   0528
$CRTAV  001  0002 0343
$CRTDN  001  0002 0367
$CRTIN  001  03D3 0364   0371
$CRTNO  001  0004 0346
$CRTPU  001  0004 0368
$CRTSP  001  0008 0369
$CRTUP  001  0001 0366
$CRUSH  001  0080 0475
$CSDPL  001  050E 0574   0575
$C0001  001  0464 0531   0537
$DATE   001  043A 0512   0513  2705
$DBGUF  001  03E0 0474   0483  2439  2721
$DBLOK  001  0001 0424   2390  2714
$DFDET  001  03E8 0495   0496
$DISKN  001  0025 0226   2363  2403  2725  2734  2773  3041  3178  5010
$DKERR  001  0008 0405
$DKSIZ  001  03D7 0449   0457  0498
$DK100  001  0001 0451
$DK200  001  0002 0452
$DK400  001  0004 0453
$DK600  001  0008 0454
$DK800  001  0010 0455
$DPLSV  001  0449 0523   0525  2556*
$DTNMB  001  0040 0270
$DTRDR  001  0040 0358
$ENDNU  001  0600 0617   0627  0651  0672  0708  0717  0719  0721
$ERDPL  001  046F 0542   0544
$ERFIL  001  0040 0297
$ERHRD  001  0004 0429
$ERKEY  001  0080 0301
$ERLOG  001  0345 0231
$ERMAD  001  0472 0544   0545
$ERPND  001  0004 0402
$ERRCT  001  03CF 0303
$ERRPG  001  03CE 0291
$ERSFL  001  0035 0296
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  03/02/22  PAGE  83
$ERSTK  001  0030 0294
$ER050  001  0363 0232
$ER1N2  001  0050 0299
$EXADR  001  0517 0577   0579
$EXCMD  001  0001 0331
$EXFTR  001  043B 0513   0518  2407  2408
$FCIND  001  0010 0409
$FDIND  001  0040 0416
$FEARR  001  0004 0224
$FEMAP  001  0588 0610   0611
$FILIB  001  03DA 0460   0461  3892  3932  3934  3947  3954  3955
$FITIN  001  0010 0385
$FUIND  001  0020 0414
$GUFIO  001  0583 0607   0608
$GUFIR  001  0008 0259
$HISTE  001  042E 0510   0511
$HIST1  001  0435 0511   0512
$HRDER  001  0020 0355
$INDR1  001  03D4 0371   0397  2385  2410  2437  2712  2718  2778
$INDR2  001  03D5 0397   0422
$INDR3  001  03D6 0422   0449  2390  2714
$INLNO  001  03CF 0289   0291  0303  0310
$INRPT  001  0020 0267
$IOIND  001  03D2 0338   0364
$IOPGS  001  0010 0478   2439  2721
$IOYES  001  0002 0253
$IPLDV  001  05FF 0614   0617
$IRKEY  001  0020 0477
$KEYBD  001  03E1 0483   0488
$KEYCD  001  03C3 0247   0281  4249  4282  4330*
$KEYDT  001  0040 0391
$KE090  001  00DE 0227
$KE130  001  01D5 0228
$KSAVE  001  0C07 2181
$KYBSY  001  0010 0264   4282
$LDRTN  001  0571 0602
$LEVEL  001  03DF 0472   0474
$LIST   001  0002 0426
$LMRGN  001  03C1 0242   0244
$LNPTR  001  0080 0361
$LOADB  001  054A 0586
$LOADR  001  051A 0579   0582  2369  2560
$LPRIO  001  03EA 0496
$LPROS  001  03E5 0491   0493
$LPRP3  001  03E4 0490   0491
$MOUNT  001  0020 0440
$MPDWN  001  0001 0340
$NEXTB  001  03E6 0493   0494
$NEXTL  001  03E7 0494   0495
$NOENB  001  0008 0432
$NOLST  001  0004 0256
$NUCBS  001  03C0 0239   0240  3876  3877
$NWRKF  001  0080 0445
$NWRKR  001  0040 0442
$PASWD  001  042D 0509   0510
$PAUSD  001  04BA 0563   0565
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  03/02/22  PAGE  84
$PAUSE  001  0002 0333
$PGMDT  001  0020 0388   2410  2778
$PGMST  001  0010 0352
$PKERT  001  0419 0507   0509
$PLST1  001  0454 0528   0529
$PLST2  001  045B 0529   0530
$PLST3  001  0462 0530   0531
$PRDEV  001  044B 0525   0527
$PRESN  001  0002 0376
$PROCI  001  0001 0373
$PRPOS  001  03C2 0244   0247
$PSDBR  001  04FA 0568
$PSDXR  001  04F2 0567   0568
$PSTEP  001  0004 0334
$PSTMT  001  0008 0335
$PTCH1  001  03F5 0498   0502
$READY  001  0080 0418
$REORD  001  0040 0476
$RLOAD  001  051E 0582   0584  2566
$RMRGN  001  03C0 0240   0242
$RSTR   001  04D6 0565   0567  0569  0574
$RUNIT  001  0001 0312
$SFAID  001  050D 0570
$SPRNT  001  0465 0537   0539  4260  4264  4284
$SRTRN  001  04FE 0569   0570
$STEPT  001  0002 0313
$SWPCR  001  0511 0575   0577
$TABLN  001  03CB 0284   0287
$TFLOW  001  0008 0319
$TRACE  001  0004 0314
$TRALL  001  0010 0320
$TROVR  001  054E 0589   0592
$TRUNK  001  0080 0272   4330
$TRVAR  001  0020 0321
$UNMSK  001  048D 0550   0553  4280
$USRDR  001  03DC 0461   0462  3956  3959
$VMDEF  001  0080 0325
$VOLF1  001  03FE 0504   0505  3911  3913
$VOLF2  001  040E 0506   3917  3919
$VOLID  001  03F6 0502   0503  0507  3880  3882
$VOLR1  001  03F6 0503   0504  3923  3925
$VOLR2  001  0406 0505   0506  3905  3907
$WAITF  001  057F 0605   0607  2364  2404  2735  2774  3179  4265  4285  5011
$WFDEF  001  0040 0519
$WFLOK  001  0008 0382   2385
$WFNME  001  0443 0518   0523  2236
$WSIND  001  0004 0379
$XIND1  001  03D0 0310   0329
$XIND2  001  03D1 0329   0338
$XIND3  001  03D8 0457   0460
$XPREC  001  0040 0322
$XRSAV  001  03C7 0282   0284  2221  2367  2558  2564
$ZTRAD  001  05A2 0611
$12K    001  0004 0466
$16CKY  001  0008 0468
$16K    001  0002 0465
```

```
$22IMP  001  0001 0463
#$$#BL  001  0000 1241
#$$#CK  001  0000 1369
#$$#CN  001  0000 1337
#$$#CO  001  0000 1129
#$$#CS  001  0000 1189
#$$#DR  001  0000 0933
#$$#ER  001  0000 1133
#$$#FS  001  0000 1229
#$$#IN  001  0000 1373
#$$#PW  001  0000 1377
#$$#RS  001  0000 1209
#$$#SA  001  0000 1197
#$$#SS  001  0000 1193
#$$#VU  001  0600 1153
#$$#0T  001  0700 0925
#$$#1T  001  0000 0929
#$$BCO  001  0600 0941
#$$BOV  001  0800 1213
#$$DPR  001  0700 0949   2341
#$$DRE  001  0889 0965
#$$DSP  001  2800 0985
#$$ECM  001  0C00 1245
#$$EFK  001  0C00 1265
#$$ERR  001  0C00 1237
#$$EXM  001  0C00 1125
#$$FIL  001  0E00 1205
#$$FIS  001  0E00 1201
#$$FML  001  0200 1333
#$$FMS  001  0200 1173
#$$GRA  001  0889 1097
#$$GUF  001  0C00 1233
#$$INL  001  0600 1313
#$$INS  001  0600 0937
#$$KAL  001  0C00 1101
#$$KCA  001  0C00 1317
#$$KCH  001  0C00 1069
#$$KCN  001  0C00 1185
#$$KCT  001  0C00 1037
#$$KDE  001  0C00 1033
#$$KDI  001  0D00 1113
#$$KDN  001  0C00 1021
#$$KDO  001  0E00 1117
#$$KED  001  0C00 0957
#$$KEN  001  0C00 0961
#$$KEX  001  0C00 0981
#$$KGO  001  0C00 0953
#$$KHE  001  0C00 1137
#$$KKE  001  0C00 1365
#$$KLI  001  0C00 1041
#$$KLL  001  0920 1341
#$$KLO  001  0C00 1045
#$$KME  001  0D00 1025
#$$KMO  001  0C00 0969
#$$KNA  001  0C00 1081
#$$KOV  001  0E00 1001
```

```
SYMBOL  LEN VALUE DEFN   REFERENCES
#$$KPA  001  0C00 0977
#$$KPO  001  0C00 1065
#$$KPR  001  0C00 1089
#$$KRE  001  0C00 1009
#$$KRL  001  0700 1105
#$$KRM  001  0C00 0973
#$$KRN  001  0700 0993
#$$KRO  001  0D00 0997
#$$KRS  001  0C00 1321
#$$KRU  001  0C00 1017
#$$KRV  001  0800 1109
#$$KSA  001  0C00 1053   2177  2324
#$$KSE  001  0E00 1093
#$$KSO  001  0C20 1145
#$$KSS  001  0C00 1077
#$$KSV  001  0980 1073   2198  2350
#$$KSY  001  0C00 1085
#$$KWI  001  0C00 1013
#$$KWR  001  0C00 1005
#$$LOA  001  0600 0945
#$$MIP  001  0C00 1141
#$$SDS  001  0C00 1253
#$$SFF  001  0E00 1257
#$$SFL  001  0F00 1249
#$$SFO  001  1500 1221
#$$SFS  001  0C00 1217
#$$SPA  001  0C00 1057   2333
#$$SPO  001  0806 1061
#$$SPS  001  0C00 1049
#$$STR  001  1600 1225
#$$TDC  001  1000 1029
#$$TSY  001  1000 0989
#$$TVK  001  0FC0 1165
#$$UAL  001  0C00 1181
#$$UAT  001  0900 1277
#$$UCD  001  0900 1285
#$$UCN  001  0C00 1269
#$$UCP  001  0700 1273
#$$UDE  001  0C00 1289
#$$UDI  001  0C00 1293
#$$UEX  001  0C00 1177
#$$UIN  001  0C00 1281
#$$UPA  001  0C00 1261
#$$UPO  001  0C00 1329
#$$UPT  001  0C00 1325
#$$VCR  001  2000 1121
#$$VLO  001  0600 1157
#$$VOD  001  0600 1161
#$$VVM  001  0000 1169
#$$VXI  001  0600 1149
#$$ZDU  001  1100 1301
#$$ZLB  001  1100 1345
#$$ZLO  001  1100 1305
#$$ZLV  001  0F00 1361
#$$ZL1  001  0F00 1349
#$$ZL2  001  0F00 1353
```

