```
000.001                 1  DEBUG   EQU     1               DON'T ASSEMBLE FOR DEBUG
                        3  ***     HDOSOVL2        - MOUNT/DISMOUNT OVERLAY
                        4  *
                        5  *
                        6  *       G. Chandler             79.02.sc
                        7  *
                        8  *       COPYRIGHT HEATH CO., 1979
                        9  *


                       11  ***     HDOSOVL2 HANDLES REQUESTS FOR MOUNT AND DISMOUNT.
                       12  *       THIS OVERLAY MUST BE NO LARGER THAN THE SYSTEM I/O
                       13  *       HANDLER OVERLAY.


                       15  **      SYSTEM SYMBOLS
                       16
000.000                17          XTEXT   ASCII

                       19X **      ASCII CHARACTER EQUIVALENCES.
                       20X
000.015                21X CR      EQU     13              CARRIAGE RETURN
000.012                22X LF      EQU     10              LINE FEED
000.200                23X NULL    EQU     200Q            PAD CHARACTER
000.000                24X NUL2    EQU     0
000.007                25X BELL    EQU     7               BELL CHARACTER
000.177                26X RUBOUT  EQU     177Q
000.010                27X BKSP    EQU     10Q             CTL-H
000.026                28X C.SYN   EQU     26Q             SYNC
000.002                29X C.STX   EQU     2               STX
000.047                30X QUOTE   EQU     47Q
000.011                31X TAB     EQU     11Q
000.033                32X ESC     EQU     33Q
000.012                33X NL      EQU     12Q             NEW LINE (HDOS SYSTEMS)
000.212                34X ENL     EQU     NL+200Q         NL + END-OF-LINE-FLAG
000.014                35X FF      EQU     14Q             FORM FEED
000.001                36X CTLA    EQU     01Q             CTL-A
000.002                37X CTLB    EQU     02Q             CTL-B
000.003                38X CTLC    EQU     03Q             CTL-C
000.004                39X CTLD    EQU     04Q             CTL-D
000.017                40X CTLO    EQU     17Q             CTL-O
000.020                41X CTLP    EQU     20Q             CTL-P
000.021                42X CTLQ    EQU     21Q             CTL-Q
000.023                43X CTLS    EQU     23Q             CTL-S
000.032                44X CTLZ    EQU     32Q             CTL-Z
000.000                45          XTEXT   HTR
```

```
                        48X **      HTR - PAM/8 EQUIVALENCES.
                        49X *
                        50X *       THIS DECK CONTAINS SYMBOLIC DEFINITIONS USED TO
                        51X *       MAKE USE OF THE PAM/8 CODE AND CONTROL BYTES.

                        53X **      IO PORTS
                        54X
000.360                 55X IP.PAD  EQU     360Q            PAD INPUT PORT
000.360                 56X OP.CTL  EQU     360Q            CONTROL OUTPUT PORT
000.360                 57X OP.DIG  EQU     360Q            DIGIT SELECT OUTPUT PORT
000.361                 58X OP.SEG  EQU     361Q            SEGMENT SELECT OUTPUT PORT

                        60X **      FRONT PANEL CONTROL BITS.
                        61X
000.020                 62X CB.SSI  EQU     00010000B       SINGLE STEP INTERRUPT
000.040                 63X CB.MTL  EQU     00100000B       MONITOR LIGHT
000.100                 64X CB.CLI  EQU     01000000B       CLOCK INTERRUPT ENABLE
000.200                 65X CB.SPK  EQU     10000000B       SPEAKER ENABLE

                        67X **      MONITOR MODE FLAGS.
                        68X
000.000                 69X DM.MR   EQU     0               MEMORY READ
000.001                 70X DM.MW   EQU     1               MEMORY WRITE
000.002                 71X DM.RR   EQU     2               REGISTER READ
000.003                 72X DM.RW   EQU     3               REGISTER WRITE

                        74X **      USER OPTION BITS.
                        75X *
                        76X *       THESE BITS ARE SET IN CELL .MFLAG.
                        77X
000.200                 78X UO.HLT  EQU     10000000B       DISABLE HALT PROCESSING
000.100                 79X UO.NFR  EQU     CB.CLI          NO REFRESH OF FRONT PANEL
000.002                 80X UO.DDU  EQU     00000010B       DISABLE DISPLAY UPDATE
000.001                 81X UO.CLK  EQU     00000001B       ALLOW PRIVATE INTERRUPT PROCESSING

                        83X **      MONITOR IDENTIFICATION FLAGS
                        84X *
                        85X *       THESE BYTES IDENTIFY THE ROM MONITOR.
                        86X *       THEY ARE THE VARIOUS VALUES OF LOCATION .IDENT
                        87X
000.021                 88X M.PAM8  EQU     021Q            'LXI' INSTRUCTION AT 000.000 IN PAM-8
000.303                 89X M.FOX   EQU     303Q            'JMP' INSTRUCTION AT 000.000 IN FOX ROM
```

```
                        91X **      ROUTINE ENTRY POINTS.
                        92X *
                        93X
000.000                 94X .IDENT  EQU     0000A       IDENTIFICATION LOCATION
000.053                 95X .DLY    EQU     0053A       DELAY
001.267                 96X .LOAD   EQU     1267A       TAPE LOAD
001.374                 97X .DUMP   EQU     1374A       TAPE DUMP
002.136                 98X .ALARM  EQU     2136A       ALARM ROUTINE
002.140                 99X .HORN   EQU     2140A       HORN
002.172                100X .CTC    EQU     2172A       CHECK TAPE CHECKSUM
002.205                101X .TPERR  EQU     2205A       TAPE ERROR ROUTINE
002.264                102X .PCHL   EQU     2264A       PCHL INSTRUCTION
002.265                103X .SRS    EQU     2265A       SCAN RECORD START
002.325                104X .RNP    EQU     2325A       READ NEXT PAIR
002.331                105X .RNB    EQU     2331A       READ NEXT BYTE
002.347                106X .CRC    EQU     2347A       CRC-16 CALCULATOR
003.017                107X .WNP    EQU     3017A       WRITE NEXT PAIR
003.024                108X .WNB    EQU     3024A       WRITE NEXT BYTE
003.122                109X .DOD    EQU     3122A       DECODE FOR OCTAL DISPLAY
003.260                110X .RCK    EQU     3260A       READ CONSOLE KEYSET
003.356                111X .DODA   EQU     3356A       SEGMENT CODE TABLE


                        113X **     RAM CELLS USED BY H8MTR.
                        114X *
                        115X
040.000                 116X .START  EQU     40000A      START DUMP ADDRESS
040.002                 117X .IOWRK  EQU     40002A      IN OR OUT INSTRUCTION
040.005                 118X .REGI   EQU     40005A      DISPLAYED REGISTER INDEX
040.006                 119X .DSPROT EQU     40006A      PERIOD FLAG BYTE
040.007                 120X .DSPMOD EQU     40007A      DISPLAY MODE
040.010                 121X .MFLAG  EQU     40010A      USER OPTION BYTE
040.011                 122X .CTLFLG EQU     40011A      PANEL CONTROL BYTE
040.013                 123X .ALEDS  EQU     40013A      ABUSS LEDS
040.021                 124X .DLEDS  EQU     40021A      DBUSS LEDS
040.024                 125X .ABUSS  EQU     40024A      ABUSS REGISTER
040.027                 126X .CRCSUM EQU     40027A      CRCSUM WORD
040.031                 127X .TPERRX EQU     40031A      TAPE ERROR EXIT VECTOR
040.033                 128X .TICCNT EQU     40033A      CLOCK TICK COUNTER
040.035                 129X .REGPTR EQU     40035A      REGISTER POINTER
040.037                 130X .UIVEC  EQU     40037A      USER INTERRUPT VECTORS
000.000                 131          XTEXT   BOODEF


                        133X **     BOODEF - SPECIAL BOOT-HDOS INTERFACE DEFINTIONS.
                        134X
047.000                 135X SB.ORG  EQU     47000A      ORG FOR LOAD OF INITIAL HDOS.SAV
014.000                 136X SB.OVMX EQU     14000A      SIZE OF HOLD AREA FOR SWAPPED USER CODE
                        137X *                           (=MAX SIZE OF HDOSOVL.SYS)
000.000                 138          XTEXT   HDSROM
```

```
                    140X **     HDOS H17 ROM ENTRY POINTS.
031.253             141X        ORG     31253A
031.253             142X DWRITE EQU     *
031.253             143X        DS      31256A-31253A
031.256             144X DREAD  EQU     *
031.256             145X        DS      31275A-31256A
031.275             146X S.READ EQU     *
031.275             147X        DS      31321A-31266A
031.330             148X S.WRITE EQU    *
031.330             149X        DS      31325A-31311A
031.344             150X ERR.FNO EQU    *
031.344             151X        DS      31331A-31325A
031.350             152X ERR.ILR EQU    *
031.350             153X        DS      31335A-31331A
031.354             154X CFF    EQU     *
031.354             155X        DS      31363A-31335A
032.002             156X DCA    EQU     *
032.002             157X        DS      32114A-31363A
032.133             158X FFB    EQU     *
032.133             159X        DS      32166A-32114A
032.205             160X FFL    EQU     *
032.205             161X        DS      32204A-32166A
                    162X *LDD   EQU     *
032.223             163X        DS      32372A-32204A+1
033.012             164X LDO    EQU     *
033.012             165X        DS      33135A-33002A
033.145             166X PDI    EQU     *
033.145             167X        DS      33154A-33124A
033.175             168X REL.   EQU     *
033.175             169X        DS      33156A-33154A
033.177             170X REL    EQU     *
033.177             171X        DS      33212A-33156A
033.233             172X TFE    EQU     *
033.233             173X        DS      33232A-33206A
033.257             174X RUC    EQU     *
033.257             175         XTEXT   FILDEF


                    177X **     FILDEF - FILE TYPE DEFINITIONS.
                    178X *
                    179X *      DB      377Q,FT.XXX
                    180X
                    181X
000.000             182X FT.ABS EQU     0           ABSOLUTE BINARY
000.001             183X FT.PIC EQU     1           POSITION INDEPENDANT CODE
000.002             184X FT.REL EQU     2           RELOCATABLE CODE
000.003             185X FT.BAC EQU     3           COMPILED BASIC CODE
033.257             186         XTEXT   H17DEF
```

```
                    188X **     H17 CONTROL INFORMATION.
                    189X
000.177             190X DP.DC  EQU     07FH            DISK CONTROL PORT
                    191X
000.001             192X DF.HD  EQU     00000001B       HOLE DETECT
000.002             193X DF.TO  EQU     00000010B       TRACK 0 DETECT
000.004             194X DF.WP  EQU     00000100B       WRITE PROTECT
000.010             195X DF.SD  EQU     00001000B       SYNC DETECT
                    196X
000.001             197X DF.WG  EQU     00000001B       WRITE GATE ENABLE
000.002             198X DF.DS0 EQU     00000010B       DRIVE SELECT 0
000.004             199X DF.DS1 EQU     00000100B       DRIVE SELECT 1
000.010             200X DF.DS2 EQU     00001000B       DRIVE SELECT 2
000.020             201X DF.MO  EQU     00010000B       MOTOR ON (BOTH DRIVES)
000.040             202X DF.DI  EQU     00100000B       DIRECTION (0=OUT)
000.100             203X DF.ST  EQU     01000000B       STEP COMMAND (ACTIVE HIGH)
000.200             204X DF.WR  EQU     10000000B       WRITE ENABLE RAM
                    205X
                    206X
                    207X
                    208X **     DISK UART PORTS AND CONTROL FLAGS.
                    209X
000.174             210X UP.DP  EQU     07CH            DATA PORT
000.175             211X UP.FC  EQU     07DH            FILL CHARACTER
000.175             212X UP.ST  EQU     07DH            STATUS FLAGS
000.176             213X UP.SC  EQU     07EH            SYN CHARACTER (OUTPUT)
000.176             214X UP.SR  EQU     07EH            SYNC RESET (INPUT)
                    215X
000.001             216X UF.RDA EQU     00000001B       RECEIVE DATA AVAILABLE
000.002             217X UF.ROR EQU     00000010B       RECEIVER OVERRUN
000.004             218X UF.RPE EQU     00000100B       RECEIVER PARITY ERROR
000.100             219X UF.FCT EQU     01000000B       FILL CHAR TRANSMITTED
000.200             220X UF.TBM EQU     10000000B       TRANSMITTER BUFFER EMPTY
                    221X
                    222X
                    223X
                    224X **     CHARACTER DEFINITIONS.
                    225X
000.375             226X C.DSYN EQU     0FDH            PREFIX SYNC CHARACTER
033.257             227         XTEXT   HOSDEF


                    229X **     HOSDEF - DEFINE HOS PARAMETER.
                    230X *
                    231X
                    232X
000.026             233X VERS   EQU     1*16+6          VERSION 1.6
                    234X
000.377             235X SYSCALL EQU    377Q            SYSCALL INSTRUCTION
                    236X
                    237X
000.000             238X        ORG     0
                    239X
                    240X *      RESIDENT FUNCTIONS
                    241X
```

```
000.000  242X .EXIT   DS    1    EXIT (MUST BE FIRST)
000.001  243X .SCIN   DS    1    SCIN
000.002  244X .SCOUT  DS    1    SCOUT
000.003  245X .PRINT  DS    1    PRINT
000.004  246X .READ   DS    1    READ
000.005  247X .WRITE  DS    1    WRITE
000.006  248X .CONSL  DS    1    SET/CLEAR CONSOLE OPTIONS
000.007  249X .CLRCO  DS    1    CLEAR CONSOLE BUFFER
000.010  250X .LOADO  DS    1    LOAD AN OVERLAY
000.011  251X .VERS   DS    1    RETURN HDOS VERSION NUMBER
000.012  252X .SYSRES DS    1    PRECEDING FUNCTIONS ARE RESIDENT
         253X
         254X
         255X *       *HDOSOVL0.SYS*  FUNCTIONS
         256X
000.040  257X        ORG   40A
         258X
000.040  259X .LINK   DS    1    LINK  (MUST BE FIRST)
000.041  260X .CTLC   DS    1    CTL-C
000.042  261X .OPENR  DS    1    OPENR
000.043  262X .OPENW  DS    1    OPENW
000.044  263X .OPENU  DS    1    OPENU
000.045  264X .OPENC  DS    1    OPENC
000.046  265X .CLOSE  DS    1    CLOSE
000.047  266X .POSIT  DS    1    POSITION
000.050  267X .DELET  DS    1    DELETE
000.051  268X .RENAM  DS    1    RENAME
000.052  269X .SETTP  DS    1    SETTOP
000.053  270X .DECODE DS    1    NAME DECODE
000.054  271X .NAME   DS    1    GET FILE NAME FROM CHANNEL
000.055  272X .CLEAR  DS    1    CLEAR CHAN
000.056  273X .CLEARA DS    1    CLEAR ALL CHANS
000.057  274X .ERROR  DS    1    LOOKUP ERROR
000.060  275X .CHFLG  DS    1    CHANGE FLAGS
000.061  276X .DISMT  DS    1    FLAG SYSTEM DISK DISMOUNTED
000.062  277X .LOADD  DS    1    LOAD DEVICE DRIVER
         278X
         279X
         280X *       *HDOSOVL1.SYS*  FUNCTIONS
         281X
000.200  282X        ORG   200Q
         283X
000.200  284X .MOUNT  DS    1    MOUNT  (MUST BE FIRST)
000.201  285X .DMOUN  DS    1    DISMOUNT
000.202  286X .MONMS  DS    1    MOUNT/NO MESSAGE
000.203  287X .DMNMS  DS    1    DISMOUNT/NO MESSAGE
000.204  288X .RESET  DS    1    RESET = DISMOUNT/MOUNT OF UNIT
000.205  289        XTEXT OVLDEF
```

```
                        291X **      OVERLAY TABLE ENTRYS.
                        292X
000.000                 293X         ORG     0
                        294X
000.000                 295X OVL.COD DS      2          FIRST SECTOR OF OVERLAY CODE
000.002                 296X OVL.SIZ DS      2          OVERLAY SIZE
000.004                 297X OVL.ENT DS      2          OVERLAY ENTRY POINT
000.006                 298X OVL.FLB DS      1          OVERLAY FLAG BYTE
000.007                 299X         DS      1          DUMMY BYTE TO ROUND TABLE SIZE UP TO 8
000.010                 300X OVL.ENS EQU     *          OVERLAY ENTRY SIZE
                        301X
                        302X *       OVERLAY INDICES
                        303X
000.000                 304X         ORG     0
                        305X
000.000                 306X OVL0    DS      1
000.001                 307X OVL1    DS      1
000.002                 308          XTEXT   DEVDEF



                        310X **      DEVICE TABLE ENTRYS.
                        311X
000.000                 312X         ORG     0
                        313X
000.000                 314X DEV.NAM DS      2          DEVICE NAME
000.000                 315X DV.EL   EQU     00000000B  END OF DEVICE LIST FLAG
000.001                 316X DV.NU   EQU     00000001B  DEVICE ENTRY NOT IN USE
                        317X
000.002                 318X DEV.RES DS      1          DRIVER RESIDENSE CODE
000.001                 319X DR.IM   EQU     00000001B  DRIVER IN MEMORY
000.002                 320X DR.PR   EQU     00000010B  DRIVER PERMINANTLY RESIDENT
                        321X
000.003                 322X DEV.JMP DS      1          JMP TO PROCESSOR
000.004                 323X DEV.DDA DS      2          DRIVER ADDRESS
000.006                 324X DEV.FLG DS      1          FLAG BYTE
000.001                 325X DT.DD   EQU     00000001B  DIRECTORY DEVICE
000.002                 326X DT.CR   EQU     00000010B  CAPABLE OF READ OPERATION
000.004                 327X DT.CW   EQU     00000100B  CAPABLE OF WRITE OPERATION
                        328X
000.007                 329X DEV.SPG DS      1          SECTORS PER GROUP THIS DEVICE
000.010                 330X DEV.MUM DS      1          MOUNTED UNIT MASK
000.011                 331X DEV.MNU DS      1          MAXIMUM NUMBER OF UNITS
000.012                 332X DEV.UNT DS      2          ADDRESS OF UNIT SPECIFIC DATA TABLE
                        333X
000.014                 334X DEV.DVL DS      2          DRIVER BYTE LENGTH
000.016                 335X DEV.DVG DS      1          DRIVER ROUTINE GROUP ADDRESS
                        336X
000.017                 337X DEVELEN EQU     *          DEVICE TABLE ENTRY LENGTH
```

```
                    339X **     UNIT SPECIFIC DEVICE DATA TABLE ENTRIES
                    340X
000.000             341X        ORG     0
                    342X
000.000             343X UNT.FLG DS     1           UNIT SPECIFIC  *DEV.FLG*
000.001             344X UNT.GRT DS     2           ADDRESS OF GROUP RESERVATION TABLE (IF DT.DD)
000.003             345X UNT.GTS DS     2           GRT SECTOR NUMBER
000.005             346X UNT.DIS DS     2           DIRECTORY FIRST SECTOR NUMBER
                    347X
000.007             348X UNT.SIZ EQU    *           SIZE OF UNIT SPECIFIC DATA TABLE PER UNIT
000.007             349         XTEXT   DIRDEF


                    351X **     DIRECTORY ENTRY FORMAT.
                    352X
000.000             353X        ORG     0
                    354X
                    355X
000.377             356X DF.EMP  EQU     377Q        FLAGS ENTRY EMPTY
000.376             357X DF.CLR  EQU     376Q        FLAGS ENTRY EMPTY, REST OF DIR ALSO CLEAR
                    358X
000.000             359X DIR.NAM DS     8           NAME
000.010             360X DIR.EXT DS     3           EXTENSION
000.013             361X DIR.PRO DS     1           PROJECT
000.014             362X DIR.VER DS     1           VERSION
000.015             363X DIRIDL  EQU    *           FILE IDENTIFICATION LENGTH
                    364X
000.015             365X DIR.CLU DS     1           CLUSTER FACTOR
000.016             366X DIR.FLG DS     1           FLAGS
000.017             367X        DS      1           RESERVED
000.020             368X DIR.FGN DS     1           FIRST GROUP NUMBER
000.021             369X DIR.LGN DS     1           LAST GROUP NUMBER
000.022             370X DIR.LSI DS     1           LAST SECTOR INDEX (IN LAST GROUP)
000.023             371X DIR.CRD DS     2           CREATION DATE
000.025             372X DIR.ALD DS     2           LAST ALTERATION DATE
                    373X
000.027             374X DIRELEN EQU    *           DIRECTORY ENTRY LENGTH
000.027             375         XTEXT   DISDEF


                    377X **     DIRECTORY BLOCK FORMAT.
                    378X
000.000             379X        ORG     0
                    380X
000.000             381X DIS.ENT EQU    *           FIRST ENTRY ADDRESS
000.000             382X        DS      22*DIRELEN  22 DIRECTORY ENTRYS PER BLOCK
001.372             383X        DS      1           0 BYTE = END OF ENTRYS IN THIS BLOCK
                    384X
001.373             385X        ORG     512-5       AT END OF BLOCK
001.373             386X DIS.ENL DS     1           LENGTH OF EACH ENTRY (=DIRELEN)
001.374             387X DIS.SEC DS     2           BLOCK # OF THIS BLOCK,
001.376             388X DIS.LNK DS     2           BLOCK # OF NEXT BLOCK, =0 IF THIS IS LAST
002.000             389         XTEXT   IOCDEF
```

```
                    391X **     I/O CHANNEL DEFINITIONS.
                    392X
000.000             393X        ORG     0
                    394X
000.000             395X IOC.LNK DS     2           ADDRESS OF NEXT CHANNEL, =0 IF LAST
000.002             396X IOC.DDA DS     2           THREAD JUMP TO DEVICE DRIVER (VIA DEV TABLE)
                    397X
000.004             398X IOC.FLG DS     1           FILE TYPE FLAGS
000.001             399X FT.DD   EQU    00000001B   =1 IF DIRECTORY DEVICE
000.002             400X FT.OR   EQU    00000010B   =1 IF OPEN FOR READ
000.004             401X FT.OW   EQU    00000100B   =1 IF OPEN FOR WRITE
000.010             402X FT.OU   EQU    00001000B   =1 IF OPEN FOR UPDATE
000.003             403X IOC.SQL EQU    *-IOC.DDA   LENGTH OF INFO FOR SEQUENTIAL FILE (FROM IOC)
                    404X
000.005             405X IOC.GRT DS     2           ADDRESS OF GROUP RESERVATION TABLE
000.007             406X IOC.SPG DS     1           SECTORS PER GROUP, THIS DEVICE
000.010             407X IOC.CGN DS     1           CURRENT GROUP NUMBER
000.011             408X IOC.CSI DS     1           CURRENT SECTOR INDEX (IN CURRENT GROUP)
000.012             409X IOC.LGN DS     1           LAST GROUP NUMBER
000.013             410X IOC.LSI DS     1           LAST SECTOR INDEX (IN LAST GROUP)
000.010             411X IOC.DRL EQU    *-IOC.FLG   LENGTH OF INFO NORMALLY COPIED BACK TO
                    412X *                          THE CHANNEL TABLE
000.014             413X IOC.DTA DS     2           DEVICE TABLE ADDRESS FOR THIS DEVICE
000.016             414X IOC.DES DS     2           SECTOR NUMBER OF DIRECTORY ENTRY
000.020             415X IOC.DEV DS     2           DEVICE CODE
000.022             416X IOC.UNI DS     1           UNIT NUMBER (0-9)
000.021             417X IOC.DIL EQU    *-IOC.DDA   LENGTH OF INFO FOR DIRECTORY FILE (FROM IOC)
                    418X
000.023             419X IOC.DIR DS     DIRELEN     DIRECTORY ENTRY
                    420X
000.052             421X IOCELEN EQU    *           IOC ENTRY LENGTH
                    422X
000.001             423X IOCCTD  EQU    1           INDEX OF USER CHANNEL #0 IN CHANTAB (FIRST = 0)
000.052             424          XTEXT  DDDEF


                    426X **     DEVICE DRIVER COMMUNICATION FLAGS.
                    427X *
                    428X
000.000             429X        ORG     0
                    430X
000.000             431X DC.REA  DS     1           READ
000.001             432X DC.WRI  DS     1           WRITE
000.002             433X DC.RER  DS     1           READ REGARDLESS
000.003             434X DC.OPR  DS     1           OPEN FOR READ
000.004             435X DC.OPW  DS     1           OPEN FOR WRITE
000.005             436X DC.OPU  DS     1           OPEN FOR UPDATE
000.006             437X DC.CLO  DS     1           CLOSE
000.007             438X DC.ABT  DS     1           ABORT
000.010             439X DC.MOU  DS     1           MOUNT DEVICE
000.011             440X DC.LOD  DS     1           LOAD DEVICE DRIVER
000.012             441X DC.MAX  DS     1           MAXIMUM ENTRY INDEX
000.013             442          XTEXT  ECDEF
```

```
                    444X **     ERROR CODE DEFINITIONS.
                    445X
000.000             446X        ORG     0
000.000             447X        DS      1           NO ERROR #0
000.001             448X EC.EOF DS      1           END OF FILE
000.002             449X EC.EOM DS      1           END OF MEDIA
000.003             450X EC.ILC DS      1           ILLEGAL SYSCALL CODE
000.004             451X EC.CNA DS      1           CHANNEL NOT AVAILABLE
000.005             452X EC.DNS DS      1           DEVICE NOT SUITABLE
000.006             453X EC.IDN DS      1           ILLEGAL DEVICE NAME
000.007             454X EC.IFN DS      1           ILLEGAL FILE NAME
000.010             455X EC.NRD DS      1           NO ROOM FOR DEVICE DRIVER
000.011             456X EC.FNO DS      1           CHANNEL NOT OPEN
000.012             457X EC.ILR DS      1           ILLEGAL REQUEST
000.013             458X EC.FUC DS      1           FILE USAGE CONFLICT
000.014             459X EC.FNF DS      1           FILE NAME NOT FOUND
000.015             460X EC.UND DS      1           UNKNOWN DEVICE
000.016             461X EC.ICN DS      1           ILLEGAL CHANNEL NUMBER
000.017             462X EC.DIF DS      1           DIRECTORY FULL
000.020             463X EC.IFC DS      1           ILLEGAL FILE CONTENTS
000.021             464X EC.NEM DS      1           NOT ENOUGH MEMORY
000.022             465X EC.RF  DS      1           READ FAILURE
000.023             466X EC.WF  DS      1           WRITE FAILURE
000.024             467X EC.WPV DS      1           WRITE PROTECTION VIOLATION
000.025             468X EC.WP  DS      1           DISK WRITE PROTECTED
000.026             469X EC.FAP DS      1           FILE ALREADY PRESENT
000.027             470X EC.DDA DS      1           DEVICE DRIVER ABORT
000.030             471X EC.FL  DS      1           FILE LOCKED
000.031             472X EC.FAO DS      1           FILE ALREADY OPEN
000.032             473X EC.IS  DS      1           ILLEGAL SWITCH
000.033             474X EC.UUN DS      1           UNKNOWN UNIT NUMBER
000.034             475X EC.FNR DS      1           FILE NAME REQUIRED
000.035             476X EC.DIW DS      1           DEVICE IS NOT WRITABLE (OR WRITE LOCKED)
000.036             477X EC.UNA DS      1           UNIT NOT AVAILABLE
000.037             478X EC.ILV DS      1           ILLEGAL VALUE
000.040             479X EC.ILO DS      1           ILLEGAL OPTION
000.041             480X EC.VPM DS      1           VOLUME PRESENTLY MOUNTED ON DEVICE
000.042             481X EC.NVM DS      1           NO VOLUME PRESENTLY MOUNTED
000.043             482X EC.FOD DS      1           FILE OPEN ON DEVICE
000.044             483X EC.NPM DS      1           NO PROVISIONS MADE FOR REMOUNTING MORE DISKS
000.045             484X EC.DNI DS      1           DISK NOT INITIALIZED
000.046             485X EC.DNR DS      1           DISK IS NOT READABLE
000.047             486X EC.DSC DS      1           DISK STRUCTURE IS CORRUPT
000.050             487X EC.NCV DS      1           NOT CORRECT VERSION OF HDOS
000.051             488X EC.NOS DS      1           NO OPERATING SYSTEM MOUNTED
000.052             489X EC.IOI DS      1           ILLEGAL OVERLAY INDEX
000.053             490X EC.OTL DS      1           OVERLAY TO LARGE
000.054             491         XTEXT   DDFDEF
```

```
                    493X **     DIRECTORY DEVICE FORMAT DEFINITION.
                    494X *
                    495X
                    496X
000.002             497X HDS.SPG EQU    2           2 SECTORS PER GROUP REQUIRED FOR NOW
                    498X
000.000             499X         ORG    0
000.000             500X DDF.BOO DS     9           2K BOOT PROGRAM
000.011             501X DDF.BOL EQU    *           LENGTH OF BOOT
000.011             502X DDF.LAB DS     1           LABEL SECTOR
000.012             503X DDF.RGT DS     2           RESERVED GROUP TABLE
000.014             504X DDF.USR DS     0           BEGINNING OF OPEN SPACE
000.014             505          XTEXT  LABDEF


                    507X **     DISK LABEL SECTOR FORMATS.
                    508X
000.000             509X         ORG    0
000.000             510X LAB.SER DS     1           SERIAL NUMBER OF VOLUME
000.001             511X LAB.IND DS     2           INITIALIZATION DATE
000.003             512X LAB.DIS DS     2           SECTOR NUMBER OF 1ST DIRECTORY SECTOR
000.005             513X LAB.GRT DS     2           INDEX OF GRT SECTOR
000.007             514X LAB.SPG DS     1           SECTORS PER GROUP
                    515X
000.000             516X LAB.DAT EQU    0           DATA VOLUME ONLY
000.001             517X LAB.SYS EQU    1           SYSTEM VOLUME
000.002             518X LAB.NOD EQU    2           => LAB.NOD MEANS VOLUME HAS NO DIRECTORY
                    519X
000.010             520X LAB.VLT DS     1           VOLUME TYPE
000.011             521X LAB.VER DS     1           VERSION OF INIT17 THAT INITED DISK
000.012             522X         DS     7           UNUSED
000.021             523X LAB.LAB DS     60          LABEL
000.074             524X LAB.LBL EQU    *-LAB.LAB   LABEL LENGTH
000.115             525          XTEXT  ABSDEF


                    527X **     ABS FORMAT EQUIVALENCES.
                    528X
000.000             529X         ORG    0
                    530X
000.000             531X ABS.ID  DS     1           377Q = BINARY FILE FLAG
000.001             532X         DS     1           FILE TYPE (FT.ABS)
000.002             533X ABS.LDA DS     2           LOAD ADDRESS
000.004             534X ABS.LEN DS     2           LENGTH OF ENTIRE RECORD
000.006             535X ABS.ENT DS     2           ENTRY POINT
                    536X
000.010             537X ABS.COD DS     0           CODE STARTS HERE
000.010             538          XTEXT  PICDEF
```

```
                        540X **      PIC FORMAT EQUIVALENCES.
                        541X
000.000                 542X         ORG     0
                        543X
000.000                 544X PIC.ID  DS      1           377Q = BINARY FILE FLAG
000.001                 545X         DS      1           FILE TYPE (FT.PIC)
000.002                 546X PIC.LEN DS      2           LENGTH OF ENTIRE RECORD
000.004                 547X PIC.PTR DS      2           INDEX OF START OF PIC TABLE
                        548X
000.006                 549X PIC.COD DS      0           CODE STARTS HERE
000.006                 550          XTEXT   DIFDEF


                        552X **      DIRECTORY FILE FLAGS.
                        553X
000.200                 554X DIF.SYS EQU     10000000B           SYSTEM FILE
000.100                 555X DIF.LOC EQU     01000000B           LOCKED FOR CHANGE
000.040                 556X DIF.WP  EQU     00100000B           WRITE PROTECTED
000.020                 557X DIF.CNT EQU     00010000B           CONTIGUOUS FILE
                        558X
000.006                 559          XTEXT   NAMDEF


                        561X **      SYSTEM FILE NAME CONVENTIONS
                        562X *
                        563X *       RGT     .SYS        RESERVED GROUP TABLE (1 SECTOR)
                        564X *       GRT     .SYS        GROUP RESERVATION TABLE (1 SECTOR)
                        565X *       DIRECT  .SYS        DIRECTORY
                        566X *       HOS     .SYS        SYSTEM IMAGE PROGRAM FOR SYSTEM
                        567X
000.006                 568          XTEXT   MTRDEF


                        570X **      HDOS MONITOR PRIVATE RAM AREA DEFINITIONS.
                        571X
000.000                 572X         ORG     0
000.000                 573X M.SYSM  DS      1           SYSCALL ITTERATION COUNT
000.001                 574X M.SALO  DS      1           STAND-ALONE FLAG
000.002                 575X M.CSLC  DS      1           LINES IN CONSOLE BUFFER
000.003                 576X M.CPRE  DS      1           CONSOLE PREVIOUS CHARACTER
000.004                 577X M.CRUB  DS      1           CONSOLE RUBOUT FLAG
000.005                 578X M.CINT  DS      1           CONSOLE INTERRUPT FLAG
000.006                 579X M.CIN   DS      2           CONSOLE CB IN POINTER
000.010                 580X M.COUT  DS      2           CONSOLE CB OUT POINTER
000.012                 581X M.CFWA  DS      2           CONSOLE CB FWA POINTER
000.014                 582X M.CLWA  DS      2           CONSOLE CB LWA POINTER
000.016                 583X M.CDLY  DS      1           CONSOLE PAD CHARACTER COUNT
000.017                 584X M.CPCA  DS      2           ADDRESS OF CHARACTER BEING PADDED
```

```
000.021                 587           XTEXT   HOSEQU

                        589X **       HDOS SYSTEM EQUIVALENCES.
                        590X *
                        591X
024.000                 592X S.GRT0   EQU     24000A          SYSTEM AREA FOR  GRT0
025.000                 593X S.GRT1   EQU     25000A          SYSTEM AREA FOR  GRT1
026.000                 594X S.GRT2   EQU     26000A          SYSTEM AREA FOR  GRT2
                        595X
030.000                 596X ROMBOOT  EQU     30000A          ROM BOOT ENTRY
                        597X
040.100                 598X          ORG     40100A          FREE SPACE FROM PAM-8
                        599X
040.100                 600X          DS      8               JUMP TO SYSTEM EXIT
040.110                 601X D.CON    DS      16              DISK CONSTANTS
040.130                 602X SYDD     EQU     *               SYSTEM DISK ENTRY POINT
040.130                 603X D.VEC    DS      24*3            SYSTEM ROM ENTRY VECTORS
040.240                 604X D.RAM    DS      31              SYSTEM ROM WORK AREA
040.277                 605X S.VAL    DS      36              SYSTEM VALUES
040.343                 606X S.INT    DS      115             SYSTEM INTERNAL WORK AREAS
041.126                 607X          DS      16
041.146                 608X S.SOVR   DS      2               STACK OVERFLOW WARNING
041.150                 609X          DS      42200A-*        SYSTEM STACK
001.032                 610X STACKL   EQU     *-S.SOVR        STACK SIZE
                        611X
042.200                 612X STACK    EQU     *               LWA+1 SYSTEM STACK
042.200                 613X USERFWA  EQU     *               USER FWA
                        614
042.200                 615           XTEXT   EDVEC

                        617X **       JMP VECTORS FOR ROM CODE
                        618X *
                        619X *        SEE DISK ROM FOR ADDRESSES
                        620X *
                        621X *        HOSEQU MUST BE ALTERED WHEN THIS TABLE IS ALTERED.