```
#$$ZL3  001  0C00 1357
#$$ZTR  001  1000 1297
#$$ZUT  001  0C00 1309
#$#BLN  001  18D4 1240
#$#CKT  001  2118 1368
#$#CNF  001  2000 1336
#$#COR  001  0800 1128
#$#CSA  001  1000 1188
#$#DRT  001  0000 0932
#$#ERM  001  0928 1132
#$#FSP  001  1880 1228
#$#INV  001  212C 1372
#$#PWR  001  2300 1376
#$#RSP  001  1780 1208
#$#SAV  001  1180 1196
#$#SSA  001  1128 1192
#$#VUF  001  0B08 1152
#$#0TR  001  0000 0924
#$#1TR  001  0080 0928
#$@#BL  001  0001 1242
#$@#CK  001  0004 1370
#$@#CN  001  0001 1338
#$@#CO  001  003A 1130
#$@#CS  001  003A 1190
#$@#DR  001  0008 0934
#$@#ER  001  0032 1134
#$@#FS  001  0030 1230
#$@#IN  001  003A 1374
#$@#PW  001  00C0 1378
#$@#RS  001  0030 1210
#$@#SA  001  0108 1198
#$@#SS  001  0001 1194
#$@#VU  001  0002 1154
#$@#0T  001  0018 0926
#$@#1T  001  0018 0930
#$@BCO  001  0018 0942
#$@BOV  001  0018 1214
#$@DPR  001  0005 0950   2340
#$@DRE  001  0001 0966
#$@DSP  001  0004 0986
#$@ECM  001  0006 1246
#$@EFK  001  0002 1266
#$@ERR  001  0003 1238
#$@EXM  001  0003 1126
#$@FIL  001  0009 1206
#$@FIS  001  0009 1202
#$@FML  001  0052 1334
#$@FMS  001  0052 1174
#$@GRA  001  0003 1098
#$@GUF  001  0010 1234
#$@INL  001  0010 1314
#$@INS  001  0010 0938
#$@KAL  001  000F 1102
#$@KCA  001  000C 1318
#$@KCH  001  000C 1070
#$@KCN  001  0010 1186
```

```
                                        CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                         VER 15, MOD 00  03/02/22  PAGE  88
#$@KCT  001  0009 1038
#$@KDE  001  0010 1034
#$@KDI  001  0005 1114
#$@KDN  001  0010 1022
#$@KDO  001  000C 1118
#$@KED  001  000E 0958
#$@KEN  001  0006 0962
#$@KEX  001  0003 0982
#$@KGO  001  0002 0954
#$@KHE  001  000C 1138
#$@KKE  001  0006 1366
#$@KLI  001  0011 1042
#$@KLL  001  0001 1342
#$@KLO  001  0008 1046
#$@KME  001  0003 1026
#$@KMO  001  0004 0970
#$@KNA  001  0008 1082
#$@KOV  001  0009 1002
#$@KPA  001  0005 0978
#$@KPO  001  000D 1066
#$@KPR  001  0009 1090
#$@KRE  001  0002 1010
#$@KRL  001  0004 1106
#$@KRM  001  0003 0974
#$@KRN  001  0003 0994
#$@KRO  001  000A 0998
#$@KRS  001  000A 1322
#$@KRU  001  0003 1018
#$@KRV  001  000D 1110
#$@KSA  001  0011 1054   2323
#$@KSE  001  0004 1094
#$@KSO  001  000D 1146
#$@KSS  001  000B 1078
#$@KSV  001  0002 1074   2349
#$@KSY  001  000F 1086
#$@KWI  001  0002 1014
#$@KWR  001  0002 1006
#$@LOA  001  0013 0946
#$@MIP  001  000D 1142
#$@SDS  001  0004 1254
#$@SFF  001  0008 1258
#$@SFL  001  0005 1250
#$@SFO  001  0003 1222
#$@SFS  001  0011 1218
#$@SPA  001  0004 1058   2332
#$@SPO  001  0003 1062
#$@SPS  001  0001 1050
#$@STR  001  0002 1226
#$@TDC  001  0003 1030
#$@TSY  001  0003 0990
#$@TVK  001  0001 1166
#$@UAL  001  0011 1182
#$@UAT  001  000C 1278
#$@UCD  001  000B 1286
#$@UCN  001  0009 1270
#$@UCP  001  000F 1274
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  03/02/22  PAGE  89
#$@UDE  001  000E 1290
#$@UDI  001  0008 1294
#$@UEX  001  000E 1178
#$@UIN  001  000F 1282
#$@UPA  001  0004 1262
#$@UPO  001  0005 1330
#$@UPT  001  0012 1326
#$@VCR  001  0008 1122
#$@VLO  001  0002 1158
#$@VOD  001  0016 1162
#$@VVM  001  0030 1170
#$@VXI  001  0002 1150
#$@ZDU  001  0008 1302
#$@ZLB  001  0002 1346
#$@ZLO  001  000C 1306
#$@ZLV  001  0006 1362
#$@ZL1  001  0007 1350
#$@ZL2  001  000D 1354
#$@ZL3  001  000A 1358
#$@ZTR  001  0001 1298
#$@ZUT  001  0014 1310
#$BCOM  001  0080 0940
#$BOLV  001  1780 1212
#$DPRI  001  014C 0948   2339
#$DREA  001  0200 0964
#$DSPL  001  0240 0984
#$ECMA  001  1900 1244
#$EFKE  001  1990 1264
#$ERRP  001  18C0 1236
#$EXMS  001  07D4 1124
#$FILN  001  1724 1204
#$FIST  001  1700 1200
#$FMLN  001  1E00 1332
#$FMST  001  0D00 1172
#$GRAP  001  0690 1096
#$GUFU  001  1880 1232
#$INLN  001  1C84 1312
#$INST  001  0020 0936
#$KALL  001  06A4 1100
#$KCAL  001  1CC4 1316
#$KCHA  001  053C 1068
#$KCND  001  0F80 1184
#$KCTL  001  03BC 1036
#$KDEL  001  035C 1032
#$KDIS  001  0744 1112
#$KDNT  001  0300 1020
#$KDOV  001  0780 1116
#$KEDI  001  0188 0956
#$KENA  001  01C4 0960
#$KEXT  001  0234 0980
#$KGOS  001  0180 0952
#$KHEL  001  0A30 1136
#$KKEY  001  2100 1364
#$KLIS  001  0400 1040
#$KLLA  001  2004 1340
#$KLOG  001  0444 1044
```

```
SYMBOL  LEN VALUE DEFN   REFERENCES
#$KMER  001  030C 1024
#$KMOU  001  0204 0968
#$KNAM  001  05C0 1080
#$KOVM  001  0290 1000
#$KPAS  001  0220 0976
#$KPOO  001  0508 1064
#$KPRT  001  063C 1088
#$KREA  001  02BC 1008
#$KRLA  001  0700 1104
#$KRMO  001  0214 0972
#$KRNU  001  0280 0992
#$KROV  001  028C 0996
#$KRSU  001  1D24 1320
#$KRUN  001  02CC 1016
#$KRVL  001  0710 1108
#$KSAV  001  0488 1052   2322
#$KSET  001  0680 1092
#$KSOV  001  0AC8 1144
#$KSSP  001  0594 1076
#$KSVL  001  058C 1072   2348
#$KSYM  001  0600 1084
#$KWID  001  02C4 1012
#$KWRI  001  02B4 1004
#$LOAD  001  0100 0944
#$MIPP  001  0A80 1140
#$SDSY  001  192C 1252
#$SFFI  001  193C 1256
#$SFLO  001  1918 1248
#$SFOV  001  1844 1220
#$SFSY  001  1800 1216
#$SPAC  001  04CC 1056   2331
#$SPOV  001  04DC 1060
#$SPSY  001  0484 1048
#$STRO  001  1850 1224
#$TDCK  001  0350 1028
#$TSYK  001  0250 0988
#$TVKB  001  0BAC 1164
#$UALL  001  0F00 1180
#$UATR  001  1A38 1276
#$UCDI  001  1AD8 1284
#$UCNF  001  19B8 1268
#$UCPL  001  19DC 1272
#$UDEL  001  1B24 1288
#$UDIS  001  1B5C 1292
#$UEXL  001  0EA8 1176
#$UINI  001  1A88 1280
#$UPAC  001  1980 1260
#$UPOV  001  1D24 1328
#$UPTF  001  1D5C 1324
#$VCRT  001  07B4 1120
#$VLOA  001  0B80 1156
#$VODK  001  0B88 1160
#$VVMR  001  0C00 1168
#$VXIT  001  0B00 1148
#$ZDUM  001  1BA4 1300
#$ZLBM  001  2008 1344
```

```
#$ZLOA  001  1BC4 1304
#$ZLVR  001  20B0 1360
#$ZL1M  001  2010 1348
#$ZL2M  001  2030 1352
#$ZL3M  001  2088 1356
#$ZTRA  001  1B9C 1296
#$ZUTM  001  1C14 1308
##DNEA  001  0001 0846   3373  3390  3393*
##DNEF  001  0003 0847   3349  3392* 3431  3458  3474* 3494  3533
##DNER  001  0005 0848   3368* 3438  3470* 3578*
##DNE1  001  0004 0845   3372
##DNHC  001  0000 0842   3364*
##DNHR  001  0003 0844
##DNHY  001  0001 0843
##DPEA  001  0009 0820   5028  5033
##DPEN  001  0007 0819   4499* 4503* 4519  4532  4886  4887  5017
##DPER  001  000B 0821
##DPE1  001  0004 0818   5015
##DPHC  001  0000 0816   3353  3354  5014
##DPHR  001  0003 0817
##DUEA  001  0009 0831   2500  2501  2527  2708* 2821
##DUED  001  0012 0836   2705*
##DUEF  001  000B 0832   2507  2525  2536  2586  2706* 2814
##DUEH  001  002B 0837   2498  2711*
##DUEI  001  000C 0833   2707* 3196
##DUEL  001  000F 0835   2709*
##DUEN  001  0007 0830   2710* 3199  4516
##DUER  001  0031 0838   3680
##DUES  001  000D 0834   2377  2380  2383* 2712* 2713* 2716*
##DUE1  001  000C 0829   3691
##DUHA  001  0001 0825   2840  3216  3685  3727*
##DUHB  001  0003 0826   3187  3190  3716* 3727  3728
##DUHC  001  0004 0827   3195  3686  3689  3701* 3726*
##DUHR  001  000B 0828
##LAAA  001  0002 0857   2830  2855
##LAHC  001  0001 0856   3353  3354  3436  3471  3499
##LN    001  0001 0885   2316
##LNE   001  0006 0891   3368  3379  3448  3469  3470  3496  3498  3518  3578
##LNEF  001  0002 0889   3373  3392  3425  3431  3474  3490  3491  3493  3503  3534  3538  3563
                         3565  3566
##LNEZ  001  0002 0890   3368  3470  3495  3578
##LNH   001  0004 0888   3368* 3506
##LNHY  001  0001 0886
##LNHZ  001  0002 0887
##LP    001  0004 0861   5052
##LPE   001  000C 0866   5019
##LPEN  001  0008 0863   2397  2750  2752  3878  3888  4488  4532  4772  4814  5017
##LPEZ  001  0002 0864
##LPH   001  0004 0865
##LPHZ  001  0003 0862
##LU    001  0002 0870   2294  3242  3742  3746  3749
##LUE   001  0032 0881   2873  3201  3696  3699  3699* 3722  3757
##LUED  001  0003 0878   2705
##LUEF  001  0002 0874   2536  2537  2586  2706  2830  2831  2886
##LUEH  001  0019 0879   2235  2238  2498  2711  2711
##LUEI  001  0001 0875   2707
```

```
##LUEL  001  0002 0877   2709  2884
##LUEN  001  0008 0873   2236  2710  3199  4516  4742  4884
##LUES  001  0001 0876
##LUEZ  001  0006 0880
##LUH   001  000C 0872   3755
##LUHZ  001  0007 0871
##MNHM  001  002A 0914   3358
##MPHM  001  0055 0899
##MUEG  001  0020 0906
##MUEK  001  0040 0905
##MUEO  001  0004 0909   2196  2383
##MUEP  001  0080 0904
##MUER  001  0008 0908   2196  2380  2716
##MUEV  001  0002 0910
##MUEX  001  0010 0907   2196  2377
##MUHM  001  000A 0903   3686
##RN    001  0000 0805
##RP    001  0001 0806   5051  5056
##R1    001  0007 0808
##R2    001  0005 0807
#@#BAD  001  0455 0749
#@#IO1  001  0459 0757   2307
#@#IO2  001  045D 0758
#@#TAT  001  0941 0785
#@#TBA  001  09A1 0789
#@#TFS  001  0941 0783
#@#TSY  001  0941 0787
#@#VFP  001  0700 0775
#@#VLP  001  093D 0778
#@#WDB  001  050C 0770
#@#WFT  001  0500 0768
#@@#BA  001  0001 0750
#@@#IO  001  0001 0762   2308
#@@#SC  001  0002 0759   2724
#@@#TA  001  0010 0786
#@@#TB  001  0010 0790
#@@#TS  001  0005 0788
#@@#TW  001  0020 0784
#@@#VM  001  0100 0779
#@@#WD  001  00BD 0771
#@@#WF  001  0003 0769
#@@#04  001  0004 0761
#@@#08  001  0008 0760
#@@BOV  001  0018 0738
#@@ECM  001  0006 0752
#@@ERR  001  0003 0746
#@@GUF  001  0010 0742
#@@LDS  001  0002 0748
#@@SDS  001  0004 0744
#@@SFF  001  0008 0756
#@@SFL  001  0005 0754
#@@SFO  001  0005 0764
#@@SFS  001  0011 0740
#@@VSF  001  0010 0792
#@@VSL  001  000F 0793
#@@VTR  001  0001 0777
```

```
SYMBOL  LEN VALUE DEFN   REFERENCES
#@BOVL  001  0400 0737
#@ECMA  001  0481 0751
#@ERRP  001  0441 0745
#@GUFU  001  0401 0741
#@LDSV  001  044D 0747
#@SDSY  001  04AD 0743
#@SFFI  001  04BD 0755
#@SFLO  001  0499 0753
#@SFOV  001  04C4 0763
#@SFSY  001  0480 0739
#@VSFI  001  09A1 0791
#@VTRL  001  0708 0776
#@WAF1  001  0401 0736
#@WAR1  001  0400 0735
#KSAVE  001  0000 0001
@@E001  001  0000 1915   1917
@@E003  001  0001 1917   1919
@@E004  001  0002 1919   1921
@@E005  001  0003 1921   1923
@@E006  001  0004 1923   1925
@@E007  001  0005 1925   1927
@@E008  001  0006 1927   1929
@@E009  001  0007 1929   1931
@@E010  001  0008 1931   1933
@@E011  001  0009 1933   1935
@@E012  001  000A 1935   1937
@@E013  001  000B 1937   1939
@@E014  001  000C 1939   1941
@@E015  001  000D 1941   1943
@@E016  001  000E 1943   1945
@@E017  001  000F 1945   1947
@@E018  001  0010 1947   1949
@@E019  001  0011 1949   1951
@@E020  001  0012 1951   1953
@@E021  001  0013 1953   1955
@@E023  001  0014 1955   1957
@@E024  001  0015 1957   1959
@@E025  001  0016 1959   1961
@@E026  001  0017 1961   1963
@@E027  001  0018 1963   1965
@@E028  001  0019 1965   1967
@@E029  001  001A 1967   1969
@@E030  001  001B 1969   1971
@@E031  001  001C 1971   1973
@@E032  001  001D 1973   1975
@@E035  001  001E 1975   1977
@@E036  001  001F 1977   1979
@@E037  001  0020 1979   1981
@@E038  001  0021 1981   1983
@@E039  001  0022 1983   1985
@@E040  001  0023 1985   1987
@@E041  001  0024 1987   1989
@@E042  001  0025 1989   1991
@@E043  001  0026 1991   1993
@@E044  001  0027 1993   1995
@@E045  001  0028 1995   1997
```

```
@@E046  001  0029 1997   1999
@@E060  001  002A 1999   2001
@@E080  001  002B 2001
@@E100  001  0000 1387   1389  4798  4841
@@E101  001  0001 1389   1391  4843
@@E102  001  0002 1391   1393  4815
@@E103  001  0003 1393   1395  4819
@@E110  001  0004 1395   1397  5338
@@E112  001  0005 1397   1399
@@E113  001  0006 1399   1401
@@E114  001  0007 1401   1403
@@E115  001  0008 1403   1405
@@E116  001  0009 1405   1407
@@E117  001  000A 1407   1409
@@E120  001  000B 1409   1411
@@E122  001  000C 1411   1413
@@E123  001  000D 1413   1415
@@E124  001  000E 1415   1417
@@E129  001  000F 1417   1419
@@E130  001  0010 1419   1421  4836
@@E131  001  0011 1421   1423  2252  2255  4572
@@E133  001  0012 1423   1425
@@E134  001  0013 1425   1427
@@E135  001  0014 1427   1429  2753
@@E136  001  0015 1429   1431
@@E137  001  0016 1431   1433
@@E138  001  0017 1433   1435  5219
@@E139  001  0018 1435   1437  2231  4569
@@E142  001  0019 1437   1439
@@E143  001  001A 1439   1441
@@E150  001  001B 1441   1443
@@E151  001  001C 1443   1445
@@E160  001  001D 1445   1447
@@E162  001  001E 1447   1449
@@E163  001  001F 1449   1451
@@E164  001  0020 1451   1453
@@E200  001  0021 1453   1455  3949
@@E205  001  0022 1455   1457
@@E210  001  0023 1457   1459  5003
@@E211  001  0024 1459   1461  3173
@@E212  001  0025 1461   1463  4250
@@E213  001  0026 1463   1465  3980
@@E215  001  0027 1465   1467  2381
@@E216  001  0028 1467   1469  4320
@@E217  001  0029 1469   1471  4202
@@E220  001  002A 1471   1473
@@E221  001  002B 1473   1475
@@E222  001  002C 1475   1477  2391
@@E223  001  002D 1477   1479
@@E225  001  002E 1479   1481
@@E226  001  002F 1481   1483
@@E227  001  0030 1483   1485
@@E228  001  0031 1485   1487
@@E229  001  0032 1487   1489
@@E230  001  0033 1489   1491
@@E232  001  0034 1491   1493
```