                        622X
040.130                 623X          ORG     D.VEC
                        624X
040.130                 625X D.SYDD   DS      3               JMP  R.SYDD (MUST BE FIRST)
040.133                 626X D.MOUNT  DS      3               JMP  R.MOUNT
040.136                 627X D.XOK    DS      3               JMP  R.XOK
040.141                 628X D.ABORT  DS      3               JMP  R.ABORT
040.144                 629X D.XIT    DS      3               JMP  R.XIT
040.147                 630X D.READ   DS      3               JMP  R.READ
040.152                 631X D.READR  DS      3               JMP  R.READR
040.155                 632X D.WRITE  DS      3               JMP  R.WRITE
040.160                 633X D.CDE    DS      3               JMP  R.CDE
040.163                 634X D.DTS    DS      3               JMP  R.DTS
040.166                 635X D.SDT    DS      3               JMP  R.SDT
040.171                 636X D.MAI    DS      3               JMP  R.MAI
```

```
040.174          637X D.MAO   DS      3               JMP  R.MAO
040.177          638X D.LPS   DS      3               JMP  R.LPS
040.202          639X D.RDB   DS      3               JMP  R.RDB
040.205          640X D.SDP   DS      3               JMP  R.SDP
040.210          641X D.STS   DS      3               JMP  R.STS
040.213          642X D.STZ   DS      3               JMP  R.STZ
040.216          643X D.UDLY  DS      3               JMP  R.UDLY
040.221          644X D.WSC   DS      3               JMP  R.WSC
040.224          645X D.WSP   DS      3               JMP  R.WSP
040.227          646X D.WNB   DS      3               JMP  R.WNB
040.232          647X D.ERRT  DS      3               JMP  R.ERRT
040.235          648X D.DLY   DS      3               JMP  R.DLY
                 649
040.240          650          XTEXT   ESVAL


                 652X **      S.VAL - SYSTEM VALUE DEFINTIONS.
                 653X *
                 654X *       THESE VALUES ARE SET AND MAINTAINED BY THE SYSTEM.
                 655X *
                 656X *       THE DECK HOSEQU MUST BE MODIFIED WHEN THIS IS MODIFIED.
                 657X
                 658X
040.277          659X         ORG     S.VAL
                 660X
040.277          661X S.DATE  DS      9               SYSTEM DATE (IN ASCII)
040.310          662X S.DATC  DS      2               CODED DATE
040.312          663X S.TIME  DS      4               TIME FROM MIDNIGHT (IN TICS)
040.316          664X S.HIMEM DS      2               HARDWARE HIGH MEMORY ADRESS+1
                 665X
040.320          666X S.SYSM  DS      2               FWA RESIDENT SYSTEM
                 667X
040.322          668X S.USRM  DS      2               LWA USER MEMORY
                 669X
040.324          670X S.OMAX  DS      2               MAX OVERLAY SIZE FOR SYSTEM
                 671X
                 672X
                 673X **      THE FOLLOWING FIVE CELLS SHOULD BE MODIFIED/READ ONLY VIA THE .CONSL SYSCALL
                 674X
000.200          675X CSL.ECH EQU     10000000B       SUPPRESS ECHO
000.002          676X CSL.WRP EQU     00000010B       WRAP LINES AT WIDTH
000.001          677X CSL.CHR EQU     00000001B       OPERATE IN CHARACTER MODE
                 678X
000.000          679X I.CSLMD EQU     0               S.CSLMD IS FIRST BYTE
040.326          680X S.CSLMD DS      1               CONSOLE MODE
                 681X
000.200          682X CTP.BKS EQU     10000000B       TERMINAL PROCESSES BACKSPACES
000.040          683X CTP.MLI EQU     00100000B       MAP LOWER CASE TO UPPER ON INPUT
000.020          684X CTP.MLO EQU     00010000B       MAP LOWER CASE TO UPPER ON OUTPUT
000.010          685X CTP.2SB EQU     00001000B       TERMINAL NEEDS TWO STOP BITS
000.002          686X CTP.BKM EQU     00000010B       MAP BKSP (UPON INPUT) TO RUBOUT
000.001          687X CTP.TAB EQU     00000001B       TERMINAL SUPPORTS TAB CHARACTERS
                 688X
000.001          689X I.CONTY EQU     1               S.CONTY IS 2ND BYTE
```

```
000.000  690X          ERRNZ   *-S.CSLMD-I.CONTY
040.327  691X S.CONTY  DS      1               CONSOLE TYPE FLAGS
000.002  692X I.CUSOR  EQU     2               S.CUSOR IS 3RD BYTE
000.000  693X          ERRNZ   *-S.CSLMD-I.CUSOR
040.330  694X S.CUSOR  DS      1               CURRENT CURSOR POSITION
000.003  695X I.CONWI  EQU     3               S.CONWI IS 4TH BYTE
000.000  696X          ERRNZ   *-S.CSLMD-I.CONWI
040.331  697X S.CONWI  DS      1               CONSOLE WIDTH
         698X
000.001  699X CO.FLG   EQU     00000001B       CTL-O FLAG
000.200  700X CS.FLG   EQU     10000000B       CTL-S FLAG
         701X
000.004  702X I.CONFL  EQU     4               S.CONFL IS 5TH BYTE
000.000  703X          ERRNZ   *-S.CSLMD-I.CONFL
040.332  704X S.CONFL  DS      1               CONSOLE FLAGS
         705X
040.333  706X S.CAADR  DS      2               ADDRESS FOR ABORT PROCESSING (>256 IF VALID)
040.335  707X S.CCTAB  DS      6               ADDR FOR CTL-A, CTL-B, CTL-C PROCESSING
         708
040.343  709           XTEXT   ESINT
```

```
         711X **       S.INT - SYSTEM INTERNAL WORKAREA DEFINITIONS.
         712X *
         713X *        THESE CELLS ARE REFERENCED BY OVERLAYS AND MAIN CODE, AND
         714X *        MUST THEREFORE RESIDE IN FIXED LOW MEMORY.
         715X
         716X
040.343  717X          ORG     S.INT
         718X
         719X **       CONSOLE STATUS FLAGS
         720X
040.343  721X S.CDB    DS      1               CONSOLE DESCRIPTOR BYTE
000.000  722X CDB.H85  EQU     00000000B
000.001  723X CDB.H84  EQU     00000001B       =0 IF H8-5, =1 IF H8-4
040.344  724X S.BAUD   DS      2               [0-14]  H8-4 BAUD RATE, =0 IF H8-5
         725X *                                [15]    =1 IF BAUD RATE => 2 STOP BITS
         726X
         727X **       TABLE ADDRESS WORDS
         728X
040.346  729X S.DLINK  DS      2               ADDRESS OF DATA IN HDOS CODE
040.350  730X S.OFWA   DS      2               FWA  OVERLAY  TABLE
040.352  731X S.CFWA   DS      2               FWA  CHANNEL  TABLE
040.354  732X S.DFWA   DS      2               FWA  DEVICE   TABLE
040.356  733X S.RFWA   DS      2               FWA  RESIDENT HDOS CODE
         734X
         735X **       DEVICE DRIVER DELAYED LOAD FLAGS
         736X
040.360  737X S.DDLDA  DS      2               DRIVER LOAD ADDRESS (HIGH BYTE=0 IF NO LOAD PENDING)
040.362  738X S.DDLEN  DS      2               CODE LENGTH IN BYTES
040.364  739X S.DDGRP  DS      1               GROUP NUMBER FOR DRIVER
040.365  740X          DS      1               HOLD PLACE
         741X *S.DDSEC         DS      2               SECTOR NUMBER FOR DRIVER ( * OBSOLETE ! * )
040.366  742X S.DDDTA  DS      2               DEVICE'S ADDRESS IN DEVLST +DEV.RES
```

```
040.370                 743X S.DUOPC DS      1              OPEN OPCODE PENDEDING
                        744X
                        745X **      OVERLAY MANAGEMENT FLAGS
                        746X
000.001                 747X OVL.IN  EQU     00000001B      IN MEMORY
000.002                 748X OVL.RES EQU     00000010B      PERMINANTLY RESIDENT
000.014                 749X OVL.NUM EQU     00001100B      OVERLAY NUMBER MASK
000.200                 750X OVL.UCS EQU     10000000B      USER CODE SWAPPED FOR OVERLAY
                        751X
040.371                 752X S.OVLFL DS      1              OVERLAY FLAG
040.372                 753X S.UCSF  DS      2              FWA SWAPPED USER CODE
040.374                 754X S.UCSL  DS      2              LENGTH SWAPPED USER CODE
040.376                 755X S.OVLS  DS      2              SIZE OF OVERLAY CODE
041.000                 756X S.OVLE  DS      2              ENTRY POINT OF OVERLAY CODE
                        757X
041.002                 758X S.SSN   DS      2              SWAP AREA SECTOR NUMBER
041.004                 759X S.OSN   DS      2              OVERLAY SECTOR NUMBER
                        760X
                        761X *       SYSCALL PROCESSING WORK AREAS
                        762X
041.006                 763X S.CACC  DS      1              (ACC) UPON SYSCALL
041.007                 764X S.CODE  DS      1              SYSCALL INDEX IN PROGRESS
                        765X
                        766X *       JUMPS TO ROUTINES IN RESIDENT HDOS CODE
                        767X
041.010                 768X S.JUMPS DS      0              START OF DUMP VECTORS
041.010                 769X S.SDD   DS      3              JUMP TO STAND-IN DEVICE DRIVER
041.013                 770X S.FASER DS      3              JUMP TO FATSERR (FATAL SYSTEM ERROR)
041.016                 771X S.DIREA DS      3              JUMP TO DIREAD (DISK FILE READ)
041.021                 772X S.FCI   DS      3              JUMP TO FCI (FETCH CHANNEL INFO)
041.024                 773X S.SCI   DS      3              JUMP TO SCI (STORE CHANNEL INFO)
041.027                 774X S.GUP   DS      3              JUMP TO GUP (GET UNIT POINTER)
                        775X
041.032                 776X S.MOUNT DS      1              <>0 IF THE SYSTEM DISK IS MOUNTED
041.033                 777X S.DCS   DS      1              DEFAULT CLUSTER SIZE-1
                        778X
041.034                 779X S.BOOTF DS      1              BOOT FLAGS
000.001                 780X BOOT.P  EQU     00000001B      EXECUTE PROLOGUE UPON BOOTUP
                        781X
                        782X *       STACK VALUE SAVED FOR OVERLAY SYSCALLS
                        783X
041.035                 784X S.OVSTK DS      2              VALUE OF SP UPON SYSCALLS USING OVERLAY
                        785X
041.037                 786X         DS      1              RESERVED
```

```
                        788X **      ACTIVE I/O AREA.
                        789X *
                        790X *       THE AIO.XXX AREA CONTAINS INFORMATION ABOUT THE I/O OPERATION
                        791X *       CURRENTLY BEING PERFORMED. THE INFORMATION IS OBTAINED FROM
                        792X *       THE CHANNEL TABLE, AND WILL BE RESTORED THERE WHEN DONE.
                        793X *
                        794X *       NORMALLY, THE AIO.XXX INFORMATION WOULD BE OBTAINED DIRECTLY
                        795X *       FROM VARIOUS SYSTEM TABLES VIA POINTER REGISTERS. SINCE THE
```

```
                        796X *      8080 HAS NO GOOD INDEXED ADDRESSING, THE DATA IS MANUALLY
                        797X *      COPIED INTO THE AIO.XXX CELLS BEFORE PROCESSING, AND
                        798X *      BACKDATED AFTER PROCESSING.
                        799X
041.040                 800X AIO.VEC DS     3           JUMP INSTRUCTION
041.041                 801X AIO.DDA EQU    *-2         DEVICE DRIVER ADDRESS
041.043                 802X AIO.FLG DS     1           FLAG BYTE
041.044                 803X AIO.GRT DS     2           ADDRESS OF GROUP RESERV TABLE
041.046                 804X AIO.SPG DS     1           SECTORS PER GROUP
041.047                 805X AIO.CGN DS     1           CURRENT GROUP NUMBER
041.050                 806X AIO.CSI DS     1           CURRENT SECTOR INDEX
041.051                 807X AIO.LGN DS     1           LAST GROUP NUMBER
041.052                 808X AIO.LSI DS     1           LAST SECTOR INDEX
041.053                 809X AIO.DTA DS     2           DEVICE TABLE ADDRESS
041.055                 810X AIO.DES DS     2           DIRECTORY SECTOR
041.057                 811X AIO.DEV DS     2           DEVICE CODE
041.061                 812X AIO.UNI DS     1           UNIT NUMBER (0-9)
                        813X
041.062                 814X AIO.DIR DS     DIRELEN     DIRECTORY ENTRY
                        815X
041.111                 816X AIO.CNT DS     1           SECTOR COUNT
041.112                 817X AIO.EOM DS     1           END OF MEDIA FLAG
041.113                 818X AIO.EOF DS     1           END OF FILE FLAG
041.114                 819X AIO.TFP DS     2           TEMP FILE POINTERS
041.116                 820X AIO.CHA DS     2           ADDRESS OF CHANNEL BLOCK (IOC.DDA)


041.120                 822X S.SCR   DS     2           SYSTEM SCRATCH AREA ADDRESS
```

```
                                825          CODE    PIC             POSITION INDEPENDANT CODE
                                826
                                827
                                828  ***     HDOSOVL2.SYS - HDOS MOUNT/DISMOUNT PROCESSOR
                                829  *
                                830  *       HDOSOVL2 PROCESSES MOUNT/DISMOUNT SYSCALL FUNCTIONS.
                                831  *
                                832  *
                                833  *       ENTRY:  (SP)    = CODE
                                834  *               (SP+2)  = (HL)
                                835  *               (SP+4)  = (RET)
                                836  *               (S.CACC)= USER (ACC)
                                837  *
                                838  *       EXIT:   TO (SP+4)
                                839  *               (PSW)   = 'C' CLEAR IF OK
                                840  *                         'C' SET   IF ERROR
                                841  *                         (A)  = ERROR CODE
                                842  *
                                843  *       USES:   ALL
                                844  *
                                845
000.006  361                    846  HOSOVL2 POP     PSW             (A) = CODE
000.007  326 200                847          SUI     .MOUNT
000.011  322 021 000            848          JNC     HOS2            COMMAND IS IN RANGE
                                849
                                850  *       COMMAND ERROR
                                851
000.014  341                    852  HOS1    POP     H               RESTORE USER HL
000.015  076 003                853  ERRILC  MVI     A,EC.ILC        ILLEGAL CODE
000.017  067                    854          STC
000.020  311                    855          RET                     EXIT WITH ERROR
                                856
000.021  376 005                857  HOS2    CPI     HOSVECL         SEE IF IN RANGE
000.023  322 014 000            858          JNC     HOS1            NOT IN RANGE
000.026  041 046 000            859          LXI     H,HOSVEC
000.031  207                    860          ADD     A               (A) = 2*CODE
000.032  315 101 030            861          CALL    $DADA.
000.035  176                    862          MOV     A,M
000.036  043                    863          INX     H
000.037  146                    864          MOV     H,M
000.040  157                    865          MOV     L,A
000.041  343                    866          XTHL                    ((SP)) = PROCESSOR ADDRESS, (HL) = USER HL
000.042  072 006 041            867          LDA     S.CACC          (A) = USER ACC
000.045  311                    868          RET                     ENTER PROCESSOR
                                869
000.046                         870  HOSVEC  DS      0               JUMP VECTORS
                                871
000.000                         872          ERRNZ   *-HOSVEC/2+.MOUNT-.MOUNT
000.046  060 000                873          DW      MOUNT
                                874
000.000                         875          ERRNZ   *-HOSVEC/2+.MOUNT-.DMOUN
000.050  114 000                876          DW      DMOUN
                                877
000.000                         878          ERRNZ   *-HOSVEC/2+.MOUNT-.MONMS
000.052  147 000                879          DW      MOUNMS
                                880
```

```
000.000                 881          ERRNZ   *-HOSVEC/2+.MOUNT-.DMNMS
000.054  244 000        882          DW      DMONMS
                        883
000.000                 884          ERRNZ   *-HOSVEC/2+.MOUNT-.RESET
000.056  025 001        885          DW      RESET
                        886
000.005                 887  HOSVECL EQU     *-HOSVEC/2      MAX FUNCTION INDEX
```

```
                                    890  ***     MOUNT   - MOUNT DISK
                                    891  *
                                    892  *       MOUNT DISK ON SPECIFIED UNIT OF SELECTED DEVICE
                                    893  *
                                    894  *
                                    895  *       ENTRY:  (HL)    = ADDRESS OF DEVICE SPECIFICATION
                                    896  *
                                    897  *       EXIT:   (PSW)   = 'C' SET   IF ERROR
                                    898  *                           (A)  = ERROR CODE
                                    899  *                         'C' CLEAR IF NO ERROR
                                    900  *                         'Z' CLEAR IF AN ABORT
                                    901  *
                                    902  *       USES:   ALL
                                    903  *
                                    904  *       MODIFICATIONS:  3/6/79 B. WATZMAN TO ALLOW MOUNTING
                                    905  *                       SY0: IF NOT CURRENTLY MOUNTED
                                    906
000.060                             907  MOUNT   EQU     *
000.001                             908          IF      DEBUG
                                    909          CALL    $TYPTX
                                    910          DB      'WE ARE IN MOUNT',ENL
                                    911          ENDIF
000.060  315 147 000                912          CALL    MOUNMS
000.063  330                        913          RC
000.064  300                        914          RNZ                     THERE WAS AN ABORT
000.065  315 215 002                915          CALL    GETLAB          GET LABEL FOR MESSAGE
000.070  330                        916          RC
000.071  041 101 000                917          LXI     H,MOUA
000.074  315 254 002                918          CALL    IMM             ISSUE MESSAGE
000.077  257                        919          XRA     A               SET ZERO FLAG FOR NO ABORT
000.100  311                        920          RET
                                    921
000.101  115 157 165                922  MOUA    DB      'Mounted on',' '+200Q


                                    924  ***     DMOUN   - DISMOUNT DISK
                                    925  *
                                    926  *       DISMOUNT DISK ON SELECTED DRIVE
                                    927  *
                                    928  *
                                    929  *       IF  AN ATTEMPT IS MADE TO DISMOUNT SY0:,  THEN
                                    930  *          FORCE OVERLAY IN IF THERE IS ROOM
                                    931  *
                                    932  *
                                    933  *       ENTRY:  (HL)    = ADDRESS OF DEVICE SPECIFICATION
                                    934  *
                                    935  *       EXIT:   (PSW)   = 'C' SET   IF ERROR
                                    936  *                           (A)  = ERROR CODE
                                    937  *
                                    938  *       USES:   ALL
                                    939  *
                                    940  *       MODIFICATIONS:  3/6/79 BY B. WATZMAN TO ALLOW DISMOUNT
                                    941  *                       OF SY0: IFF BOTH STD. OVERLAY & THIS CODE
                                    942  *                       ARE BOTH RESIDENT
```

```
                                    943
000.114                             944  DMOUN   EQU     *
000.001                             945          IF      DEBUG
                                    946          CALL    $TYPTX
                                    947          DB      'WE ARE IN DMOUN',ENL
                                    948          ENDIF
000.114  315 244 000                949          CALL    DMONMS
000.117  330                        950          RC
000.120  041 127 000                951          LXI     H,DMOA
000.123  315 254 002                952          CALL    IMM
000.126  311                        953          RET
                                    954
000.127  104 151 163                955  DMOA    DB      'Dismounted from',' '+200Q
```

```
                                    958  ***     MOUNMS - MOUNT/NO MESSAGE
                                    959  *
                                    960  *       MOUNT SPECIFIED UNIT OF SELECTED DEVICE WITHOUT ISSUING MOUNT MESSAGE.