```
@@E234  001  0035 1493   1495
@@E237  001  0036 1495   1497
@@E240  001  0037 1497   1499
@@E241  001  0038 1499   1501
@@E242  001  0039 1501   1503
@@E248  001  003A 1503   1505
@@E249  001  003B 1505   1507
@@E250  001  003C 1507   1509
@@E251  001  003D 1509   1511
@@E252  001  003E 1511   1513
@@E253  001  003F 1513   1515
@@E254  001  0040 1515   1517
@@E255  001  0041 1517   1519
@@E256  001  0042 1519   1521
@@E300  001  0043 1521   1523  2803  2808  2816
@@E301  001  0044 1523   1525
@@E302  001  0045 1525   1527
@@E303  001  0046 1527   1529
@@E304  001  0047 1529   1531
@@E305  001  0048 1531   1533
@@E308  001  0049 1533   1535
@@E310  001  004A 1535   1537  2378
@@E315  001  004B 1537   1539
@@E316  001  004C 1539   1541
@@E320  001  004D 1541   1543
@@E325  001  004E 1543   1545
@@E330  001  004F 1545   1547
@@E335  001  0050 1547   1549
@@E338  001  0051 1549   1551
@@E340  001  0052 1551   1553
@@E350  001  0053 1553   1555
@@E351  001  0054 1555   1557  4214
@@E352  001  0055 1557   1559
@@E360  001  0056 1559   1561
@@E361  001  0057 1561   1563
@@E362  001  0058 1563   1565
@@E371  001  0059 1565   1567
@@E380  001  005A 1567   1569
@@E390  001  005B 1569   1571
@@E400  001  005C 1571   1573
@@E410  001  005D 1573   1575
@@E415  001  005E 1575   1577
@@E417  001  005F 1577   1579
@@E420  001  0060 1579   1581
@@E430  001  0061 1581   1583
@@E432  001  0062 1583   1585
@@E433  001  0063 1585   1587
@@E450  001  0064 1587   1589
@@E451  001  0065 1589   1591
@@E460  001  0066 1591   1593
@@E461  001  0067 1593   1595
@@E464  001  0068 1595   1597
@@E465  001  0069 1597   1599
@@E466  001  006A 1599   1601
@@E467  001  006B 1601   1603
@@E469  001  006C 1603   1605
```

```
@@E470  001  006D 1605   1607
@@E471  001  006E 1607   1609
@@E473  001  006F 1609   1611
@@E474  001  0070 1611   1613
@@E475  001  0071 1613   1615
@@E476  001  0072 1615   1617
@@E477  001  0073 1617   1619
@@E478  001  0074 1619   1621
@@E479  001  0075 1621   1623
@@E480  001  0076 1623   1625
@@E481  001  0077 1625   1627
@@E482  001  0078 1627   1629
@@E483  001  0079 1629   1631
@@E484  001  007A 1631   1633
@@E485  001  007B 1633   1635
@@E486  001  007C 1635   1637
@@E487  001  007D 1637   1639
@@E488  001  007E 1639   1641
@@E489  001  007F 1641   1643
@@E490  001  0080 1643   1645
@@E491  001  0081 1645   1647
@@E492  001  0082 1647   1649
@@E493  001  0083 1649   1651
@@E494  001  0084 1651   1653
@@E495  001  0085 1653   1655
@@E496  001  0086 1655   1657
@@E497  001  0087 1657   1659
@@E498  001  0088 1659   1661
@@E500  001  0089 1661   1663
@@E501  001  008A 1663   1665
@@E530  001  008B 1665   1667
@@E531  001  008C 1667   1669
@@E535  001  008D 1669   1671
@@E540  001  008E 1671   1673
@@E541  001  008F 1673   1675
@@E542  001  0090 1675   1677
@@E543  001  0091 1677   1679
@@E544  001  0092 1679   1681
@@E545  001  0093 1681   1683
@@E546  001  0094 1683   1685
@@E547  001  0095 1685   1687
@@E548  001  FFFF 1891
@@E549  001  0096 1687   1689
@@E550  001  0097 1689   1691
@@E551  001  0098 1691   1693
@@E552  001  0099 1693   1695
@@E553  001  009A 1695   1697
@@E554  001  009B 1697   1699
@@E555  001  009C 1699   1701
@@E556  001  009D 1701   1703
@@E558  001  009E 1703   1705
@@E570  001  009F 1705   1707
@@E571  001  00A0 1707   1709
@@E572  001  00A1 1709   1711
@@E573  001  00A2 1711   1713
@@E574  001  00A3 1713   1715
```

```
@@E575  001  FFFF 1893
@@E578  001  00A4 1715   1717
@@E579  001  FFFF 1895
@@E580  001  FFFF 1897
@@E585  001  00A5 1717   1719
@@E595  001  FFFF 1899
@@E597  001  FFFF 1901
@@E598  001  FFFF 1903
@@E600  001  00A6 1719   1721
@@E601  001  00A7 1721   1723
@@E602  001  00A8 1723   1725
@@E603  001  00A9 1725   1727
@@E604  001  00AA 1727   1729
@@E606  001  00AB 1729   1731
@@E607  001  00AC 1731   1733
@@E608  001  00AD 1733   1735
@@E609  001  00AE 1735   1737
@@E610  001  00AF 1737   1739
@@E611  001  00B0 1739   1741
@@E612  001  00B1 1741   1743
@@E613  001  00B2 1743   1745
@@E614  001  00B3 1745   1747
@@E700  001  00B4 1747   1749
@@E701  001  00B5 1749   1751
@@E710  001  00B6 1751   1753
@@E712  001  00B7 1753   1755
@@E713  001  00B8 1755   1757
@@E714  001  00B9 1757   1759
@@E715  001  00BA 1759   1761
@@E716  001  00BB 1761   1763
@@E717  001  00BC 1763   1765
@@E718  001  00BD 1765   1767
@@E720  001  00BE 1767   1769
@@E721  001  00BF 1769   1771
@@E723  001  00C0 1771   1773
@@E724  001  00C1 1773   1775
@@E725  001  00C2 1775   1777
@@E726  001  00C3 1777   1779
@@E727  001  00C4 1779   1781
@@E728  001  00C5 1781   1783
@@E729  001  00C6 1783   1785
@@E730  001  00C7 1785   1787
@@E732  001  00C8 1787   1789
@@E752  001  00C9 1789   1791
@@E753  001  00CA 1791   1793
@@E754  001  00CB 1793   1795
@@E755  001  00CC 1795   1797
@@E756  001  00CD 1797   1799
@@E757  001  00CE 1799   1801
@@E758  001  00CF 1801   1803
@@E759  001  00D0 1803   1805
@@E760  001  00D1 1805   1807
@@E761  001  00D2 1807   1809
@@E762  001  00D3 1809   1811
@@E763  001  00D4 1811   1813
@@E764  001  00D5 1813   1815
```

```
SYMBOL  LEN VALUE DEFN   REFERENCES
@@E765  001  00D6 1815   1817
@@E766  001  00D7 1817   1819
@@E767  001  00D8 1819   1821
@@E768  001  00D9 1821   1823
@@E769  001  00DA 1823   1825
@@E770  001  00DB 1825   1827
@@E771  001  00DC 1827   1829
@@E772  001  00DD 1829   1831
@@E773  001  00DE 1831   1833
@@E774  001  00DF 1833   1835
@@E775  001  00E0 1835   1837
@@E776  001  00E1 1837   1839
@@E777  001  00E2 1839   1841
@@E778  001  00E3 1841   1843
@@E779  001  00E4 1843   1845
@@E780  001  00E5 1845   1847
@@E781  001  00E6 1847   1849
@@E782  001  00E7 1849   1851
@@E783  001  00E8 1851   1853
@@E784  001  00E9 1853   1855
@@E785  001  00EA 1855   1857
@@E786  001  00EB 1857   1859
@@E790  001  00EC 1859   1861
@@E791  001  00ED 1861   1863
@@E792  001  00EE 1863   1865
@@E793  001  00EF 1865   1867
@@E794  001  00F0 1867   1869
@@E795  001  00F1 1869   1871
@@E796  001  00F2 1871   1873
@@E797  001  00F3 1873   1875
@@E798  001  00F4 1875   1877
@@E800  001  FFFF 1905
@@E801  001  FFFF 1907
@@E802  001  FFFF 1909
@@E803  001  FFFF 1911
@@E804  001  FFFF 1913
@@E900  001  00F5 1877   1879
@@E901  001  00F6 1879   1881
@@E902  001  00F7 1881   1883
@@E903  001  00F8 1883   1885
@@E905  001  00F9 1885   1887
@@E906  001  00FA 1887   1889
@@E910  001  00FB 1889
@@M300  001  0C0A 2207   4261
@@T300  001  0C0E 2211   2209
@ARR    001  0008 0016   2682  2704  2732  2992* 2993  2994* 2995  3172  3342  3516  3531  3676
                         3874  4177  4478  4762  5000  5185  5187* 5188  5336
@ASIGN  001  007C 0071   4787
@ASTER  001  005C 0069   2397  2750  2752  4497  4499  4501  4503  4519
@BCRDL  001  0050 0088
@BE     001  0081 0043
@BF     001  0090 0052
@BH     001  0084 0041
@BL     001  0082 0042   3971
@BLANK  001  0040 0065   2237  2755  2802  3878  3880  4278  4297  4312  4487  4489  4771  5341
                         5347
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  03/02/22  PAGE  99
@BM     001  0082 0054
@BNE    001  0001 0046   5332
@BNH    001  0004 0044
@BNL    001  0002 0045
@BNM    001  0002 0057
@BNOL   001  0020 0050
@BNOZ   001  0008 0049
@BNP    001  0004 0056
@BNZ    001  0001 0058
@BOL    001  00A0 0048
@BOZ    001  0088 0047
@BP     001  0084 0053
@BR     001  0001 0013   2184  2220* 2230  2237  2238  2238  2244  2246  2247  2249  2251  2253
                         2256  2259  2262  2264  2267  2270  2273  2275  2358  2373* 2379  2382
                         2392  2398  2399  2407  2412  2421  2422  2423  2424  2424  2426  2426
                         2427  2428  2428  2433  2434  2435  2435  2436  2441  2441  2442  2442
                         2447  2448  2449  2449  2450  2452  2452  2458  2458  2459  2461  2461
                         2464  2466  2467  2468  2469  2475  2480  2484  2486  2498  2500  2507
                         2525  2527  2536  2537  2537  2538  2539  2540  2540  2541  2541  2543
                         2545  2545  2575  2576  2577  2588  2594  2607  2627  2630  2632  2632
                         2634  2635  2635  2638  2638  2639  2643  2644  2645  2652  2653  2655
                         2655  2656  2666  2666  2667  2668  2668  2683  2683  2686  2686  2693
                         2695  2695  2706  2707  2708  2709  2711  2720  2720  2723  2723  2724
                         2727  2727  2737  2738  2739  2741  2754  2784  2804  2805  2809  2817
                         2825  2980  2989  2991* 2992  2993  2994  2995  2997  2998  2998  2999
                         3000  3000  3002  3002  3003  3004  3004  3008  3008  3009  3013  3013
                         3014  3016  3016  3017  3017  3018  3018  3019  3019  3020  3020  3026
                         3027  3028  3028  3029  3034  3034  3035  3035  3037  3037  3043* 3168
                         3169  3170* 3171  3172  3174  3174  3175  3175  3176  3176  3181  3182
                         3187  3189  3189  3190  3194  3195  3197  3198  3202  3202  3203  3205
                         3205  3206  3206  3207  3207  3208  3214  3217* 3227  3337  3338  3340*
                         3341  3342  3346  3347  3347  3349  3351  3352  3353  3354  3358  3360
                         3361  3368  3369  3371  3373  3375  3376  3376  3378  3384  3384  3385
                         3389  3390  3392  3393  3395  3395  3396  3412  3412  3413  3413  3414
                         3416  3418  3418  3419  3424  3424  3425  3436  3440  3440  3447  3449
                         3450  3450  3451  3458  3460  3465  3465  3467  3470  3471  3474  3483*
                         3516  3521  3521  3523  3523  3524  3524  3525  3531  3536  3536  3537
                         3538  3538  3540  3540  3543  3543  3545  3545  3546  3561  3562  3563
                         3563  3564  3564  3565  3565  3566  3566  3567  3567  3568  3569  3569
                         3570  3570  3571  3571  3572  3572  3573  3573  3578  3579  3673  3674
                         3675* 3676  3677  3681  3682  3684  3685  3689  3693  3695  3695  3697
                         3700  3701  3711  3713  3721  3721  3722  3726  3727  3728  3732* 3871
                         3872* 3873  3874  3875  3890  3891  3899  3902  3908  3914  3920  3924
                         3926  3956  3969  3971  3975  3977  3977  3978  3978  3979  3987* 4020
                         4173  4174  4175* 4176  4177  4188  4189  4190  4190  4192  4192  4193
                         4201  4203  4204  4223* 4251  4277  4294* 4304  4304* 4309  4309* 4319
                         4329  4474  4476  4477* 4478  4515  4531  4545  4569  4575  4584  4586*
                         4758  4760  4761* 4762  4764  4771  4772  4772  4773  4774  4774  4794
                         4797  4800  4809  4811  4811  4812  4813  4814  4816  4818  4820  4825
                         4825  4828  4835  4840  4844  4852  4860* 4995  4997  4998* 4999  5000
                         5006  5013  5014  5020  5020  5021  5031  5033  5037  5038  5038  5041*
                         5184  5191* 5206  5207  5207* 5225*
@BT     001  0010 0051
@BZ     001  0081 0055
@B1     001  0001 0063   2450  2469  2641  2643  2644  2653* 2654  2656* 2765  3878  3880  3888
                         3892  3899  3914  3932  3947  4163  4185  4212  4263  4279* 4293  4294
                         4296  4300  4302  4310  4311  4488  4488* 4489* 4500  4503* 4504  4528
                         4542  4546  4772  4827  4884  4886  4887  5186  5195  5198  5203  5206
                         5207
@CADDR  001  0002 0142   2209  2309  2466  2474  2475  2501  2540  2547  2576  2587  2588  2594
                         2668  2741  2782  2784  2847  2881  2998  3174  3175  3176  3205  3206
                         3207  3232  3233  3234  3243  3347  3390  3395  3412  3413  3424  3450
                         3465  3502  3523  3524  3568  3569  3570  3571  3572  3573  3743
@CARDL  001  0060 0087   0644
@CHARA  001  00C1 0072   4790
@CHARF  001  00C6 0073   4307
@CHARR  001  00D9 0074   4305
@CHARZ  001  00E9 0075   4792
@CLOFF  001  0010 0094
@CLON   001  0011 0093
@COMMA  001  006B 0066   4553  5343
@CPLUS  001  004E 0079
@DADDR  001  0002 0140   2285  2293  2480  2500  2527  2536  2541  2557  2577  2586  2607  2635
                         2708  2840  2882  2885  2997  3062  3187  3189  3190  3216  3241  3373
                         3393  3418  3490  3503  3685  3713  3716  3727  3728  3741  3745  3907
                         3913  3919  3925  3934  3954  3955  3956  3959  4189  4189  4191  4304
                         4309  4329  4329  5028  5033  5038
@DBFR1  001  0004 0129   2310  2686*
@DBFR2  001  0005 0130   2474  2576* 2782  2839* 3684* 3700  5013
@DCALK  001  0001 0081
@DCBCY  001  0009 0115
@DCBT1  001  0050 0117
@DCNT   001  0003 0128   2407* 2643* 2655* 2666  2686  2695* 2724* 2739
@DCST1  001  0040 0116
@DCTRL  001  0000 0125
@DCYL   001  0001 0126   3002*
@DD2    001  0003 0030
@DGET   001  0001 0134   2280  2288  2292  2306  2314  2321  2330  2338  2347  3240  3997  5050
@DOLAR  001  005B 0068   4783
@DOP2   001  0004 0028   2993* 2997* 2998* 3060  3061  3413* 3681* 3682*
@DPLNG  001  0006 0132   2556  2999  3058
@DPOS   001  0000 0133
@DPUT   001  0002 0135   2284  2575  2617  2634  2832  2853  3740  3744
@DSAD   001  0002 0127   2577* 2666* 2683* 2840* 3000* 3004* 3008  3009* 3013* 3016* 3020  3026*
                         3034* 3037* 3059  3685* 3728* 5033* 5038*
@DSBCY  001  0004 0106
@DSCS1  001  0000 0107
@DSIVF  001  0003 0138
@DSPIN  001  0002 0131
@DTRSZ  001  0018 0085
@DVBCY  001  0007 0108
@DVRFY  001  0031 0136
@DWAIT  001  00FF 0137
@DWBCY  001  0005 0103
@DWSIZ  001  00C0 0105
@DWTB1  001  0003 0104
@DZERO  001  00F0 0064
@D1     001  0002 0026   4201
@EOF    001  001C 0077
@EOFTC  001  0075 0162
@EOS    001  001E 0076   2239  2250  2263  2271  4315  4551  4838  5201  5349
@FDDBC  001  0000 0195
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  03/02/22  PAGE 101
@FDE1   001  000C 0200   2190
@FDFNA  001  000B 0198
@FDHLN  001  0002 0208
@FDLNC  001  0002 0193   2434
@FDNSC  001  0003 0210
@FDSD   001  0000 0206   2227
@FLACE  001  0009 0197
@FLDBC  001  0001 0196
@FLENT  001  0004 0201   2191  2197
@FLFNA  001  0002 0199
@FLHLN  001  0002 0209
@FLLNC  001  0002 0194   2434
@FLNSC  001  0001 0211
@FLSD   001  0001 0207
@HDRLN  001  0007 0092   0672  2198
@IAR    001  0010 0017
@INDEX  001  0001 0156   0157
@INST3  001  0003 0032
@INST4  001  0004 0033
@INST5  001  0005 0034
@INST6  001  0006 0035
@I1IAR  001  00C0 0020
@LINSZ  001  00F4 0084   0646
@MAPEN  001  0005 0089
@MINCR  001  2000 0083
@MINUS  001  0060 0080   2232
@NOP    001  0080 0040   2247  2253  2359  2443  2465  2497  2616  2626  2669  2675  2733  2762
                         2763  2764  3039  3194  3346  3360  3389  4226  4275  4548  5005  5037
                         5190
@NUMBR  001  007B 0070   4785
@OPD2   001  0004 0029
@OP1    001  0003 0027   2466* 2472  2547  2587  2682* 2704* 2732* 2737* 2741* 2989* 2995* 3168*
                         3171* 3172* 3338* 3341* 3342* 3347* 3352* 3395* 3424* 3432  3437  3453
                         3516* 3519  3531* 3560  3673* 3676* 3677* 3986  3988  3990  4174* 4176*
                         4177* 4476* 4478* 4556* 4574  4760* 4762* 4764* 4774* 4825* 4997* 4999*
                         5000* 5184* 5185* 5186* 5188* 5336*
@OP2    001  0005 0031   3520
@PCTRL  001  0000 0149
@PDATA  001  0003 0151
@PGCSZ  001  0020 0082   0083
@PPLNG  001  0004 0148
@PRCNT  001  0001 0150
@PRETR  001  00C0 0154   2207
@PRINT  001  0040 0152   0154
@PSR    001  0004 0015   4575* 4584* 4800* 5218*
@PWAIT  001  00FF 0158
@P1IAR  001  0020 0018
@P2IAR  001  0040 0019
@Q      001  0001 0024   2247* 2260* 2267* 2268* 2408* 2413* 2417* 2418* 2423* 2427* 2454* 2515*
                         2538* 2543* 2630* 2651  2657* 2675* 2738* 2742* 3040  3181* 3194* 3360*
                         3371* 3389* 3396* 3402  3467  3536* 3543* 3883  3972  3974  4215* 4277*
                         4569* 4575  4584  4888  5006* 5037* 5190* 5211* 5230  5355
@REGL   001  0002 0012
@RETRN  001  0080 0153   0154
@RLDWN  001  004F 0159
@RTRNC  001  0080 0161
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  03/02/22  PAGE 102
@SBLN   001  0005 0170
@SBLNL  001  0002 0184
@SCTSZ  001  0100 0100
@SDFLN  001  0007 0090
@SDF0   001  0000 0166
@SDF1   001  0001 0167
@SDF2   001  0002 0168
@SDF3   001  0003 0169
@SECCY  001  0030 0086
@SIST   001  0001 0181
@SLASH  001  0061 0067   4517  4533
@SLAST  001  0002 0183
@SMIDL  001  0003 0182
@SNULL  001  0080 0173
@SONLY  001  0000 0180
@STEXT  001  0007 0172
@STYPE  001  0006 0171
@TBCNT  001  0000 0160
@TBLEF  001  0010 0155   0157
@TBLIX  001  0011 0157
@UCB    001  0087 0039   2260  2267  2268  2413  2417  2418  2454  2515  2657  3181  3371  3378
                         3396  3467  3973  4215  4277  5006  5211  5333  5344
@UPARW  001  005A 0078
@VADDR  001  0002 0141
@VENTA  001  0056 0113
@VMDDV  001  00FE 0114
@VMFD1  001  0000 0109
@VMFD2  001  0001 0110
@VMRS3  001  0002 0112
@VMTRL  001  0001 0111
@VOLID  001  0006 0091   2755  2802  4185  4187  4191  4294  4304  4309  4319  4489* 4546  4546
                         4744  4818
@VQ     001  0001 0025   4263* 4280
@WSFIT  001  0500 0101   2289  2299
@WSTBL  001  0503 0102   2281  2299
@XR     001  0002 0014   2221* 2232  2239  2250  2254  2263  2271  2275* 2367* 2376* 2377  2380
                         2383  2399* 2414  2422  2433  2434  2447  2448  2457  2457* 2459  2467
                         2468  2472* 2473  2496* 2498  2500  2501  2507  2524* 2525  2527  2536
                         2558* 2564* 2586  2606* 2609  2628* 2629  2631  2631* 2640* 2641  2643
                         2644  2653  2654  2654  2656  2670  2670* 2705  2706  2707  2708  2709
                         2710  2711  2712  2713  2716  2814  2821  2821* 2822  2830  2830* 2831
                         2831  2838* 2839  2840  3171  3182* 3187  3190  3195  3196  3196* 3199
                         3201  3201* 3212  3214* 3215  3216  3218* 3225  3341  3348* 3349  3350*
                         3351  3352  3353  3354  3364  3368  3372  3372* 3373  3375  3379  3379*
                         3390  3392  3393  3431  3438  3438* 3447  3448  3448* 3449  3458  3469
                         3469* 3470  3474  3484* 3578  3677  3679* 3680  3680* 3681  3682  3683*
                         3684  3685  3686  3689  3691  3691* 3696  3696* 3699  3700* 3701  3716
                         3727  3728  3733* 3875  3876* 3877  3892  3905  3907  3911  3913  3917
                         3919  3923  3925  3932  3934  3947  3949  3954  3955  3956  3959  3980
                         3985* 4021  4176  4185* 4187  4191  4191* 4224* 4293* 4296  4296* 4297
                         4300  4302  4305  4307  4310  4310* 4311  4311* 4312  4315  4497  4500
                         4500* 4501  4504  4504* 4517  4528  4528* 4533  4542  4542* 4547  4551
                         4553  4556  4567  4574* 4764  4783  4785  4787  4790  4792  4801* 4826
                         4827  4827* 4838  4999  5013* 5014  5015  5015* 5017  5019  5019* 5027
                         5028  5033  5042* 5192  5195  5195* 5196  5198  5198* 5199  5201  5206
                         5221  5337  5340  5340* 5341  5343  5346  5346* 5347  5349  5351
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  03/02/22  PAGE 103
@ZERO   001  0000 0062   2222  2232  2239  2245  2250  2254  2263  2271  2297  2315  2412  2414
                         2415  2416  2422  2433  2436  2447  2473  2539  2639  2641  2654* 2667
                         2693  2736  2742  2831  2831* 3009  3197  3361  3495  3498  3537  3561
                         3562  3693  3890  3891  3892  3905  3911  3917  3923  3932  3947  4187
                         4212  4297  4305  4307  4312  4315  4319  4497  4501  4517  4533  4547
                         4549  4551  4553  4567  4773  4783  4785  4787  4790  4792  4826  4835
                         4838  4852  5031  5189  5192  5196  5199  5201  5206  5206*
DL2C01  002  12F6 3052   2992  2994  3002
DL2C05  002  12F8 3053   2998
DL2C48  001  12F2 3050   3000  3004
DL2DPL  006  12FE 3058   2999*
DL2END  001  1301 3063   3065
DL2E01  001  0001 2982   3000  3002  3004  3008  3020  3028
DL2E02  001  0002 2983   3013  3016  3034
DL2E18  001  0018 2984   3014
DL2E60  001  0060 2985   3029
DL2E7C  001  007C 2987   3026
DL2ICS  001  1268 2988   2361  2578  2618  2636  2661  2698  2771  2834  2842  3191  3703  3718
                         3729  3957  5008  5034
DL2K18  002  12F4 3051   3017
DL2K60  002  12EF 3048   3035
DL2K80  002  12F1 3049   3016  3034
DL2LST  001  12F9 3057   3000* 3002* 3004* 3008  3009* 3013* 3016* 3020  3026* 3034* 3037* 3042
                         3059
DL2PHY  001  12FB 3059
DL2RAD  002  1300 3062   3013  3954* 5007*
DL2SAD  005  1280 3060   3020* 3027* 3028* 3029  3035* 3037
DL2SEC  005  1289 3061   3008* 3014  3017* 3018  3018* 3019  3019* 3028
DL2SWH  003  12DE 3040
DL2TSD  001  0083 2986   3027
DL2000  001  126C 2990   2980  2991
DL2001  005  127C 2997   2993* 3060
DL2002  005  1285 2999   2997* 2998* 3061
DL2005  004  128A 3000   3003
DL2006  004  1298 3004   3001
DL2008  004  12B5 3018   3015
DL2010  003  12CB 3029
DL2100  004  12D9 3037   3030
DL2110  003  12DD 3039   3040
DL2900  004  12E6 3043   2989* 3039
DL2910  004  12EA 3044   2995*
DL4ICS  001  0987 2198   2401  2684
KSA#LN  002  0CBE 2884   2434* 2709
KSABCT  001  0CC3 2887   2639* 2644* 2645  2652* 2653  2655  2656  2667* 2693  2695
KSABSZ  003  0FFF 2651   2652
KSABTS  001  001C 2196   2713
KSAC00  002  0D19 2297   2275  2480  2784
KSAC01  002  0D1D 2300   2426  2441  2442  2452  2458  2461  2543  2630  2632  2723  2727  2737
                         2738  2805
KSAC04  002  1203 2860   2781
KSAC48  001  0D1A 2298   2541  2545
KSAC61  001  0D1E 2301   2423  2424
KSAC64  001  0D1F 2302   2427  2428
KSADBC  001  0CB1 2874   2422* 2424* 2428*
KSADB1  001  0CB6 2878   2448* 2449  2458* 2459  2467*
KSADIR  050  0CB0 2873
```