                                    961  *
                                    962  *       IF  AN ATTEMPT IS MADE TO MOUNT SY0:  AND  IT IS A SYSTEM DISK
                                    963  *         MOUNT A NEW SYSTEM DISK
                                    964  *
                                    965  *       ENTRY:  (A)     = NUMBER OF DISK SOUGHT (0 FOR ANY DISK).
                                    966  *               (HL)    = ADDRESS OF DEVICE SPECIFICATION
                                    967  *
                                    968  *       EXIT:   (PSW)   = 'C' SET   IF ERROR
                                    969  *                           (A)  = ERROR CODE
                                    970  *                         'C' CLEAR IF NO ERROR
                                    971  *                         'Z' CLEAR IF THERE WAS AN ABORT
                                    972  *
                                    973  *       USES:   ALL
                                    974  *
                                    975  *
                                    976
000.147  365                        977  MOUNMS  PUSH    PSW
000.001                             978          IF      DEBUG
                                    979          CALL    $TYPTX
                                    980          DB      'In MOUNMS',ENL
                                    981          ENDIF
000.150  315 263 001                982          CALL    CDM
000.153  332 242 000                983          JC      MOU3            IF AN ERROR IN DEVICE NAME
000.156  312 166 000                984          JZ      MOU1            DEVICE IS NOT MOUNTED
                                    985
000.161  361                        986          POP     PSW
000.162  067                       987          STC
000.163  076 041                   988          MVI     A,EC.VPM        VOLUME PRESENTLY MOUNTED
000.165  311                       989          RET
                                    990
000.166  361                       991  MOU1    POP     PSW
000.167  345                       992          PUSH    H
000.170  305                       993          PUSH    B
                                    994
000.171  315 203 001               995          CALL    CAB
000.174  302 241 000               996          JNZ     MOU2.5          THERE WAS AN ABORT
000.177  315 241 031               997          CALL    $WER            WRITE ENABLE RAM
000.202  315 365 002               998          CALL    MND             MOUNT NEW DISK
000.205  334 233 000               999          CC      MOU2            IF ERROR
000.210  315 115 003              1000          CALL    PGT             PROCESS GRT
000.213  334 233 000              1001          CC      MOU2            ERROR
000.216  315 105 004              1002          CALL    CDS             CLEAR DIRECTORY SPACES
000.221  334 233 000              1003          CC      MOU2            ERROR
                                   1004
000.224  301                      1005          POP     B
000.225  341                      1006          POP     H
000.226  170                      1007          MOV     A,B
000.227  266                      1008          ORA     M               SET MOUNTED
000.230  167                      1009          MOV     M,A
                                   1010
000.231  257                      1011          XRA     A               SET ZERO FLAG TO INDICATE NO ABORT
000.232  311                      1012          RET
                                   1013
```

```
000.233 376 025     1014 MOU2    CPI     EC.WP       SEE IF ERROR WAS DUE TO WRITE PROTECT
000.235 310         1015         RZ                  YES, CONTINUE AS IF NO ERROR
000.236 063         1016         INX     SP          NO, THERE'S A REAL ERROR
000.237 063         1017         INX     SP          SO RESET THE STACK
000.240 067         1018         STC                 SET THE CARRY AND FALL THROUGH TO MOU2
000.241 301         1019 MOU2.5  POP     B           ERROR RETURN
000.242 341         1020 MOU3    POP     H
000.243 311         1021         RET
```

```
                                1024  ***     DMONMS  - DISMOUNT DEVICE/NO MESSAGE
                                1025  *
                                1026  *       DISMOUNT SELECTED UNIT OF SPECIFIED DEVICE WITHOUT ISSUING DISMOUNT
                                1027  *       MESSAGE
                                1028  *
                                1029  *       IF  AN ATTEMPT IS MADE TO DISMOUNT SY0:  THEN
                                1030  *           FORCE IN OVERLAY
                                1031  *           DISMOUNT OPERATING SYSTEM
                                1032  *
                                1033  *
                                1034  *       ENTRY:  (HL)    = ADDRESS OF DEVICE SPECIFICATION
                                1035  *
                                1036  *       EXIT:   (PSW)   = 'C' SET IF ERROR
                                1037  *                               (A)  = ERROR CODE
                                1038  *
                                1039  *       USES:   ALL
                                1040  *
                                1041
 000,244                        1042  DMONMS  EQU     *
 000,001                        1043          IF      DEBUG
                                1044          CALL    $TYPTX
                                1045          DB      'In DMONMS',ENL
                                1046          ENDIF
 000,244  315 247 005           1047          CALL    $SOB            SKIP OVER LEADING BLANKS
 000,247  345                   1048          PUSH    H               SAVE DEVICE ADDRESS
 000,250  315 263 001           1049          CALL    CDM             CHECK FOR DEVICE MOUNTED
 000,253  321                   1050          POP     D               (DE) = DEVICE ADDRESS
 000,254  330                   1051          RC                      ERROR IN DEVICE SPECIFICATION
 000,255  302 264 000           1052          JNZ     DMO1            MOUNTED
 000,260  076 042               1053          MVI     A,EC.NVM        NO VOLUME PRESENTLY MOUNTED
 000,262  067                   1054          STC
 000,263  311                   1055          RET
                                1056
                                1057  *       OK TO DISMOUNT
                                1058
 000,264  315 106 002           1059  DMO1    CALL    COF             CHECK FOR OPEN FILES ON THE DEVICE
 000,267  330                   1060          RC                      SHOULD NOT DISMOUNT A DISK WITH OPEN FILES.
 000,270  305                   1061          PUSH    B               SAVE INDEX
 000,271  345                   1062          PUSH    H               SAVE DEV.NUM ADDRESS
 000,272  072 320 005           1063          LDA     UNIT
 000,275  062 061 041           1064          STA     AIO.UNI         SET UNIT NUMBER
 000,300  247                   1065          ANA     A
 000,301  302 331 000           1066          JNZ     DMO2            NOT DISMOUNTING SY0:
 000,304  052 350 040           1067          LHLD    S.OFWA          (HL) = FWA OF OVERLAY TABLE
 000,307  021 006 000           1068          LXI     D,OVL0*OVL.ENS+OVL.FLB
 000,312  031                   1069          DAD     D               (DE) = ADDR OF FLAG BYTE
 000,313  076 002               1070          MVI     A,OVL.RES
 000,315  246                   1071          ANA     M
 000,316  312 017 001           1072          JZ      DMO4            OVL0 NOT PERM. RESIDENT
 000,321  021 010 000           1073          LXI     D,OVL.ENS
 000,324  031                   1074          DAD     D
 000,325  246                   1075          ANA     M
 000,326  312 017 001           1076          JZ      DMO4            OVL1 NOT PERM. RESIDENT
                                1077
 000,331  052 323 005           1078  DMO2    LHLD    UNITAB
 000,334  315 234 030           1079          CALL    $INDL
```

```
000.337  003 000        1080            DW      UNT.GTS
000.341  325            1081            PUSH    D
000.342  315 234 030    1082            CALL    $INDL           DE = GRT RAM ADDRESS
000.345  001 000        1083            DW      UNT.GRT
000.347  341            1084            POP     H               HL = GRT SECTOR
                        1085
000.350  001 000 001    1086            LXI     B,256           (BC) = COUNT
000.353  076 001        1087            MVI     A,DC.WRI
000.355  315 130 040    1088            CALL    SYDD            WRITE GRT BACK TO DISK
000.360  341            1089            POP     H
000.361  301            1090            POP     B
000.362  170            1091            MOV     A,B
000.363  057            1092            CMA
000.364  246            1093            ANA     M
000.365  167            1094            MOV     M,A             CLEAR BIT SHOWING MOUNT
000.366  315 233 002    1095            CALL    GETLAR.
000.371  330            1096            RC                      ERROR
000.372  041 127 000    1097            LXI     H,DMOA
000.375  072 320 005    1098            LDA     UNIT
001.000  247            1099            ANA     A
001.001  300            1100            RNZ                     WAS NOT SY0:
                        1101
                        1102  *         FLAG SYSTEM DISMOUNTED
                        1103
001.002  072 032 041    1104            LDA     S.MOUNT
001.005  247            1105            ANA     A
001.006  310            1106            RZ                      FLAG SYSTEM DISMOUNTED
001.007  315 035 002    1107            CALL    CDT             CLEAR DEVICE TABLE
001.012  257            1108            XRA     A
001.013  062 032 041    1109            STA     S.MOUNT         FLAG SYSTEM DISMOUNTED
001.016  311            1110            RET
                        1111
                        1112  *         TRIED TO DISMOUNT SY0: WITH SY1: STILL MOUNTEED
                        1113
001.017  341            1114  DMO4      POP     H
001.020  301            1115            POP     B               RESTORE REGS.
001.021  076 044        1116            MVI     A,EC.NFM        NO PROVISION MADE FOR REMOUNTING HDOS
001.023  067            1117            STC
001.024  311            1118            RET
```

```
                                1122  ***     RESET   - RESET DEVICE
                                1123  *
                                1124  *       RESET THE SPECIFIED UNIT OF THE SELECTED DEVICE
                                1125  *       BY ISSUING AND DISMOUNT FOLLOWED BY A MOUNT.
                                1126  *       IF THE DEVICE NAME IS <NULL>, THEN RESET SY0: AND SY1:,
                                1127  *       OTHERWISE, THE DEVICE NAME SHOULD BE IN THE SAME FORMAT AS
                                1128  *       THAT EXPECTED BY MOUNT & DMOUN
                                1129  *
                                1130  *
                                1131  *       ENTRY:  (HL)    =  ADDRESS OF DEVICE SPECIFICATION
                                1132  *
                                1133  *       EXIT:   (PSW)   =  'C' CLEAR IF NO ERROR
                                1134  *                          'C' SET   IF    ERROR
                                1135  *                              (A)  = ERROR CODE
                                1136  *
                                1137  *       USES:   ALL
                                1138  *
                                1139
001.025                         1140  RESET   EQU     *
001.025  315 247 005            1141          CALL    $SOB
001.030  176                    1142          MOV     A,M
001.031  247                    1143          ANA     A
001.032  302 060 001           1144          JNZ     RES1            CALLER SUPPLIED DEVICE NAME
001.035  041 171 001           1145          LXI     H,RESA
001.040  315 025 001           1146          CALL    RESET           RESET SY1:
001.043  330                   1147          RC
001.044  072 334 040           1148          LDA     S.CAADR+1
001.047  247                   1149          ANA     A
001.050  300                   1150          RNZ                     THERE WAS AN ABORT
001.051  041 176 001           1151          LXI     H,RESB
001.054  315 025 001           1152          CALL    RESET           RESET SY0:
001.057  311                   1153          RET
                                1154
001.060  345                   1155  RES1    PUSH    H
001.061  315 114 000           1156          CALL    DMOUN
001.064  341                   1157          POP     H
001.065  322 077 001           1158          JNC     RES2            NO ERROR
001.070  376 042               1159          CPI     EC.NVM
001.072  312 165 001           1160          JZ      RES3            NO VOLUME MOUNTED ERROR NOT CONSIDERED FATAL
001.075  067                   1161          STC                     FLAG ANY OTHER ERRORS AFTER CPI MAY HAVE
001.076  311                   1162          RET                      CLEARED FLAG
                                1163
001.077  315 136 031           1164  RES2    CALL    $TYPTX
001.102  012 007 120           1165          DB      NL,BELL,'Please Replace Diskette in Drive',' '+200Q
001.145  345                   1166          PUSH    H
001.146  076 004               1167          MVI     A,IOC.UNI-IOC.DEV+2
001.150  315 265 005           1168          CALL    $TYPCC
001.153  315 136 031           1169          CALL    $TYPTX
001.156  012 212               1170          DB      NL,ENL
001.160  315 273 004           1171          CALL    WDO             WAIT FOR DRIVE TO OPEN
001.163  341                   1172          POP     H
001.164  300                   1173          RNZ                     AN ABORT IS PENDING
001.165  315 060 000           1174  RES3    CALL    MOUNT
001.170  311                   1175          RET
                                1176
001.171  123 131 061           1177  RESA    DB      'SY1:',0
```

```
001.176  123 131 060  1178  RESB    DB      'SY0:',0
```

```
                        1182  **      CAB     -  CHECK ABORT
                        1183  *
                        1184  *       CAB WAITS FOR A REVOLUTION OF THE DISK WHILE MONITORING THE
                        1185  *       ABORT FLAG.  AFTER ONE COMPLETE RPM, ANY ATTEMPTS TO ABORT
                        1186  *       ARE FUTILE.
                        1187  *
                        1188  *
                        1189  *       ENTRY   NONE
                        1190  *
                        1191  *       EXIT    (PSW)   =  'Z' CLEAR IF     TO ABORT
                        1192  *                       =  'Z' SET   IF NOT TO ABORT
                        1193  *
                        1194  *       USES    (PSW),(D)
                        1195  *
                        1196
001.203 026 014         1197  CAB     MVI     D,12            WAIT FOR 12 HOLES, OR AT LEAST ONE RPM
                        1198
                        1199  *       WAIT FOR A HOLE TO PASS
                        1200
001.205 315 234 001     1201  CAB1    CALL    CAB.
001.210 300             1202          RNZ                     AN ABORT IS PENDING
001.211 346 001         1203          ANI     DF.HD
001.213 302 205 001     1204          JNZ     CAB1            WE ARE WATCHING A HOLE
                        1205
                        1206  *       WAIT FOR A GAP TO PASS
                        1207
001.216 315 234 001     1208  CAB2    CALL    CAB.
001.221 300             1209          RNZ                     AN ABORT IS PENDING
001.222 346 001         1210          ANI     DF.HD
001.224 312 216 001     1211          JZ      CAB2            WE ARE WATCHING A GAP
                        1212
001.227 025             1213          DCR     D               COUNT THE TRANSITION
001.230 302 205 001     1214          JNZ     CAB1
001.233 311             1215          RET                     AT LEAST ONE RPM, AND NO ABORT
                        1216
001.234 373             1217  CAB.    EI
001.235 305             1218          PUSH    B
001.236 072 320 005     1219          LDA     UNIT
001.241 107             1220          MOV     B,A
001.242 004             1221          INR     B
001.243 257             1222          XRA     A
001.244 315 373 004     1223          CALL    BITS            SET THE DEVICE BIT
000.000                 1224          ERRNZ   DF.DS0-2
000.000                 1225          ERRNZ   DF.DS1-4
000.000                 1226          ERRNZ   DF.DS2-8
001.247 366 020         1227          ORI     DF.MO
001.251 323 177         1228          OUT     DF.DC           ON MOTOR, AND DRIVE SELECT
001.253 072 334 040     1229          LDA     S.CAADR+1
001.256 247             1230          ANA     A
001.257 333 177         1231          IN      DF.DC
001.261 301             1232          POP     B
001.262 311             1233          RET
```

```
                                1235  **      CDM - CHECK FOR DEVICE MOUNTED.