```
KSAEND  001  1268 2862   2890
KSAERL  001  0002 2195   2296  2473* 2670
KSAE01  001  0001 2185   2421  2433
KSAE02  001  0002 2189   2441  2442  2464  2484  2507  2525  2643
KSAFBA  002  0D0F 2291   2399
KSAFDB  001  0D1B 2299   2683
KSAFEL  001  0004 2191   2457
KSAFFD  001  000C 2190   2448  2459  2467  2468  2629*
KSAFLG  001  0004 2197   2631
KSAFLN  001  0D0D 2290   2412* 2421* 2426* 2435  2638  2707  2720  2723* 2727*
KSAFPL  001  0D0A 2288   2402  2634* 2637
KSAFSZ  002  0D0C 2289   2635*
KSAHDR  001  0C66 2870   2243  2258  2711
KSAHED  025  0C7E 2871   2237* 2238  2238* 2498*
KSAIOB  001  1301 3065   2283  2287  2736*
KSAIOR  001  0D3A 2337   2561
KSALT2  001  0D28 2313   2362  2474  2617* 2619  2772  2782  2832* 2835
KSALT3  001  0D22 2305   2310  2724* 2726
KSALT4  001  0D10 2292   2575* 2576* 2577* 2579
KSAL3A  002  0D27 2311   2720*
KSAL4A  002  0D12 2293
KSAMXT  001  000C 2192   2286  2645
KSANBF  001  0700 2193
KSANCC  002  0CBB 2883
KSANDA  002  0CBC 2882   2536* 2537* 2586* 2883
KSANSA  002  0CC0 2885
KSAOFA  002  0CBA 2881   2538  2539* 2540* 2541* 2545* 2546*
KSAOFS  002  0CC2 2886
KSAOVL  001  0D40 2346   2370
KSAPAC  001  0D34 2329   2567
KSAPCH  100  1267 2861
KSAPLX  001  0D2E 2320
KSARDC  001  0CB7 2879   2449* 2468*
KSARDT  001  0802 2194   2414* 2415* 2416* 2464* 2640
KSASAV  001  0CC4 2888
KSASCT  001  0CB8 2880   2450* 2461* 2464  2469*
KSASMN  002  0D21 2303   2594
KSASPU  001  0D33 2326   2556
KSASP2  001  0CC5 2889
KSAST@  004  112C 2762
KSASTA  002  1201 2859   2847
KSASTR  004  110B 2750   2384
KSAST0  004  1140 2771   2810
KSAST1  004  114C 2778   2756  2758
KSAST2  005  11A0 2814   2798
KSAST3  006  11EC 2847   2785
KSAST9  006  1159 2782   2779
KSATBL  002  0D17 2296   2466
KSATFL  002  0CB3 2875   2433* 2435* 2436* 2441* 2442* 2475  2484  2507  2525  2537  2540  2706
KSAUDB  001  11FA 2852   2839* 2840* 2843
KSAUDE  001  0C7F 2872   2606
KSAVE   004  0C54 2220   2201
KSAVEN  001  0C66 2228   2184  2220  2373  2869
KSAWRK  002  0CB5 2876   2447* 2452* 2627* 2632*
KSA001  004  0CA5 2248   2273
KSA002  003  0CB6 2253   2247* 2267*
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  03/02/22  PAGE 105
KSA003  004  0CBC 2255   2264
KSA005  004  0D46 2356   2240  2251  2265  2272
KSA006  003  0CE0 2267   2244
KSA007  004  0D91 2384   2375
KSA008  004  0D95 2385   2751  2849
KSA024  003  0CF7 2275   2277  2278  2358  2379  2382  2392  2398  2754  2804  2809  2817  2825
KSA025  004  0CFA 2276   2230  2233  2246  2249  2253  2256  2259  2262  2270  2486
KSA030  003  0DAE 2399   2386
KSA032  004  0DC8 2410
KSA035  003  0DEE 2421   2411
KSA040  004  0E01 2426   2429
KSA045  004  0E12 2433   2419  2425
KSA047  004  0E33 2442   2440
KSA050  003  0E37 2443   2413* 2438
KSA060  004  0E3A 2447
KSA070  004  0E49 2452   2462  2470
KSA080  003  0E57 2457   2453
KSA090  005  0E6D 2464   2455  2460  2466* 2472
KSA100  003  0E72 2465   2454*
KSA110  003  0E82 2469
KSA130  004  0E89 2472   2465
KSA140  006  0E90 2474   2443
KSA144  005  0EAE 2484   2526
KSA145  004  0E9B 2476
KSA150  004  0EB9 2489   2481
KSA155  004  0EC3 2496   2490
KSA156  003  0EC7 2497   2260* 2268*
KSA157  004  0ECE 2500   2497
KSA16#  004  0F25 2545   2542
KSA16@  004  0F01 2536   2600
KSA163  004  0F15 2541   2544
KSA164  004  0F29 2546   2538* 2543* 2547  2587
KSA165  004  0F33 2548   2589
KSA166  004  0EEE 2524   2483
KSA170  006  0F3E 2556   2487  2790  2846
KSA175  004  0F5E 2573   2516  2531  2551
KSA180  005  0F78 2586   2502  2508
KSA2DA  002  0D06 2285   2500* 2527* 2588* 2607* 2635  2638* 2708
KSA2IA  002  0D09 2287   2668  2741
KSA2PL  001  0D04 2284   2407* 2662  2666  2666* 2695* 2699  2739
KSA200  005  0F8C 2594   2509
KSA210  004  0F99 2606   2491
KSA221  003  0FAE 2616   2515* 2581
KSA222  003  0FBB 2626   2417* 2616
KSA224  003  0FBE 2627   2423* 2427*
KSA225  003  0FC5 2629   2630* 2633
KSA229  003  0FE9 2639   2626
KSA230  003  0FF0 2641   2671  2676
KSA235  003  0FFE 2645   2408* 2651
KSA260  004  1020 2659   2646
KSA275  003  1039 2669   2657* 2675* 2681
KSA276  004  1043 2675   2669
KSA280  004  104B 2680   2647
KSA290  004  1053 2682   2659  2680
KSA295  004  1065 2687   2682*
KSA300  003  1069 2693   2642
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  03/02/22  PAGE 106
KSA320  004  107D 2700   2694
KSA4IA  002  0D03 2283   2668*
KSA4PL  001  0CFE 2280   2643* 2655* 2683* 2685  2686  2686*
KSA500  004  1081 2704   2573  2608
KSA555  004  10B5 2718   2715
KSA557  004  10CE 2725   2722
KSA560  004  10D8 2728   2704* 2719
KSA610  004  10DC 2732   2660  2697
KSA620  003  10E0 2733   2418*
KSA630  004  10E9 2736   2737* 2740  2741*
KSA640  003  10F7 2739   2738* 2742*
KSA690  004  1107 2743   2732* 2733
SALBSE  001  195F 4782   4758  4761
SALCNT  001  19FB 4879   4773* 4811* 4814  4818  4835
SALCT6  001  0006 4744
SALCT8  001  0008 4742
SALERR  003  1975 4888   4800
SALFST  001  0001 4876   4797  4809
SALIDR  001  19FA 4869   4755* 4794  4797  4809* 4812  4840  4852*
SALND0  004  19F2 4860   4760*
SALND2  004  19F6 4861   4762*
SALPHR  001  19FE 4883   4516  4532  4546  4885  4886  4887
SALPHS  002  1A09 4885   4774
SALPH6  001  1943 4759   4544
SALPH8  001  193F 4753   4514  4530
SALPR6  001  1A06 4887   4772*
SALPR7  001  1A07 4886   4771* 4772
SAL001  002  19FD 4882   4811  4825
SAL008  001  0080 4873   4755  4794  4812  4840
SAL100  003  1951 4771
SAL200  003  195F 4783   4828
SAL250  003  1974 4791   4888
SAL350  003  198D 4800   4816  4820  4844
SAL375  004  1990 4801   4556* 4574  4764*
SAL400  003  1997 4809   4784  4786  4788  4793
SAL425  004  199A 4811   4795  4799
SAL450  003  19B1 4818   4813
SAL500  004  19BB 4825   4817
SAL525  005  19BF 4826   4774* 4825*
SAL750  003  19CA 4835   4791
SAL755  004  19CD 4836
SAL760  003  19E8 4844   4839  4842
SAL775  004  19EB 4845   4837
SAL800  003  19EF 4852   4802
SCACNT  002  1C28 5361   4549  5351* 5352*
SCACOF  001  0087 5333   4513
SCACOM  001  0001 5332   2234  4557
SCAINC  001  0001 5331   5340  5346
SCAMMA  003  1C05 5355   2234* 4513* 4557*
SCANIT  001  1BE8 5335   2229  2261  2269  4529  4543  4558  4845
SCASVE  002  1C26 5360   5337* 5352
SCASV1  001  1C25 5359
SCA100  003  1BF7 5340   5342
SCA200  003  1BFA 5341   5339
SCA250  003  1C04 5344   5355
SCA300  003  1C07 5346   5348
```