                                1236  *
                                1237  *       CDM REFORMATS THE SUPPLIED DEVICE SPECIFICATION
                                1238  *       INTO A 6 CHARACTER FIELD OF THE FORMAT
                                1239  *
                                1240  *       D E V ! <00> <ENL>
                                1241  *
                                1242  *       IN 'DEVNAME'
                                1243  *
                                1244  *
                                1245  *       ENTRY   (HL)    = ADDRESS FOR DEVICE SPECIFICATION
                                1246  *
                                1247  *       EXIT    (PSW)   = 'C' CLEAR IF NO ERROR
                                1248  *                         (B) = BIT INDEX FOR THE SPECIFIED UNIT
                                1249  *                         (HL) = ADDREESS OF DEV.NUM CELL
                                1250  *                         'Z' SET IF NOT MOUNTED
                                1251  *                         'Z' CLEAR IF MOUNTED
                                1252  *                       = 'C' SET IF ERROR
                                1253  *                         (A) = ERROR CODE
                                1254  *
                                1255  *       USES    ALL
                                1256  *
                                1257
001.263  315 247 005           1258  CDM     CALL    $SOB            SKIP BLANKS
000.001                         1259          IF      DEBUG
                                1260          CALL    TRACE
                                1261          DB      'IN CDM...',ENL
                                1262          ENDIF
001.266  176                    1263          MOV     A,M
001.267  315 213 005           1264          CALL    $MCU            MAP TO UPPER CASE
001.272  376 123                1265          CPI     'S'             CHECK 'S'
001.274  302 025 002           1266          JNE     CDM2            ERROR
001.277  043                    1267          INX     H
001.300  176                    1268          MOV     A,M
001.301  315 213 005           1269          CALL    $MCU            MAP TO UPPER CASE
001.304  376 131                1270          CPI     'Y'
001.306  302 025 002           1271          JNE     CDM2            NOT 'Y'
001.311  043                    1272          INX     H
001.312  176                    1273          MOV     A,M
001.313  326 060                1274          SUI     '0'
001.315  332 025 002           1275          JC      CDM2
001.320  376 003                1276          CPI     3
001.322  322 025 002           1277          JNC     CDM2
001.325  043                    1278          INX     H
001.326  107                    1279          MOV     B,A             (B) = UNIT
001.327  062 320 005           1280          STA     UNIT
001.332  306 060                1281          ADI     '0'
001.334  062 033 002           1282          STA     CDMB
001.337  176                    1283          MOV     A,M
001.340  376 072                1284          CPI     ':'
001.342  302 025 002           1285          JNE     CDM2            ERROR
                                1286
                                1287  *       GOT VALID DEVICE.
                                1288  *
                                1289  *       (B) = UNIT NUMBER
                                1290
```

```
                                1291  *       SET-UP TABLE BASES
                                1292
001.345  052 354 040            1293          LHLD    S.DFWA
001,350  042 321 005            1294          SHLD    DEVTAB
001.353  021 012 000            1295          LXI     D,DEV.UNT
001,356  031                    1296          DAD     D
001.357  072 320 005            1297          LDA     UNIT
001,362  315 027 041            1298          CALL    S.GUP
001.365  042 323 005            1299          SHLD    UNTTAB
                                1300
001.370  305                    1301          PUSH    B
001.371  315 224 005            1302          CALL    $MOVEL
001.374  004 000 031            1303          DW      4,CDMA,DEVNAME  COPY NAME
002,002  301                    1304          POP     B               (B) = UNIT NUMBER
002.003  052 321 005            1305          LHLD    DEVTAB
002,006  021 010 000            1306          LXI     D,DEV.MUM
002.011  031                    1307          DAD     D
002,012  072 320 005            1308          LDA     UNIT
002.015  107                    1309          MOV     B,A
002,016  257                    1310          XRA     A
002.017  315 373 004            1311          CALL    BITS            SET THE UNIT BIT
002,022  107                    1312          MOV     B,A
002.023  246                    1313          ANA     M               SET FLAGS
002,024  311                    1314          RET
                                1315
                                1316  *       ERROR
                                1317
002.025  076 005                1318  CDM2    MVI     A,EC.DNS
002.027  067                    1319          STC
002,030  311                    1320          RET
                                1321
002.031  123 131                1322  CDMA    DB      'SY'
002.033  061 072                1323  CDMB    DB      '1:'



                                1325  **      CDT     - CLEAR DEVICE TABLE
                                1326  *
                                1327  *       IF ANY DRIVERS ARE IN MEMORY, FLAG THEM PERMANENTLY RESIDENT,
                                1328  *       OTHERWISE, REMOVE THEIR ENTRY FROM THE DEVICE LIST.
                                1329  *
                                1330  *
                                1331  *       ENTRY:  NONE
                                1332  *
                                1333  *       EXIT:   NONE
                                1334  *
                                1335  *       USES:   ALL
                                1336  *
                                1337
002.035  052 354 040            1338  CDT     LHLD    S.DFWA
                                1339
002.040  176                    1340  CDT1    MOV     A,M
002.041  247                    1341          ANA     A
002.042  310                    1342          RZ                      TO THE END OF THE DEVICE TABLE
000.000                         1343          ERRNZ   DV.EL
```

```
002.043  376 001      1344          CPI   DV.NU
002.045  312 077 002  1345          JZ    CDT3          THIS ENTRY NOT IN USE
002.050  345          1346          PUSH  H             SAVE FWA OF DEVICE ENTRY
002.051  021 002 000  1347          LXI   D,DEV.RES
002.054  031          1348          DAD   D             (DE) = ADDRESS OF DEV.RES FLAGS
002.055  176          1349          MOV   A,M
002.056  346 001      1350          ANI   DR.IM
002.060  312 073 002  1351          JZ    CDT2          DRIVER NOT IN MEMORY
002.063  176          1352          MOV   A,M
002.064  366 003      1353          ORI   DR.IN+DR.PR   FLAG IN MEMORY AND PERMANENTLY RESIDENT
002.066  167          1354          MOV   M,A
002.067  341          1355          POP   H
002.070  303 077 002  1356          JMP   CDT3
                      1357
002.073  341          1358  CDT2    POP   H
002.074  076 001      1359          MVI   A,DV.NU       FLAG DEVICE ENTRY NOT USED
002.076  167          1360          MOV   M,A
                      1361
002.077  021 017 000  1362  CDT3    LXI   D,DEVELEN
002.102  031          1363          DAD   D
002.103  303 040 002  1364          JMP   CDT1


                      1366  **      COF   - CHECK OUTPUT FILE
                      1367  *
                      1368  *       CHECK THE SPECIFIED DEVICE FOR ANY OPEN FILES.
                      1369  *
                      1370  *
                      1371  *       ENTRY:  (DE)    =  ADDRESS OF DEVICE SPECIFICATION
                      1372  *
                      1373  *       EXIT:   (PSW)   = 'C' CLEAR IF NO OPEN FILES
                      1374  *                         = 'C' SET   IF    OPEN FILES
                      1375  *
                      1376  *       USES:   (PSW),(DE)
                      1377  *
                      1378
002.106  305          1379  COF     PUSH  B
002.107  345          1380          PUSH  H
002.110  032          1381          LDAX  D             SET UP THE COMPARISON STRING
002.111  062 212 002  1382          STA   COFA
002.114  023          1383          INX   D
002.115  032          1384          LDAX  D
002.116  062 213 002  1385          STA   COFA+1
002.121  023          1386          INX   D
002.122  032          1387          LDAX  D
002.123  326 060      1388          SUI   '0'
002.125  062 214 002  1389          STA   COFA+2
002.130  016 003      1390          MVI   C,3
000.000               1391          ERRNZ IOC.UNI-IOC.DEV-2
002.132  052 352 040  1392          LHLD  S.CFWA        FIRST WORD OF CHANNEL TABLE
                      1393
002.135  345          1394  COF1    PUSH  H
002.136  021 004 000  1395          LXI   D,IOC.FLG
002.141  031          1396          DAD   D
```

```
002.142  176           1397          MOV     A,M
002.143  247           1398          ANA     A
002.144  312 167 002   1399          JZ      COF2            FILE NOT OPEN
002.147  076 014       1400          MVI     A,IOC.DEV-IOC.FLG
002.151  315 101 030   1401          CALL    $DADA,          (HL) = ENTRY OF CHANNEL AND UNIT IN TABLE
002.154  305           1402          PUSH    B
002.155  021 212 002   1403          LXI     D,COFA          (DE) = COMPARISON STRING ADDRESS
002.160  315 060 030   1404          CALL    $COMP
002.163  301           1405          POP     B
002.164  312 203 002   1406          JZ      COF3            HAVE A MATCH
002.167  341           1407  COF2    POP     H               (HL) = IOC.LNK
000.000                1408          ERRNZ   IOC.LNK
002.170  315 211 030   1409          CALL    $HLIHL
002.173  175           1410          MOV     A,L
002.174  264           1411          ORA     H
002.175  302 135 002   1412          JNZ     COF1
002.200  341           1413          POP     H
002.201  301           1414          POP     B
002.202  311           1415          RET
                       1416
002.203  341           1417  COF3    POP     H
002.204  067           1418          STC
002.205  076 043       1419          MVI     A,EC.FOD        FILE OPEN ON DEVICE
002.207  341           1420          POP     H
002.210  301           1421          POP     B
002.211  311           1422          RET
                       1423
002.212                1424  COFA    DS      3               TEMPORARY COMPARISON STRING


                       1426  **      GETLAB  - GET LABEL
                       1427  *
                       1428  *       READ DISKETTE LABEL, AND STORE IN RAM.
                       1429  *
                       1430  *
                       1431  *       ENTRY:  UNIT = DEVICE UNIT NO.
                       1432  *
                       1433  *       EXIT:   (PSW)   = 'C' CLEAR IF NO ERROR
                       1434  *                         'C' SET   IF    ERROR
                       1435  *                          (A)  = ERROR CODE
                       1436  *
                       1437  *       USES:   ALL
                       1438  *
                       1439
002.215  072 320 005   1440  GETLAB  LDA     UNIT
000.001                1441          IF      DEBUG
                       1442          CALL    TRACE
                       1443          DB      'IN GETLAB',ENL
                       1444          ENDIF
002.220  062 061 041   1445          STA     AIO.UNI         SET UNIT
002.223  076 007       1446          MVI     A,DC.ABT
002.225  315 130 040   1447          CALL    SYDD            DRIVER ABORT
002.230  315 241 031   1448          CALL    $WER
002.233  001 000 001   1449  GETLAB. LXI     B,256
```

```
002.236  021 000 027  1450          LXI     D,LABEL
002.241  041 011 000  1451          LXI     H,DDF.LAB
002.244  076 002      1452          MVI     A,DC.RER
002.246  315 130 040  1453          CALL    SYDD
002.251  076 046      1454          MVI     A,EC.DNR        COULD NOT READ LABEL, NOT PROPERLY INITIALIZED
002.253  311          1455          RET


                      1457  **      IMM - ISSUE MOUNT MESSAGE.
                      1458  *
                      1459  *       IMM TYPES THE MOUNTING MESSAGE:
                      1460  *
                      1461  *       VOLUME NNN MOUNTED ON DEV:
                      1462  *       LABEL: XXXX ... XXX
                      1463  *
                      1464  *       ENTRY   LABEL SECTOR READ
                      1465  *               (HL) = ADDRESS OF MESSAGE VERB STRING (.PRINT FORMAT)
                      1466  *       EXIT    NONE
                      1467  *       USES    ALL
                      1468
                      1469
002.254  072 000 027  1470  IMM     LDA     LABEL+LAB.SER
000.001              1471          IF      DEBUG
                      1472          CALL    TRACE
                      1473          DB      'IN IMM',ENL
                      1474          ENDIF
002.257  345          1475          PUSH    H               SAVE VERB
002.260  117          1476          MOV     C,A
002.261  006 000      1477          MVI     B,0
002.263  041 344 002  1478          LXI     H,IMMB
002.266  076 003      1479          MVI     A,3
002.270  315 157 031  1480          CALL    $UDD            UNPACK VOLUME NUMBER
002.273  315 224 005  1481          CALL    $MOVEL
002.276  004 000 312  1482          DW      4,DEVNAME,IMMC  SET DEV NAME
002.304  041 335 002  1483          LXI     H,IMMA
002.307  377 003      1484          DB      SYSCALL,.PRINT  PRINT MESSAGE
002.311  341          1485          POP     H
002.312  377 003      1486          DB      SYSCALL,.PRINT  PRINT VERB
002.314  041 351 002  1487          LXI     H,IMMC
002.317  377 003      1488          DB      SYSCALL,.PRINT  PRINT THE REST OF IT
002.321  041 021 027  1489          LXI     H,LABEL+LAB.LAB
002.324  315 042 005  1490          CALL    $DTB            DELETE TRAILING BLANKS
002.327  315 265 005  1491          CALL    $TYPCC          TYPE LABEL
002.332  303 034 005  1492          JMP     $CRLF           CRLF AND EXIT
                      1493
002.335  126 157 154  1494  IMMA    DB      'Volume '
002.344  130 130 130  1495  IMMB    DB      'XXX',',' '+200Q
002.351  104 105 126  1496  IMMC    DB      'DEV:',NL,'Label:',' '+200Q
```

```
                                1498  **      MND - MOUNT SYSTEM DISK.
                                1499  *
                                1500  *       MND MOUNTS A NEW DISK INTO 'SY' UNIT 'UNIT'
                                1501  *
                                1502  *       1) ABORT DRIVER
                                1503  *       2) READ LABEL RECORD
                                1504  *       3) SET VOLUME NUMBER FOR DRIVER
                                1505  *
                                1506  *       EXIT    'C' CLEAR IF OK
                                1507  *                LABEL = LABEL SECTOR
                                1508  *               'C' SET IF ERROR
                                1509
                                1510
002.365  315 215 002            1511  MND     CALL    GETLAB          GET LABEL
000.001                         1512          IF      DEBUG
                                1513          CALL    TRACE
                                1514          DB      'IN MND',ENL
                                1515          ENDIF
002.370  330                    1516          RC                      BAD ERROR
                                1517
                                1518  *       CALL DEVICE MOUNT ROUTINE
                                1519
002.371  072 010 027            1520          LDA     LABEL+LAB.VLT   (A) = VOLUME TYPE
002.374  376 002                1521          CPI     LAB.NOD
002.376  322 111 003            1522          JNC     MND2            DEVICE DOESNT HAVE A DIRECTORY
003.001  072 000 027            1523          LDA     LABEL+LAB.SER
003.004  157                    1524          MOV     L,A
003.005  046 000                1525          MVI     H,0             (HL) = SERIAL NUMBER
003.007  076 010                1526          MVI     A,DC.MOU
003.011  315 130 040            1527          CALL    SYDD            MOUNT UNIT
003.014  330                    1528          RC                      BAD ERROR
                                1529
                                1530  *       SETUP ENTRY IN DEVLST
                                1531
003.015  052 003 027            1532          LHLD    LABEL+LAB.DIS
003.020  353                    1533          XCHG
003.021  052 323 005            1534          LHLD    UNTTAB
003.024  315 124 005            1535          CALL    $INDS           SAVE  DIRECTORY SECTOR POINTER IN UNIT TABLE
003.027  005 000                1536          DW      UNT.DIS
003.031  353                    1537          XCHG
003.032  052 005 027            1538          LHLD    LABEL+LAB.GRT
003.035  353                    1539          XCHG
003.036  315 124 005            1540          CALL    $INDS           SAVE  GRT SECTOR POINTER IN TABLE
003.041  003 000                1541          DW      UNT.GTS
                                1542
                                1543  *       SEE IF WRITE-PROTECTED
                                1544
003.043  345                    1545          PUSH    H
003.044  001 000 000            1546          LXI     B,0             SET UP A DUMMY WRITE OF ZERO BYTES
003.047  150                    1547          MOV     L,B             TO SEE IF DISKETTE IS WRITE PROTECTED
003.050  140                    1548          MOV     H,B
003.051  076 001                1549          MVI     A,DC.WRI        TRY IT
003.053  315 130 040            1550          CALL    SYDD
003.056  341                    1551          POP     H
                                1552
003.057  365                    1553          PUSH    PSW             SAVE CARRY FLAG
```

```
003.060  315 103 005  1554          CALL    $INDLB
003.063  000 000      1555          DW      UNT.FLG
003.065  107          1556          MOV     B,A
003.066  361          1557          POP     PSW             RESTORE CARRY
003.067  170          1558          MOV     A,B             A = FLAG
003.070  332 100 003  1559          JC      MND.5           WAS WRITE PROTECTED
                      1560
003.073  366 004      1561          ORI     DT.CW           SET CAPABLE OF WRITE
003.075  303 102 003  1562          JMP     MND.6
                      1563
003.100  346 373      1564  MND.5   ANI     377Q-DT.CW      SET INCAPABLE OF WRITE
                      1565
003.102  315 160 005  1566  MND.6   CALL    $INDSB          UPDATE FLAG BYTE
003.105  000 000      1567          DW      UNT.FLG
003.107  267          1568          ORA     A               CLEAR 'C'
003.110  311          1569          RET                     AND RETURN
                      1570
                      1571  *       DEVICE DOES NOT HAVE A DIRECTORY.
                      1572
003.111  076 045      1573  MND2    MVI     A,EC.DNI        DISK NOT INITIALIZED
003.113  067          1574          STC                     FLAG ERROR
003.114  311          1575          RET
```

```
                      1577  **      PGT - PREPARE GRT.
                      1578  *
                      1579  *       PGT PREPARES THE GROUP RESERVATION TABLE BY READING BOTH THE
                      1580  *       GRT AND THE RGT INTO MEMORY.
                      1581  *
                      1582  *       THE GROUPS UNRESERVED VIA THE RGT ARE FLAGGED FREE (LINK TO GROUP 1)
                      1583  *
                      1584  *       EACH DIRECTORY ENTRY IS THEN CHECKED, AND ITS GROUP IS
                      1585  *       FOLLOWED THROUGH THE GRT, THE CHAIN IS DUPLICATED INTO THE GRT BEING
                      1586  *       BUILT.
                      1587  *
                      1588  *       WHEN THIS PROCESS IS COMPLETE, ANY UNUSED GROUPS ARE CHAINED TO
                      1589  *       THE FREE LIST.
                      1590  *
                      1591  *       ENTRY   NONE
                      1592  *       EXIT    (HL) = SECTOR ADDRESS OF LAST DIRECTORY BLOCK CONTAINING FILES
                      1593  *       USES    ALL
                      1594  *       MODIFICATIONS:  B. WATZMAN 3/7/89 TO ALLOW USING EITHER SY1: OR SY0:
                      1595  *
                      1596
003.115  315 241 031  1597  PGT     CALL    $WER            WRITE ENABLE PROTECTED RAM
000.001               1598          IF      DEBUG
                      1599          CALL    TRACE
                      1600          DB      'IN PGT',ENL
                      1601          ENDIF
                      1602
                      1603  *       READ THE RGT INTO GRT MEMORY
                      1604
003.120  052 323 005  1605          LHLD    UNTTAB
003.123  315 234 030  1606          CALL    $INDL           DE = GRT ADDRESS
```

```
003.126  001 000      1607          DW      UNT.GRT
003.130  353          1608          XCHG
003.131  042 103 004  1609          SHLD    PGTG            SAVE GRT ADDRESS
003.134  353          1610          XCHG
003.135  345          1611          PUSH    H
003.136  041 012 000  1612          LXI     H,DDF.RGT       RGT SECTOR ADDRESS
003.141  315 042 004  1613          CALL    PGT10.          READ RGT INTO GRT AREA
003.144  341          1614          POP     H
003.145  330          1615          RC                      ERROR
                      1616
                      1617  *       READ IN THE GRT
                      1618
003.146  315 234 030  1619          CALL    $INDL
003.151  003 000      1620          DW      UNT.GTS         DE = GRT SECTOR
003.153  345          1621          PUSH    H
003.154  353          1622          XCHG                    HL = GRT SECTOR
003.155  021 000 027  1623          LXI     D,PGTA          DE = WORK SPACE
003.160  315 042 004  1624          CALL    PGT10.
003.163  341          1625          POP     H
003.164  330          1626          RC
                      1627
                      1628  *       INITIALIZE FOR BUILD TABLE LOOP
                      1629
003.165  315 234 030  1630          CALL    $INDL
003.170  005 000      1631          DW      UNT.DIS
003.172  052 120 041  1632          LHLD    S.SCR                                   /79.11.GC/
003.175  315 124 005  1633          CALL    $INDS           STORE FIRST DIR. SECT.  /79.11.GC/
003.200  376 001      1634          DW      DIS.LNK                                 /79.12.GC/
                      1635
003.202  257          1636          XRA     A
003.203  062 000 027  1637          STA     PGTA            CLEAR OLD FREE CHAIN
                      1638
                      1639  *       READ DIRECTORY BLOCK
                      1640
003.206  052 120 041  1641  PGT3    LHLD    S.SCR                                   /79.11.GC/
003.211  315 234 030  1642          CALL    $INDL                                   /79.11.GC/
003.214  376 001      1643          DW      DIS.LNK                                 /79.12.GC/
003.216  353          1644          XCHG                    HL = DIRECTORY LINK SECTOR NUMBER /79.11.GC/
                      1645  *                               DE = SECTOR SCRATCH ADDRESS     /79.12.GC/
                      1646
003.217  174          1647          MOV     A,H
003.220  265          1648          ORA     L
003.221  312 354 003  1649          JZ      PGT7            ALL DONE
                      1650
003.224  001 000 002  1651          LXI     B,512                                   /79.12.GC/
003.227  257          1652          XRA     A                                       /79.12.GC/
000.000               1653          ERRNZ   DC.REA                                 /79.12.GC/
003.230  315 130 040  1654          CALL    SYDD                                    /79.12.GC/
003.233  330          1655          RC                                              /79.12.GC/
                      1656
                      1657  *       SAVE BLOCK INFO IN CONVENIENT PLACE
                      1658
003.234  052 120 041  1659          LHLD    S.SCR                                   /79.12.GC/
003.237  315 103 005  1660          CALL    $INDLB                                  /79.11.GC/
003.242  373 001      1661          DW      DIS.ENL                                 /79.11.GC/
003.244  062 101 004  1662          STA     PGTE            SAVE DIRECTORY ENTRY LENGTH  /79.11.GC/
```

```
003.247  315 234 030  1663           CALL    $INDL                                   /79.12.GC/
003.252  374 001      1664           DW      DIS.SEC                                 /79.12.GC/
003.254  353          1665           XCHG                                            /79.12.GC/
003.255  042 077 004  1666           SHLD    PGTD            SAVE THIS BLOCK NUMBER  /79.12.GC/
003.260  353          1667           XCHG                                            /79.12.GC/
                      1668
                      1669  *        SCAN DIRECTORY FOR ENTRYS, TRANSFER THE CHAIN TO THE NEW GRT.
                      1670
003.261  176          1671  PGT4     MOV     A,M             (A) = 1ST CHARACTER OF NAME
000.000               1672           ERRNZ   DF.EMP-377Q
003.262  074          1673           INR     A
003.263  312 336 003  1674           JZ      PGT6            SPACE IS EMPTY
000.000               1675           ERRNZ   DF.CLR-376Q
003.266  074          1676           INR     A
003.267  312 354 003  1677           JZ      PGT7            ALL DONE
003.272  372 336 003  1678           JM      PGT6            ** DEBUG ** SHOULD NOT OCCUR
003.275  345          1679           PUSH    H               SAVE ADDRESS OF DIRECTORY ENTRY
                      1680
003.276  315 103 005  1681           CALL    $INDLB
003.301  020 000      1682           DW      DIR.FGN
003.303  157          1683           MOV     L,A             L  = FIRST GROUP NUMBER
                      1684
                      1685  *        COPY CHAIN TO GRT
                      1686
003.304  046 027      1687  PGT5     MVI     H,PGTA/256
003.306  176          1688           MOV     A,M
                      1689
003.307  365          1690           PUSH    PSW
003.310  072 104 004  1691           LDA     PGTG+1
003.313  147          1692           MOV     H,A             SET UP THE HIGH ORDER BYTE OF GRT ADDR
003.314  361          1693           POP     PSW
                      1694
003.315  065          1695           DCR     M               SEE IF FREE
003.316  167          1696           MOV     M,A
003.317  302 035 004  1697           JNZ     PGTERR          WAS NOT FREE ! DOUBLE LINKAGE & EXIT
003.322  247          1698           ANA     A
003.323  157          1699           MOV     L,A
003.324  302 304 003  1700           JNZ     PGT5            MORE TO GO
003.327  052 077 004  1701           LHLD    PGTD
003.332  042 032 004  1702           SHLD    PGTB            SAVE SECTOR ADDRESS OF BLOCK
003.335  341          1703           POP     H               (HL) = DIRECTORY SECTOR POINTER
                      1704
003.336  072 101 004  1705  PGT6     LDA     PGTE            DIRECTORY ENTRY LENGTH
003.341  315 101 030  1706           CALL    $DADA.          HL = HL + DIR. LENGTH
                      1707
003.344  176          1708           MOV     A,M
003.345  247          1709           ANA     A               SEE IF ENTRY
003.346  302 261 003  1710           JNZ     PGT4            MORE ENTRYS TO GO
003.351  303 206 003  1711           JMP     PGT3            GET NEW SECTOR
                      1712
                      1713  *        ALL DONE, LINK UNUSED GUYS
                      1714
003.354  016 000      1715  PGT7     MVI     C,0             (C) = NEXT FREE GROUP
003.356  052 103 004  1716           LHLD    PGTG
003.361  021 377 000  1717           LXI     D,255
003.364  031          1718           DAD     D               HL = GRT ADDRESS + 255
```

```
                                1719
003.365  076 001               1720  PGT8    MVI     A,1
003.367  276                   1721          CMP     M
003.370  302 375 003           1722          JNE     PGT9            NOT FREE
003.373  161                   1723          MOV     M,C             LINK TO NEXT FREE
003.374  115                   1724          MOV     C,L             SAVE THIS ONES INDEX
003.375  055                   1725  PGT9    DCR     L
003.376  302 365 003           1726          JNZ     PGT8            NOT ALL PROCESSED
004.001  161                   1727          MOV     M,C             SET FREE CHAIN
                                1728
004.002  052 323 005           1729          LHLD    UNTTAB
004.005  315 234 030           1730          CALL    $INDL
004.010  003 000               1731          DW      UNT.GTS
004.012  325                   1732          PUSH    D
004.013  315 234 030           1733          CALL    $INDL           DE = GRT ADDRESS
004.016  001 000               1734          DW      UNT.GRT
004.020  341                   1735          POP     H               HL = GRT SECTOR
                                1736
004.021  001 000 001           1737          LXI     B,256
004.024  076 001               1738          MVI     A,DC.WRI
004.026  315 130 040           1739          CALL    SYDD            UPDATE GRT
004.031  041 000 000           1740          LXI     H,0             (HL) = SECTOR NUMBER OF LAST BLOCK WITH FILES
004.032                        1741  PGTB    EQU     *-2
004.034  311                   1742          RET


                                1744  **      PGTERR - LINKAGE ERROR IN DISK FILE STRUCTURE.
                                1745
004.035  076 047               1746  PGTERR  MVI     A,EC.DSC        DISK STRUCTURE IS CORRUPT
004.037  067                   1747          STC
000.001                         1748          IF      DEBUG
                                1749          CALL    TRACE
                                1750          DB      'AT PGTERR',ENL
                                1751          HLT
                                1752          ENDIF
004.040  341                   1753  PGTERR. POP     H
004.041  311                   1754          RET
                                1755


                                1757  **      PGT10. - GET ONE SECTOR
                                1758
004.042  001 000 001           1759  PGT10.  LXI     B,256
000.000                         1760          ERRNZ   DC.REA
004.045  257                   1761          XRA     A
004.046  315 130 040           1762          CALL    SYDD
004.051  311                   1763          RET
                                1764
```

```
                                1766  **      PGT11.  - GET THE SECOND SECTOR OF THE DIRECTORY BLOCK
                                1767
004.052  345                   1768  PGT11.  PUSH    H
004.053  315 067 004           1769          CALL    PGT12.          DE = SECTOR SCRATCH
004.056  052 075 004           1770          LHLD    PGTC
004.061  043                   1771          INX     H
004.062  315 042 004           1772          CALL    PGT10.
004.065  341                   1773          POP     H
004.066  311                   1774          RET

                                1776  **      PGT12.  - GET THE POINTER TO THE SECTOR SCRATCH AREA
                                1777
004.067  353                   1778  PGT12.  XCHG                                            /79.11.GC/
004.070  052 120 041           1779          LHLD    S.SCR                                   /79.11.GC/
004.073  353                   1780          XCHG            DE = SECTOR SCRATCH             /79.11.GC/
004.074  311                   1781          RET                                             /79.11.GC/

004.075  000 000               1783  PGTC    DW      0       DIRECTORY LINK SECTOR
004.077  000 000               1784  PGTD    DW      0       CURRENT BLOCK NUMBER
004.101  000                   1785  PGTE    DB      0       DIRECTORY ENTRY LENGTH
004.102  000                   1786  PGTF    DB      0       SECOND SECT. OF BLOCK ALREADY READ FLAG
                                1787  *                       != 0  =>  TRUE
004.103  000 000               1788  PGTG    DW      0       SAVED GRT ADDRESS

                                1790  **      CDS - CLEAR DIRECTORY SPACES.
                                1791  *
                                1792  *       CDS IS CALLED TO FLAG THE UNUSED ENTRYS AT THE
                                1793  *       END OF THE DIRECTORY AS CLEAR.
                                1794  *
                                1795  *       WHEN A FILE IS DELETED, ITS ENTRY IS FLAGED EMPTY. CDS LOCATES
                                1796  *       EMPTY SPOTS WHICH ARE AFTER THE LAST FILE IN THE DIRECTORY,
                                1797  *       AND FLAGS THEM CLEAER.
                                1798  *
                                1799  *       ENTRY   (HL) = SECTOR NUMBER OF LAST DIRECTORY BLOCK WITH FILES
                                1800  *       EXIT    NONE
                                1801  *       USES    ALL
                                1802
                                1803
004.105  001 000 002           1804  CDS     LXI     B,512
000.001                         1805          IF      DEBUG
                                1806          CALL    TRACE
                                1807          DB      'IN CDS',ENL
                                1808          ENDIF
004.110  315 067 004           1809          CALL    PGT12.          DE = SECTOR SCRATCH
004.113  325                   1810          PUSH    D               SAVE #SECSCR
004.114  257                   1811          XRA     A
000.000                         1812          ERRNZ   DC.REA
004.115  315 130 040           1813          CALL    SYDD            READ DIRECTORY BLOCK
004.120  330                   1814          RC                      ERROR
```

```
004.121  321          1815          POP     D
004.122  142          1816          MOV     H,D
004.123  153          1817          MOV     L,E             (DE) = (HL) = $SECSCR
                      1818
                      1819  *       FIND LAST FILE NAME IN THIS BLOCK
                      1820
004.124  176          1821  CDS1    MOV     A,M
004.125  247          1822          ANA     A
004.126  312 147 004  1823          JZ      CDS3            END OF BLOCK
004.131  372 136 004  1824          JM      CDS2            EMPTY OR CLEAR
004.134  124          1825          MOV     D,H
004.135  135          1826          MOV     E,L             (DE) = ADDRESS OF THAT FILE NAME
                      1827
004.136  315 260 004  1828  CDS2    CALL    CDS6.           A = DIRECTORY ENTRY LENGTH
004.141  315 101 030  1829          CALL    $DADA.
004.144  303 124 004  1830          JMP     CDS1            TRY NEXT ONE
                      1831
                      1832  *       ALL EMPTY SPOTS FOLLOWING THAT LAST NAME TO BE FLAGGED CLEAR
                      1833
004.147  353          1834  CDS3    XCHG                    (HL) = ADDRESS OF LAST FILE ENTRY
                      1835
004.150  176          1836  CDS4    MOV     A,M             (A) = ENTRY FIRST BYTE
004.151  247          1837          ANA     A
004.152  312 174 004  1838          JZ      CDS5            END OF BLOCK
004.155  107          1839          MOV     B,A             SAVE ENTRY FLAG
004.156  362 163 004  1840          JP      CDS4.5          IS NOT EMPTY OR CLEAR
004.161  066 376      1841          MVI     M,DF.CLR        IS CLEAR NOW
004.163  315 260 004  1842  CDS4.5  CALL    CDS6.           A = DIRECTORY ENTRY LENGTH
004.166  315 101 030  1843          CALL    $DADA.
004.171  303 150 004  1844          JMP     CDS4
                      1845
                      1846  *       BLOCK IS CORRECTED, WRITE BACK TO DISK
                      1847
004.174  305          1848  CDS5    PUSH    B               SAVE (B) FLAG
004.175  001 000 002  1849          LXI     B,512
004.200  052 120 041  1850          LHLD    S.SCR                                   /79.11.GC/
004.203  353          1851          XCHG                    DE = SCRATCH POINTER    /79.11.GC/
004.204  041 374 001  1852          LXI     H,DIS.SEC                               /79.11.GC/
004.207  031          1853          DAD     D               HL = POINTER TO DIS.SEC /79.11.GC/
004.210  315 211 030  1854          CALL    $HLIHL          HL = SECTOR             /79.11.GC/
004.213  076 001      1855          MVI     A,DC.WRI
004.215  315 130 040  1856          CALL    SYDD            WRITE BLOCK BACK
004.220  301          1857          POP     B
004.221  330          1858          RC                      CANT WRITE IT, FORGET IT
                      1859
                      1860  *       IF THE LAST ENTRY IN THIS BLOCK IS NOT CLEAR, MUST CONTINUE
                      1861  *       CORRECTIONS TO NEXT BLOCK
                      1862
004.222  076 376      1863          MVI     A,DF.CLR
004.224  270          1864          CMP     B
004.225  310          1865          RE                      ALL CLEAR
                      1866
004.226  052 120 041  1867          LHLD    S.SCR                                   /79.11.GC/
004.231  315 234 030  1868          CALL    $INDL                                   /79.11.GC/
004.234  376 001      1869          DW      DIS.LNK                                 /79.11.GC/
004.236  353          1870          XCHG                    HL = LINK SECTOR; DE = SCR.  /79.11.GC/
```

```
                                1871
004.237  174                   1872            MOV     A,H
004.240  265                   1873            ORA     L
004.241  310                   1874            RZ                      NO MORE TO CORRECT
                                1875
004.242  001 000 002           1876            LXI     B,512
004.245  257                   1877            XRA     A
000.000                        1878            ERRNZ   DC.REA
004.246  315 130 040           1879            CALL    SYDD            READ NEXT BLOCK
004.251  330                   1880            RC                      ERROR
                                1881
004.252  052 120 041           1882            LHLD    S.SCR           HL = SCRATCH POINTER            /79.11.GC/
004.255  303 150 004           1883            JMP     CDS4            TRY THIS ONE



                                1885    **      CDS6.   -  GET DIRECTORY ENTRY LENGTH
                                1886
004.260  345                   1887    CDS6.   PUSH    H                                               /79.11.GC/
004.261  052 120 041           1888            LHLD    S.SCR                                           /79.11.GC/
004.264  315 103 005           1889            CALL    $INDLB                                          /79.11.GC/
004.267  373 001               1890            DW      DIS.ENL                                         /79.11.GC/
004.271  341                   1891            POP     H                                               /79.11.GC/
004.272  311                   1892            RET                                                     /79.11.GC/



                                1894    **      WDO     - WAIT FOR DRIVE TO OPEN
                                1895    *
                                1896    *       WAIT UNTIL USER OPENS SELECTED DRIVE.
                                1897    *       DRIVE IS ASSUMED TO BE OPEN WHEN THERE ARE NO MORE TRANSITIONS
                                1898    *       BETWEEN NOLE DETECT, AND NO HOLE DETECT IN 200 MIL. SEC.