```
SCA400  004  1C17 5351   5344
SCA500  004  1C21 5354   5336* 5350
SCSCNT  001  1BE2 5234   2245  5189* 5203* 5209
SCSERR  002  1BE7 5237   5218
SCSFRC  001  00FF 5232   5221
SCSLNG  004  1BBE 5230   2235*
SCSPL1  002  1BE4 5235   5187  5203
SCSPL2  001  1BE5 5236   5186
SCSQUO  001  007D 5231   5192  5196  5199
SCSTRG  001  1B6D 5183   2242  2257
SCS005  004  1B8B 5191   5188*
SCS006  003  1B95 5195   5212
SCS010  003  1BA7 5201   5197
SCS020  003  1BB3 5205   5190* 5211*
SCS025  004  1BBD 5209   5230
SCS029  004  1BC8 5212   5205  5210
SCS030  001  1BCC 5217   5193  5202
SCS040  003  1BD7 5221   5200
SCS050  004  1BDA 5225   5184* 5220
SCS051  004  1BDE 5226   5185* 5186*
SFIAST  001  005C 4010   3888
SFIBSE  003  168D 4017   3872  3873
SFICTR  001  1761 3994   3890* 3899  3902  3908* 3914* 3920* 3926* 3969
SFIDPL  001  1762 3997   3958
SFIEFE  001  00FE 4013   3908  3969
SFIEFF  001  00FF 4014   3996
SFIEND  001  176A 4018
SFIERR  003  0CF7 2277   3950  4009
SFIETD  001  0006 4019   2807  3975
SFIEXT  004  1760 3990   3874*
SFIE02  001  0002 4011   3920
SFIE03  001  0003 4012   3902  3926
SFIE06  001  0006 4015   3905  3911  3917  3923
SFIE07  001  0007 4016   3907  3913  3919  3925
SFIFND  003  173B 3974   2763*
SFINDF  001  164F 3870   2291  2311  2356  2628  2766  2806
SFINTR  001  1769 4002   2765* 2805* 2807  3975  3978  4003
SFIONE  001  176A 4005   3977
SFIRDA  002  1764 3998   3956*
SFISBR  004  175C 3988   3871*
SFISTR  003  1738 3972   2762*
SFISXR  004  1758 3986   3875*
SFITTC  001  1768 4001   3891* 3977* 3978
SFIVOL  004  1670 3883   2764*
SFI050  004  166F 3882   3883
SFI100  004  1676 3888   3881
SFI200  003  168D 3899   3971  3979  4017
SFI210  003  169C 3905   3924
SFI220  003  16AD 3911   3900
SFI230  003  16BE 3917   3901  3912
SFI240  003  16CF 3923   3903  3918
SFI320  003  16E0 3932   3889
SFI340  005  16E6 3934   3893
SFI350  004  16EB 3938   3884  3909  3915  3921  3927
SFI500  003  1700 3947   3879
SFI505  003  1706 3949   3933
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  03/02/22  PAGE 108
SFI510  005  170D 3954   3948
SFI520  004  1726 3963   3943
SFI540  003  1731 3969   3940
SFI542  003  1737 3971   3972
SFI543  003  173A 3973   3974
SFI545  003  174E 3980   3906  3973  3976
SFI550  004  1755 3985   3942  3965  3970  3986
SFI560  004  1759 3987   3988
SFI570  004  175D 3989   3990
SGECNT  001  1A93 5057   5014* 5020* 5031
SGEC01  002  1A95 5058   5020
SGEDPL  001  1A8B 5049   5009  5013  5033* 5035  5038*
SGEEND  001  1A96 5060
SGERAD  002  1A92 5056   5038
SGETDB  001  1A0A 4996   3938  4995  4998
SGE050  003  1A20 5005   5006* 5037*
SGE055  003  1A38 5013   5005
SGE060  005  1A42 5017   5021
SGE070  004  1A58 5027   5018
SGE080  004  1A6E 5033
SGE900  004  1A7F 5041   4997* 5030  5032
SGE901  004  1A83 5042   4999*
SGE902  004  1A87 5043   5000*
SMAEND  001  1F6D 5096
SMALES  001  1B3E 5070   5071
SMBFDA  001  1B58 5076   2557  3907* 3913* 3919* 3925* 3934* 3955* 4189* 4212  4329* 5007  5077
SMDAAD  001  1B6C 5086   2577  3216* 5092
SMFNAM  001  1B54 5074   2236* 2710  3199  4516* 5075
SMFUDA  001  1B68 5084   3959* 5028* 5085
SMIND1  001  1B3E 5071   2357  2374  2482  2489  2550  2757  2797  2824  3213* 3226* 3479* 3894*
                         3939  3941  3964  3981* 5004* 5022* 5029  5072  5097
SMNDBA  001  1B6A 5085   2474* 2782* 3350  3568  5086
SMNDEA  001  1B5E 5079   2303  2480  2501  2588  2607  2784  2859  3713  3716  5080
SMNETD  001  1B62 5081   2547* 2587* 2594* 2822* 2847* 3348  5082
SMNSCT  001  1B60 5080   2222* 2227* 2475* 2781* 2789  2815  3711* 5081
SMNULT  001  1B5C 5078   2484  2789  2814* 2815  5079
SMPDB1  001  1B6D 5092   5053  5093  5094  5100  5101
SMPEAD  001  1B66 5083   5027* 5084
SMPIBS  001  1B6D 5093
SMPSWD  001  1B4C 5073   2397  2750  2752  3878  3888  4487* 4488  4488* 4499* 4503* 4519  4532*
                         5017  5074
SMUDBA  001  1B5A 5077   2576  2838  3215* 3683  5078
SMUDB1  001  1B6D 5094   2295  3235  4000  5095
SMUDB2  001  1D6D 5095   3236  5096
SMUDEA  001  1B56 5075   2376  2496  2524  3212* 3225* 5076
SMUPEN  001  1B64 5082   2609* 3679  5083
SMVOID  001  1B44 5072   2755  2802  3880  4187  4319  4489* 4546* 5073
SM1FNE  001  0080 5087   2374  2482  2489  2757  2797  3213  3226  3964  3981
SM1NPD  001  0040 5088
SM1PDS  001  0010 5090   3941  5029
SM1PNF  001  0008 5091   2357  3894  3939  5004  5022
SM1STN  001  0020 5089   2550  2824  3479
SRCACT  002  13A4 3234   3176* 3182  3206  3207* 3214
SRCBA1  002  13A6 3235   3174
SRCBA2  002  13A8 3236   3175
SRCBFR  002  13B1 3243   3189*
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  03/02/22  PAGE 109
SRCBF1  002  13A0 3232   3174* 3176  3205* 3207
SRCBF2  002  13A2 3233   3175* 3189  3205  3206*
SRCCNT  001  13A9 3237   3195* 3197  3202*
SRCC01  002  13AB 3238   3187  3202
SRCDAD  002  13AE 3241   3190*
SRCDPL  001  13AC 3239   3192
SRCGET  001  13AC 3240
SRCHFN  001  1301 3167   3963
SRCSCT  001  13AF 3242
SRC010  004  1305 3170   3169  3170
SRC020  004  131F 3178   3208
SRC030  004  1343 3195   3188
SRC035  005  1350 3199   3203
SRC040  004  1374 3212   3200
SRC050  003  137C 3214   3227
SRC055  003  1362 3204   3181* 3194* 3198
SRC060  004  1394 3225   3204
SRC900  004  1388 3217   3168*
SRC910  004  138C 3218   3171*
SRC920  004  1390 3219   3172*
STOCLN  002  14CF 3494   3412
STOC48  002  14D5 3497   3540  3545
STOENA  002  14E4 3507
STOENC  002  14DE 3504   3353* 3561* 3563  3563* 3564  3565* 3566* 3567* 3568* 3569* 3570  3571*
                         3572
STOENL  001  0004 3503   3349  3384  3414  3458
STORAM  002  14DC 3502   3450* 3523  3524  3573*
STORCS  006  14EB 3534   3390  3418  3465  3538* 3540* 3545*
STORCW  004  14CB 3491   3418  3465
STORC0  006  14EA 3533   3536  3537* 3546*
STORC1  002  14DA 3500   3376  3436  3440  3471  3521  3543  3569
STORDR  002  14D1 3495
STOREC  004  14CD 3492   3474
STOREL  002  14D3 3496   3450  3571
STORET  004  148F 3453   3395
STORE1  001  0001 3339   3364
STORE2  001  0002 3501   3540  3545  3567
STORHL  002  14E2 3506   3567
STORIN  001  13B2 3336   2548  2595  2823  2848
STORNM  002  14D7 3498   3573
STORPA  005  146B 3432   3375* 3412* 3413  3424
STORWA  004  14CB 3493   3393
STORWC  001  14D8 3499   3354* 3358  3361  3376* 3440* 3521*
STORWE  004  14CD 3490   3349* 3368  3373  3384  3392  3425  3470  3491  3492  3493  3507  3578
STORWK  006  14ED 3532   3384* 3414* 3458* 3533  3534  3538
STOR10  003  13FA 3371   3359
STOR14  003  13FD 3372   3362
STOR15  004  1400 3373   3380
STOR20  004  141B 3384   3374
STOR30  003  143B 3401   3337  3340  3391  3402
STOR31  003  143C 3402   3346* 3378*
STOR35  004  143E 3407   3347* 3395* 3419
STOR38  004  1442 3412   3401
STOR39  005  144A 3414   3413*
STOR40  005  1468 3431   3420  3432
STOR45  005  1470 3437   3351*
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  03/02/22  PAGE 110
STOR46  005  146D 3436   3437
STOR47  004  1490 3458   3377
STOR48  004  14B3 3474   3466
STOR50  003  14E5 3516   3451  3574
STOR51  006  14E8 3518   3519  3520  3525
STOR52  006  14EB 3519   3447* 3523* 3572*
STOR53  006  14ED 3520   3449* 3524* 3532  3570*
STOR55  004  14EE 3521
STOR60  003  1500 3531   3385  3416  3460
STOR65  004  150E 3540
STOR66  004  1519 3545   3541
STOR70  004  1524 3558   3360* 3371* 3467  3505
STOR80  004  14BA 3479   3468  3558
STOR90  004  14BE 3483   3338* 3369  3426  3441  3452  3472  3475
STOR95  004  14C2 3484   3341*
STOR99  004  14C6 3485   3342*
STOSAV  004  1523 3560   3562* 3564* 3565  3566
STO047  004  148C 3452   3453
STO048  005  14AB 3471   3352* 3579
STO064  003  151D 3546   3536* 3543*
STO067  004  1520 3547   3516* 3522  3531* 3560
STO39A  003  1459 3420   3389* 3396*
STO39B  005  1460 3425   3424*
STO70A  002  14E0 3505   3347
STUCLU  001  160F 3749   3711
STUCNT  001  1610 3756   3689* 3693  3695*
STUC00  002  160E 3748   3713
STUC01  001  164E 3759   3695  3701
STUDPL  001  1607 3744   3728* 3730
STUERR  003  0CF7 2278   3714
STUE01  001  0001 3671   3726
STUE02  001  0002 3672   3711
STUFID  001  1566 3670   2610
STUHDR  001  1610 3754   3726* 3727* 3747  3756
STULST  001  1601 3740   3684* 3685* 3700  3704  3719
STUNHD  012  161B 3755   3721  3721*
STUNNT  050  164D 3757   3722*
STU000  004  156A 3675   3674  3675
STU010  003  1599 3693   3697
STU020  005  15A9 3699   3681* 3694
STU040  005  15BE 3711   3687
STU050  005  15D0 3716
STU060  005  15DF 3722   3682*
STU900  004  15F5 3732   3673* 3706
STU910  004  15F9 3733   3677*
STU920  004  15FD 3734   3676*
SUFBSE  001  18A9 4512   4474  4477
SUFFER  001  1876 4475   2248
SUFND0  004  1937 4586   4476* 4576
SUFND2  004  193B 4587   4478*
SUF100  004  18A9 4513   4498  4502
SUF200  003  18E1 4542   4520
SUF400  003  18EC 4545   4575
SUF600  003  18F5 4547   4518  4534
SUF625  003  18F8 4548
SUF650  004  190E 4556   4550
```