                                1899    *
                                1900    *
                                1901    *       ENTRY:  UNIT    = UNIT NUMBER
                                1902    *
                                1903    *       EXIT:   (PSW)   = 'Z' CLEAR IF AN ABORT IS PENDING
                                1904    *                         = 'Z' SET   IF NO ABORT
                                1905    *
                                1906    *       USES:   ALL
                                1907    *
                                1908
004.273  072 320 005           1909    WDO     LDA     UNIT
004.276  062 061 041           1910            STA     AIO.UNI
000.001                        1911            IF      DEBUG
                                1912            CALL    TRACE
                                1913            DB      'IN WDO',ENL
                                1914            ENDIF
004.301  315 205 040           1915            CALL    D.SDP           SET UP DEVICE PARAMETERS
004.304  315 213 040           1916            CALL    D.STZ           SEEK TRACK ZERO
                                1917
                                1918    *       WAIT FOR NO MORE HOLES
                                1919
004.307  001 000 000           1920    WDO2    LXI     B,0             ZERO HOLE COUNT
```

```
004.312  052 033 040  1921          LHLD    .TICCNT
004.315  021 144 000  1922          LXI     D,100
004.320  031          1923          DAD     D
004.321  315 224 030  1924          CALL    $CHL
004.324  124          1925          MOV     D,H
004.325  135          1926          MOV     E,L             (DE) = - (CURRENT TIC COUNT + 100.)
004.326  315 234 001  1927  WDO3    CALL    CAB.            CHECK ABORT
004.331  300          1928          RNZ                     AN ABORT IS PENDING
004.332  346 001      1929          ANI     DF.HD
004.334  041 372 004  1930          LXI     H,WDOA
004.337  276          1931          CMP     M
004.340  167          1932          MOV     M,A
004.341  312 345 004  1933          JZ      WDO4            NO TRANSITION
004.344  003          1934          INX     B
004.345  052 033 040  1935  WDO4    LHLD    .TICCNT
004.350  031          1936          DAD     D
004.351  174          1937          MOV     A,H
004.352  247          1938          ANA     A
004.353  372 326 004  1939          JM      WDO3            WAIT AT LEAST 200 MIL SEC.
004.356  170          1940          MOV     A,B
004.357  302 307 004  1941          JNZ     WDO2            >= 2 HOLES
004.362  171          1942          MOV     A,C
004.363  376 002      1943          CPI     2
004.365  322 307 004  1944          JNC     WDO2            >= 2 HOLES
004.370  257          1945          XRA     A               FLAG NO ABORTS
004.371  311          1946          RET                     ( ONE HOLE IS OK IN CASE WE STOPPED OVER ONE!)
                      1947
004.372  000          1948  WDOA    DB      0
```

```
004.373                 1952            XTEXT   BITS


                        1954X **        BITS    - BIT SET
                        1955X *
                        1956X *         BITS SETS THE SPECIFIED BIT IN THE ACCUMULATOR.
                        1957X *
                        1958X *         ENTRY:  A       = ORIGINAL  A
                        1959X *                 B       = NUMBER OF BIT TO SET ( 7=HIGH,...,0=LOW )
                        1960X *
                        1961X *         EXIT:   A       = ORIGINAL  A  WITH  BIT(B) SET
                        1962X *
                        1963X *         USES:   PSW
                        1964X *
                        1965X
004.373  305           1966X BITS      PUSH    B
                        1967X
004.374  365           1968X           PUSH    PSW
004.375  076 200       1969X           MVI     A,10000000B
004.377  004           1970X           INR     B
005.000  007           1971X BITS1     RLC
005.001  005           1972X           DCR     B
005.002  302 000 005   1973X           JNZ     BITS1
                        1974X
005.005  117           1975X           MOV     C,A
005.006  361           1976X           POP     PSW
005.007  261           1977X           ORA     C
                        1978X
005.010  301           1979X           POP     BC
005.011  311           1980X           RET
005.012                 1981            XTEXT   CDEHL


                        1983X **        $CDEHL - COMPARE (DE) TO (HL)
                        1984X *
                        1985X *         $CDEHL COMPARES (DE) TO (HL) FOR EQUALITY.
                        1986X *
                        1987X *         ENTRY   NONE
                        1988X *         EXIT    'Z' SET IF (DE) = (HL)
                        1989X *         USES    A,F
                        1990X
                        1991X
030.216                 1992X $CDEHL    EQU     30216A          IN H17 ROM
005.012                 1993            XTEXT   CFD
```

```
                                1995X **      $CFD - CHECK FILE DELIMITER.
                                1996X *
                                1997X *       $CFD CHECKS AN ASCII CHARACTER TO SEE IF IT IS A LEGAL FILE
                                1998X *       NAME DELIMITER. LEGAL DELIMITERS ARE
                                1999X *
                                2000X *       , = / <BLANK>  <00>
                                2001X *
                                2002X *       ENTRY   (A) = CHARACTER
                                2003X *       EXIT    'C' CLEAR IF OK
                                2004X *               'C' SET IF ERROR
                                2005X *                (A) = ERROR CODE
                                2006X *       USES    A,F
                                2007X
                                2008X
005.012  247                    2009X $CFD    ANA     A
005,013  310                    2010X         RZ                      IS 00
005.014  376 054                2011X         CPI     ','
005,016  310                    2012X         RE                      IS ,
005.017  376 075                2013X         CPI     '='
005,021  310                    2014X         RE                      IS =
005.022  376 057                2015X         CPI     '/'
005,024  310                    2016X         RE                      IS /
005.025  376 040                2017X         CPI     ' '
005,027  310                    2018X         RE                      IS / /
005.030  076 007                2019X         MVI     A,EC.IFN        ILLEGAL FILE NAME
005,032  067                    2020X         STC
005.033  311                    2021X         RET
005,034                         2022          XTEXT   CHL


                                2024X **      $CHL - COMPLEMENT (HL).
                                2025X *
                                2026X *       (HL) = -(HL)            TWO'S COMPLEMENT
                                2027X *
                                2028X *       ENTRY   NONE
                                2029X *       EXIT    NONE
                                2030X *       USES    A,F,H,L
                                2031X
                                2032X
030,224                         2033X $CHL    EQU     30224A          IN H17 ROM
005.034                         2034          XTEXT   COMP


                                2036X **      $COMP - COMPARE TWO CHARACTER STRINGS.
                                2037X *
                                2038X *       $COMP COMPARES TWO BYTE STRINGS.
                                2039X *
                                2040X *       ENTRY   (C) = COMPARE COUNT
                                2041X *               (DE) = FWA OF STRING #1
                                2042X *               (HL) = FWA OF STRING #2
                                2043X *       EXIT    'Z' CLEAR, IS MIS-MATCH
                                2044X *                (C) = LENGTH REMAINING
```

```
                        2045X *                 (DE) = ADDRESS OF MISMATCH IN STRING#1
                        2046X *                 (HL) = ADDRESS OF MISMATCH IN STRING #2
                        2047X *                 'C' SET, HAVE MATCH
                        2048X *                 (C) = 0
                        2049X *                 (DE) = (DE) + (0C)
                        2050X *                 (HL) = (HL) + (0C)
                        2051X *         USES    A,F,C,D,E,H,L
                        2052X
                        2053X
030.060                 2054X $COMP     EQU     30060A          IN H17 ROM
005.034                 2055            XTEXT   CRLF


                        2057X **        $CRLF - TYPE CARRIAGE RETURN/ LINE FEED
                        2058X *
                        2059X *         $CRLF IS USED TO GENERATE PADDED CRLF'S.
                        2060X *
                        2061X *         ENTRY   NONE
                        2062X *         EXIT    (A) = 0
                        2063X *         USES    A,F
                        2064X
                        2065X
005.034  076 012       2066X $CRLF     MVI     A,NL
005.036  377 002       2067X           DB      SYSCALL,,SCOUT
005.040  257           2068X           XRA     A
005.041  311           2069X           RET
005.042                 2070            XTEXT   DADA2


                        2072X **        $DADA. - ADD (0,A) TO (H,L)
                        2073X *
                        2074X *         ENTRY   NONE
                        2075X *         EXIT    (HL) = (HL) + (0A)
                        2076X *         USES    A,F,H,L
                        2077X
                        2078X
030.101                 2079X $DADA.    EQU     30101A          IN H17 ROM
005.042                 2080            XTEXT   DTB


                        2082X **        $DTB - DELETE TRAILING BLANKS.
                        2083X *
                        2084X *         $DTB DELETES THE TRAILING BLANKS FROM A CODED LINE.
                        2085X *
                        2086X *         ENTRY   (HL) = LINE FWA
                        2087X *         EXIT    (A) = LENGTH OF RESULT (ENCLUDING 00 TERMINATOR BYTE)
                        2088X *         USES    A,F
                        2089X
                        2090X
005.042  325            2091X $DTB      PUSH    D               SAVE (DE)
```

```
005.043  124            2092X          MOV     D,H
005.044  135            2093X          MOV     E,L             (DE) = FWA
005.045  033            2094X          DCX     D               (DE) = FWA-1
005.046  176            2095X $DTB1    MOV     A,M
005.047  043            2096X          INX     H
005.050  247            2097X          ANA     A               FIND END OF LINE
005.051  302 046 005    2098X          JNZ     $DTB1
005.054  053            2099X          DCX     H               (HL) = ADDRESS OF TERMINATING ZERO BYTE
                        2100X
                        2101X *        GOT END OF LINE. DELETE TRAILING BLANKS
                        2102X
005.055  053            2103X $DTB2    DCX     H               BACKUP ONE CHARACTER
005.056  315 216 030    2104X          CALL    $CDEHL
005.061  312 072 005    2105X          JE      $DTB3           GONE PAST FRONT OF LINE, MUST BE ALL BLANKS
005.064  176            2106X          MOV     A,M
005.065  376 040        2107X          CPI     ' '
005.067  312 055 005    2108X          JE      $DTB2           GOT BLANK
                        2109X
                        2110X *        HAVE TRIMED LINE. COMPUTE LENGTH
                        2111X
005.072  043            2112X $DTB3    INX     H
005.073  066 000        2113X          MVI     M,0             TERMINATE LINE
005.075  175            2114X          MOV     A,L
005.076  223            2115X          SUB     E               (A) = LENGTH +1 (FOR 00 BYTE)
005.077  353            2116X          XCHG
005.100  043            2117X          INX     H               (HL) = LINE FWA
005.101  321            2118X          POP     D               RESTORE (DE)
005.102  311            2119X          RET
005.103                 2120           XTEXT   HLIHL


                        2122X **       $HLIHL - LOAD HL INDIRECT THROUGH HL.
                        2123X *
                        2124X *        (HL) = ((HL))
                        2125X *
                        2126X *        ENTRY   NONE
                        2127X *        EXIT    NONE
                        2128X *        USES    A,H,L
                        2129X
030.211                 2130X $HLIHL   EQU     30211A          IN H17 ROM
005.103                 2131           XTEXT   INDL


                        2133X **       $INDL - INDEXED LOAD.
                        2134X *
                        2135X *        $INDL LOADS DE WITH THE TWO BYTES AT (HL)+DISPLACMENT
                        2136X *
                        2137X *        THIS ACTS AS AN INDEXED FULL WORD LOAD.
                        2138X *
                        2139X *        (DE) = ( (HL) + DSPLACEMENT )
                        2140X *
                        2141X *        ENTRY   ((RET)) = DISPLACMENT (FULL WORD)
```

```
                                2142X *                 (HL) = TABLE ADDRESS
                                2143X *         EXIT    TO (RET+2)
                                2144X *         USES    A,F,D,E
                                2145X
                                2146X
    030.234                     2147X $INDL     EQU     30234A          IN H17 ROM
    005.103                     2148            XTEXT   INDXX


                                2150X **        $INDLB  -  INDEXED LOAD BYTE
                                2151X *
                                2152X *         BYTE INDEXED LOAD PRIMITIVE
                                2153X *
                                2154X *         ENTRY:  HL      = BASE ADDRESS
                                2155X *                 (RET)   = FULL WORD RELOCATION
                                2156X *
                                2157X *         EXIT:   A       = ( HL + (RET) )
                                2158X *
                                2159X *         USES:   A
                                2160X *
                                2161X
    005.103  353                2162X $INDLB    XCHG                    DE = BASE
    005.104  343                2163X           XTHL                    SAVE  .DE.
    005.105  325                2164X           PUSH    D               SAVE  BASE
    005.106  305                2165X           PUSH    B               SAVE  .BC.
                                2166X
    005.107  116                2167X           MOV     C,M
    005.110  043                2168X           INX     H
    005.111  106                2169X           MOV     B,M             BC = OFFSET
    005.112  043                2170X           INX     H               HL =  .RET.
                                2171X
    005.113  353                2172X           XCHG                    HL = BASE
    005.114  011                2173X           DAD     B               HL = BASE + OFFSET
    005.115  176                2174X           MOV     A,M             A  = ( BASE + OFFSET )
    005.116  353                2175X           XCHG                    HL =  .RET.
                                2176X
    005.117  301                2177X           POP     B               RESTORE  .BC.
    005.120  321                2178X           POP     D               RESTORE  BASE
    005.121  343                2179X           XTHL                    HL = .DE. ; (SP) = .RET.
    005.122  353                2180X           XCHG                    DE = .DE. ; HL = BASE
    005.123  311                2181X           RET


                                2183X **        $INDS   -  INDEXED STORE
                                2184X *
                                2185X *         INDEXED STORE PRIMITIVE.
                                2186X *
                                2187X *         ENTRY:  HL = BASE ADDRESS
                                2188X *                 DE = VALUE TO STORE
                                2189X *
                                2190X *         EXIT:   ( HL + (RET) ) = DE
                                2191X *
```

```
                                2192X *       USES:   NONE
                                2193X *
                                2194X
005.124  315 301 005  2195X $INDS   CALL    XCHGBC
005.127  343          2196X         XTHL                    SAVE  .BC.
005.130  325          2197X         PUSH    D
005.131  315 201 005  2198X         CALL    ILDEHL          DE = OFFSET
005.134  315 301 005  2199X         CALL    XCHGBC          BC = .RET.
005.137  353          2200X         XCHG                    DE = BASE ; HL = OFFSET
005.140  031          2201X         DAD     D               HL = BASE + OFFSET
005.141  353          2202X         XCHG
005.142  343          2203X         XTHL                    SAVE  BASE
005.143  353          2204X         XCHG                    DE = VALUE
005.144  315 206 005  2205X         CALL    ISDEHL
005.147  341          2206X         POP     H               HL = BASE
005.150  315 301 005  2207X         CALL    XCHGBC
005.153  343          2208X         XTHL                    RESTORE  .BC.
005.154  315 301 005  2209X         CALL    XCHGBC
005.157  311          2210X         RET


                                2212X **      $INDSB  -  INDEXED BYTE STORE
                                2213X *
                                2214X *       INDEXED BYTE STORE.
                                2215X *
                                2216X *       ENTRY:  A       = VALUE TO STORE
                                2217X *               HL      = BASE ADDRESS
                                2218X *               (RET)   = OFFSET
                                2219X *
                                2220X *       EXIT:   NONE
                                2221X *
                                2222X *       USES:   PSW
                                2223X *
                                2224X
005.160  353          2225X $INDSB  XCHG                    DE = BASE
005.161  343          2226X         XTHL                    SAVE  .DE.
005.162  325          2227X         PUSH    D               SAVE  BASE
005.163  305          2228X         PUSH    B               SAVE  .BC.
                       2229X
005.164  116          2230X         MOV     C,M
005.165  043          2231X         INX     H
005.166  106          2232X         MOV     B,M             BC = OFFSET
005.167  043          2233X         INX     H               HL =  .RET.
                       2234X
005.170  353          2235X         XCHG                    HL = BASE
005.171  011          2236X         DAD     B               HL = BASE + OFFSET
005.172  167          2237X         MOV     M,A             ( BASE + OFFSET ) = A
005.173  353          2238X         XCHG
                       2239X
005.174  301          2240X         POP     B               RESTORE  .BC.
005.175  321          2241X         POP     D               RESTORE  BASE
005.176  343          2242X         XTHL                    HL = .DE. ; (SP) = .RET.
005.177  353          2243X         XCHG                    DE = .DE. ; HL = BASE
005.200  311          2244X         RET
```

```
005.201                 2245          XTEXT   ILDEHL


                        2247X **      ILDEHL  -  INDEXED LOAD OF DE FROM HL
                        2248X *
                        2249X *       'DE' GET THE FULL WORD VALUE POINTED TO BY 'HL', AND 'HL' IS
                        2250X *       INCREMENTED BY TWO.
                        2251X *
                        2252X *       ENTRY:  HL      = ADDRESS OF FULL WORD VALUE
                        2253X *
                        2254X *       EXIT:   DE      = (HL)
                        2255X *               HL      = HL + 2
                        2256X *
                        2257X *       USES:   DE
                        2258X *
                        2259X
005.201  136           2260X ILDEHL  MOV     E,M
005.202  043           2261X         INX     H
005.203  126           2262X         MOV     D,M
005.204  043           2263X         INX     H
005.205  311           2264X         RET
005.206                 2265          XTEXT   ISDEHL


                        2267X **      ISDEHL  -  INDEXED STORE OF DE AT HL
                        2268X *
                        2269X *       STORE 'DE' AT THE ADDRESS POINTED TO BY 'HL', AND INCREMENT 'HL'
                        2270X *       BY 2.
                        2271X *
                        2272X *       ENTRY:  DE      = VALUE
                        2273X *               HL      = ADDRESS OF VALUE
                        2274X *
                        2275X *       EXIT:   (HL)    = DE
                        2276X *               HL      = HL + 2
                        2277X *
                        2278X *       USES:   HL
                        2279X *
                        2280X
005.206  163           2281X ISDEHL  MOV     M,E
005.207  043           2282X         INX     H
005.210  162           2283X         MOV     M,D
005.211  043           2284X         INX     H
005.212  311           2285X         RET
005.213                 2286          XTEXT   MCU
```

```
                                2288X **      MCU - MAP LOWER CASE TO UPPER CASE.