```
                                 CROSS REFERENCE
SYMBOL  LEN VALUE DEFN   REFERENCES                           VER 15, MOD 00  03/02/22  PAGE 111
SUF680  004  1926 4570   4554  4569* 4571
SUF750  003  192E 4575   4515  4531  4545  4559
SUF780  003  1931 4576   4584
SUF800  003  1934 4584   4548  4552  4568
SURCHN  001  0A50 2199   2476  2783  3712
SVOBSE  001  177D 4186   4173  4175
SVOBUF  001  1B6D 5101   4276* 4318
SVOCT1  001  17C4 4235   4192* 4236
SVOCT2  001  17C5 4238   4190* 4201  4239
SVOEND  001  00FF 4164   4276* 4318
SVOERR  003  0CF7 4009   4226
SVOINP  001  0100 4163   4276  4318
SVOLID  001  176B 4172   4185  4294
SVOLN1  001  0001 4160   4190  4192
SVOONE  001  17C6 4241   4190  4192
SVO001  001  00F1 4161   4300
SVO002  001  00F2 4162   4302
SVO100  005  177D 4187   4193
SVO200  003  178E 4191   4188
SVO260  004  17A5 4212   4331
SVO270  004  17B0 4215   4203  4251  4321
SVO274  004  17B4 4223   4174* 4213
SVO276  004  17B8 4224   4176*
SVO280  004  17BC 4226   4215*
SVO290  004  17C0 4227   4177*
SVO300  004  17C7 4249   4204
SVO310  004  17CB 4250
SVO315  003  17CF 4251
SVO320  001  17D2 4259   4303  4308  4316
SVO330  001  17E4 4273   4277*
SVO333  004  17F0 4278   4275
SVO335  004  17FA 4280   4263*
SVO350  004  1802 4282   4283
SVO360  003  1818 4296   4298
SVO400  003  1832 4305   4301
SVO440  003  1842 4310   4306
SVO445  003  1845 4311   4313
SVO450  005  185C 4319

TOTAL STATEMENTS IN ERROR IN THIS ASSEMBLY =     0

OL105 I  THE CODE LENGTH OF #KSAVE IS  7209 DECIMAL.
OL103 I  TOTAL NUMBER OF LIBRARY SECTORS REQUIRED IS   22
         NAME-#KSAVE,PACK-R1R1R1,UNIT-R1,RETAIN-P,LIBRARY-R,CATEGORY-000
```

```
        OVERLAY LINKAGE EDITOR CORE USAGE MAP    03/02/22



 START    CATEGORY  NAME AND       CODE LENGTH
ADDRESS               ENTRY    HEXADECIMAL DECIMAL

 0C00         0     #KSAVE          1C29    7209


OL100 I  THE TOTAL CORE USED BY #KSAVE IS  7209 DECIMAL.
OL101 I  THE START CONTROL ADDRESS OF THIS MODULE IS 0C00.
OL104 I  TOTAL NUMBER OF LIBRARY SECTORS REQUIRED IS   29
         NAME-#KSAVE,PACK-R1R1R1,UNIT-R1,RETAIN-P,LIBRARY-O
```