                                2289X *
                                2290X *       MCU MAPS A LOWER CASE ALPHABETIC TO UPPER
                                2291X *       CASE.
                                2292X *
                                2293X *       ENTRY   (A) = CHARACTER
                                2294X *       EXIT    (A) = CHARACTER RESULT
                                2295X *       USES    A,F
                                2296X
                                2297X
005.213  376 141               2298X $MCU    CPI     'a'
005.215  330                   2299X         RC                      NOT LOWER CASE
005.216  376 173               2300X         CPI     'z'+1
005.220  320                   2301X         RNC                     NOT LOWER CASE
005.221  326 040               2302X         SUI     'a'-'A'
005.223  311                   2303X         RET
005.224                        2304          XTEXT   MOVE


                                2306X **      $MOVE - MOVE DATA
                                2307X *
                                2308X *       $MOVE MOVES A BLOCK OF BYTES TO A NEW MEMORY ADDRESS.
                                2309X *       IF THE MOVE IS TO A LOWER ADDRESS, THE BYTES ARE MOVED FROM
                                2310X *       FIRST TO LAST.
                                2311X *
                                2312X *       IF THE MOVE IS TO A HIGHER ADDRESS, THE BYTES ARE MOVED FROM
                                2313X *       LAST TO FIRST.
                                2314X *
                                2315X *       THIS IS DONE SO THAT AN OVERLAPED MOVE WILL NOT 'RIPPLE'.
                                2316X *
                                2317X *       ENTRY   (BC) = COUNT
                                2318X *               (DE) = FROM
                                2319X *               (HL) = TO
                                2320X *       EXIT    MOVED
                                2321X *               (DE) = ADDRESS OF NEXT FROM BYTE
                                2322X *               (HL) = ADDRESS OF NEXT *TO* BYTE
                                2323X *               'C' CLEAR
                                2324X *       USES    ALL
                                2325X
                                2326X
030.252                         2327X $MOVE   EQU     30252A          IN H17 ROM
005.224                         2328          XTEXT   MOVEL


                                2330X **      $MOVEL - MOVE DATA
                                2331X *
                                2332X *       $MOVEL MOVES A BLOCK OF BYTES TO A NEW MEMORY ADDRESS.
                                2333X *       IF THE MOVE IS TO A LOWER ADDRESS, THE BYTES ARE MOVED FROM
                                2334X *       FIRST TO LAST.
                                2335X *
                                2336X *       IF THE MOVE IS TO A HIGHER ADDRESS, THE BYTES ARE MOVED FROM
                                2337X *       LAST TO FIRST.
```

```
                                2338X *
                                2339X *       THIS IS DONE SO THAT AN OVERLAPED MOVE WILL NOT 'RIPPLE'.
                                2340X *
                                2341X *       CALL    $MOVEL
                                2342X *       DW      COUNT
                                2343X *       DW      FROM
                                2344X *       DW      TO
                                2345X *
                                2346X *       ENTRY   ((SP)) = RET
                                2347X *               (RET+0) = COUNT (WORD VALUE)
                                2348X *               (RET+2) = FROM
                                2349X *               (RET+4) = TO
                                2350X *       EXIT    TO (RET+6)
                                2351X *               (DE) = ADDRESS OF NEXT FROM BYTE
                                2352X *               (HL) = ADDRESS OF NEXT *TO* BYTE
                                2353X *               'C' CLEAR
                                2354X *       USES    ALL
                                2355X
                                2356X
005.224  341                    2357X $MOVEL  POP     H               (HL) = RET
005.225  116                    2358X         MOV     C,M
005.226  043                    2359X         INX     H
005.227  106                    2360X         MOV     B,M             (BC) = COUNT
005.230  043                    2361X         INX     H
005.231  136                    2362X         MOV     E,M
005.232  043                    2363X         INX     H
005.233  126                    2364X         MOV     D,M             (DE) = FROM
005.234  043                    2365X         INX     H
005.235  325                    2366X         PUSH    D               ((SP)) = FROM
005.236  136                    2367X         MOV     E,M
005.237  043                    2368X         INX     H
005.240  126                    2369X         MOV     D,M             (DE) = TO
005.241  043                    2370X         INX     H
005.242  343                    2371X         XTHL                    ((SP)) = RET, (HL) = FROM
005.243  353                    2372X         XCHG                    (DE) = FROM , (HL) = TO
005.244  303 252 030            2373X         JMP     $MOVE           MOVE IT
005.247                         2374          XTEXT   SOB


                                2376X **      $SOB - SKIP OVER BLANKS.
                                2377X *
                                2378X *       $SOB IS CALLED TO SKIP AN ARBITRARILY LONG STRING OF BLANKS AND TABS.
                                2379X *
                                2380X *       ENTRY   (HL) = FWA OF (POSSIBLE) BLANK STRING
                                2381X *       EXIT    (HL) = LWA+1 OF BLANK STRING (UNCHANGED IF NO BLANKS)
                                2382X *               (A) = FIRST NON-BLANK, NON-TAB CHARACTER EEN
                                2383X *       USES    A,F,H,L
                                2384X
                                2385X
005.247  053                    2386X $SOB    DCX     H               PRE-DECREMENT
005.250  043                    2387X $SOB1   INX     H
005.251  176                    2388X         MOV     A,M
005.252  376 040                2389X         CPI     ' '
005.254  312 250 005            2390X         JE      $SOB1           GOT BLANK
```

```
005.257  376 011        2391X           CPI     TAB
005.261  312 250 005    2392X           JE      $SOB1           GOT TAB
005.264  311            2393X           RET
000.001                 2394            IF      DEBUG
                        2395            XTEXT   TRACE
                        2396            ENDIF
005.265                 2397            XTEXT   TYPCC


                        2399X **        $TYPCC - TYPE A CHARACTER STRING BY COUNT.
                        2400X *
                        2401X *         $TYPCC TYPES A STRING OF CHARACTERS. THE CALLER SUPPLIES
                        2402X *         THE CHARACTER ADDRESS AND COUNT.
                        2403X *
                        2404X *         ENTRY   (HL) = ADDRESS
                        2405X *                 (A) = COUNT
                        2406X *         EXIT    (HL) = LAST CHARACTER ADDRESS+1
                        2407X *         USES    A,F,H,L
                        2408X
                        2409X
005.265                 2410X $TYPCC    EQU     *
005.265  247            2411X           ANA     A
005.266  310            2412X           RZ                      NOTHING TO TYPE
005.267  365            2413X           PUSH    PSW             SAVE COUNT
005.270  176            2414X           MOV     A,M             (A) = CHARACTER
005.271  043            2415X           INX     H
005.272  377 002        2416X           DB      SYSCALL,.SCOUT
005.274  361            2417X           POP     PSW
005.275  075            2418X           DCR     A
005.276  303 265 005    2419X           JMP     $TYPCC
005.301                 2420            XTEXT   TYPTX


                        2422X **        $TYPTX - TYPE TEXT.
                        2423X *
                        2424X *         $TYPTX IS CALLED TO TYPE A BLOCK OF TEXT ON THE SYSTEM CONSOLE.
                        2425X *
                        2426X *         IMBEDDED ZERO BYTES INDICATE A CARRIAGE RETURN LINE FEED.
                        2427X *         A BYTE WITH THE 200Q BIT SET IS THE LAST BYTE IN THE MESSAGE.
                        2428X *
                        2429X *         ENTRY   (RET) = TEXT
                        2430X *         EXIT    TO (RET+LENGTH)
                        2431X *         USES    A,F
                        2432X
                        2433X
031.136                 2434X $TYPTX    EQU     31136A          IN H17 ROM
                        2435X
031.144                 2436X $TYPTX.   EQU     31144A          IN H17 ROM
005.301                 2437            XTEXT   UDD
```

```
                        2439X **      $UDD - UNPACK DECIMAL DIGITS.
                        2440X *
                        2441X *       UDD CONVERTS A 16 BIT VALUE INTO A SPECIFIED NUMBER OF
                        2442X *       DECIMAL DIGITS. THE RESULT IS ZERO FILLED.
                        2443X *
                        2444X *       ENTRY   (B,C) = ADDRESS VALUE
                        2445X *               (A) = DIGIT COUNT
                        2446X *               (H,L) = MEMORY ADDRESS
                        2447X *       EXIT    (HL) = (HL) + (A)
                        2448X *       USES    ALL
                        2449X
                        2450X
    031.157             2451X $UDD    EQU     31157A          IN H17 ROM
    005.301             2452          XTEXT   WER


                        2454X **      $WER - WRITE ENABLE RAM.
                        2455X *
                        2456X *       $WER IS CALLED TO ENABLE WRITTING TO THE H17 CONTROLLER'S
                        2457X *       RAM AREA.
                        2458X *
                        2459X *       ENTRY   NONE
                        2460X *       EXIT    NONE
                        2461X *       USES    NONE
                        2462X
                        2463X
    031.241             2464X $WER    EQU     31241A          IN H17 ROM


                        2466X **      $WDR - WRITE DISABLE RAM.
                        2467X *
                        2468X *       $WDR IS CALLED TO DISABLE WRITTING TO THE H17 CONTROLLER'S
                        2469X *       RAM AREA.
                        2470X *
                        2471X *       ENTRY   NONE
                        2472X *       EXIT    NONE
                        2473X *       USES    NONE
                        2474X
                        2475X
    031.222             2476X $WDR    EQU     31222A          IN H17 ROM
    005.301             2477          XTEXT   XCHGBC


                        2479X **      XCHGBC  -  XCHG BC
                        2480X *
                        2481X *       EXCHANGE THE 'BC' REGISTER PAIR WITH THE 'HL' REGISTER PAIR.
                        2482X *
                        2483X *       ENTRY:  BC      = ORIGINAL BC
                        2484X *               HL      = ORIGINAL HL
                        2485X *
```

```
                        2486X *         EXIT:   BC      = ORIGINAL HL
                        2487X *                 HL      = ORIGINAL BC
                        2488X *
                        2489X *         USES:   BC,HL
                        2490X *
                        2491X
     005,301  365       2492X XCHGBC    PUSH    PSW
     005,302  170       2493X           MOV     A,B
     005,303  104       2494X           MOV     B,H
     005,304  147       2495X           MOV     H,A
     005,305  171       2496X           MOV     A,C
     005,306  115       2497X           MOV     C,L
     005,307  157       2498X           MOV     L,A
     005,310  361       2499X           POP     PSW
     005,311  311       2500X           RET
     005,312            2501            XTEXT   ZERO


                        2503X **        $ZERO - ZERO MEMORY
                        2504X *
                        2505X *         $ZERO ZEROS A BLOCK OF MEMORY.
                        2506X *
                        2507X *         ENTRY   (HL) = ADDRESS
                        2508X *                 (B) = COUNT
                        2509X *         EXIT    (A) = 0
                        2510X *         USES    A,B,F,H,L
                        2511X
                        2512X
     031,212            2513X $ZERO     EQU     31212A          IN H17 ROM
     005,312            2514            XTEXT   ZEROS


                        2516X **        8 CONSTANT ZERO BYTES.
                        2517X
     031,320            2518X $ZEROS    EQU     31320A          IN H17 ROM
```

```
                        2521  ***     DATA FOR MOUNT & DISMOUNT
                        2522  *
                        2523
                        2524
005.312 104 105 126     2525  DEVNAME DB      'DEV',0,ENL
                        2526
005.320 000             2527  UNIT    DB      0               DEVICE UNIT NUMBER
005.321 000 000         2528  DEVTAB  DW      0               DEVICE TABLE ADDRESS
005.323 000 000         2529  UNTTAB  DW      0               DEVICE UNIT TABLE ADDRESS
                        2530
027.000                 2531  LABEL   EQU     S.GRT+256       USE ROM/RAM SCRATCH AREA FOR LABEL INFORMATION
                        2532
027.000                 2533  PGTA    EQU     S.GRT+256       WORK BUFFER FOR GRT (256 BYTES)
```

```
                                2536  ********************************************************************************
                                2537  *                                                                              *
                                2538  *       NOTE: THIS OVERLAY, AND ITS RELOCATION TABLE MUST USE LESS THAN        *
                                2539  *       SB.OVMX BYTES.                                                         *
                                2540  *                                                                              *
                                2541  *       THE SIZE OF THE RELOCATION TABLE CANNOT BE KNOWN AT ASSEMBLY TIME,     *
                                2542  *       SO THE '450' FIGURE USED BELOW IS APPROX, AND MUST BE WATCHED!         *
                                2543  *                                                                              *
                                2544  ********************************************************************************
                                2545
005.325  103 107               2546          DW      'GC'              DUMY AREA  FOR UNWANTED RELOCATIONS
005.327  000 000 000           2547          DB      0,0,0,0,0,0,0,0         PATCH AREA
005.337  000 000 000           2548          DB      0,0,0,0,0,0,0,0
005.347  000 000 000           2549          DB      0,0,0,0,0,0,0,0
005.357  000 000 000           2550          DB      0,0,0,0,0,0,0,0
005.367  000 000 000           2551          DB      0,0,0,0,0,0,0,0
005.377  000 000 000           2552          DB      0,0,0,0,0,0,0,0
                                2553
373.227                         2554          ERRPL   *+400-SB.OVMX   TOO LARGE
                                2555
                                2556
                                2557
                                2558          LON     G
                                2559
006.007  012 000 024           2560          END     HOSOVL2
         000 027 000
         046 000 050
         000 052 000
         054 000 056
         000 061 000
         066 000 072
         000 075 000
         115 000 121
         000 124 000
         151 000 154
         000 157 000
         172 000 175
         000 203 000
         206 000 211
         000 214 000
         217 000 222
         000 245 000
         251 000 256
         000 265 000
         273 000 302
         000 317 000
         327 000 332
         000 367 000
         373 000 376
         000 010 001
         026 001 033
         001 036 001
         041 001 052
         001 055 001
         062 001 066
         001 073 001
```

```
151 001 161
001 166 001
206 001 214
001 217 001
225 001 231
001 237 001
245 001 264
001 270 001
275 001 302
001 307 001
316 001 323
001 330 001
335 001 343
001 351 001
360 001 366
001 372 001
376 001 000
002 004 002
013 002 020
002 046 002
061 002 071
002 104 002
112 002 117
002 126 002
145 002 156
002 165 002
176 002 216
002 264 002
274 002 300
002 302 002
305 002 315
002 325 002
330 002 333
002 366 002
377 002 022
003 025 003
037 003 061
003 071 003
076 003 103
003 121 003
132 003 142
003 161 003
176 003 222
003 240 003
245 003 256
003 264 003
270 003 273
003 277 003
311 003 320
003 325 003
330 003 333
003 337 003
347 003 352
003 357 003
371 003 377
003 003 004
```

```
                054 004 057
                004 063 004
                111 004 127
                004 132 004
                137 004 145
                004 153 004
                157 004 164
                004 172 004
                256 004 265
                004 274 004
                327 004 335
                004 342 004
                354 004 360
                004 366 004
                003 005 052
                005 062 005
                070 005 125
                005 132 005
                135 005 145
                005 151 005
                155 005 255
                005 262 005
                277 005 000
                000
ASSEMBLY COMPLETE
 2560 STATEMENTS
    0 ERRORS DETECTED
10862 BYTES FREE
```

```
$CDEHL  030216  1992E  2104
$CFD    005012' 2009L
$CHL    030224  1924   2033E
$COMP   030060  1404   2054E
$CRLF   005034' 1492   2066L
$DADA.  030101   861   1401   1706   1829   1843   2079E
$DTB    005042' 1490   2091L
$DTB1   005046' 2095L  2098
$DTB2   005055' 2103L  2108
$DTB3   005072' 2105   2112L
$HLIHL  030211  1409   1854   2130E
$INDL   030234  1079   1082   1606   1619   1630   1642   1663   1730   1733   1868   2147E
$INDLB  005103' 1554   1660   1681   1889   2162L
$INDS   005124' 1535   1540   1633   2195L
$INDSB  005160' 1566   2225L
$MCU    005213' 1264   1269   2298L
$MOVE   030252  2327E  2373
$MOVEL  005224' 1302   1481   2357L
$SOB    005247' 1047   1141   1258   2386L
$SOB1   005250' 2387L  2390   2392
$TYPCC  005265' 1168   1491   2410E  2419
$TYPTX  031136  1164   1169   2434E
$TYPTX. 031144  2436E
$UDD    031157  1480   2451E
$WDR    031222  2476E
$WER    031241   997   1448   1597   2464E
$ZERO   031212  2513E
$ZEROS  031320  2518E
.ABUSS  040024   125E
.ALARM  002136    98E
.ALEDS  040013   123E
.CHFLG  000060   275L
.CLEAR  000055   272L
.CLEARA 000056   273L
.CLOSE  000046   265L
.CLRCO  000007   249L
.CONSL  000006   248L
.CRC    002347   106E
.CRCSUM 040027   126E
.CTC    002172   100E
.CTLC   000041   260L
.CTLFLG 040011   122E
.DECODE 000053   270L
.DELET  000050   267L
.DISMT  000061   276L
.DLEDS  040021   124E
.DLY    000053    95E
.DMNMS  000203   287L   881
.DMOUN  000201   285L   875
.DOD    003122   109E
.DODA   003356   111E
.DSPMOD 040007   120E
.DSPROT 040006   119E
.DUMP   001374    97E
.ERROR  000057   274L
.EXIT   000000   242L
.HORN   002140    99E
.IDENT  000000    94E
```

```
.IOWRK  040002  117E
.LINK   000040  259L
.LOAD   001267   96E
.LOADD  000062  277L
.LOADO  000010  250L
.MFLAG  040010  121E
.MONMS  000202  286L  878
.MOUNT  000200  284L  847  872  872  875  878  881  884
.NAME   000054  271L
.OPENC  000045  264L
.OPENR  000042  261L
.OPENU  000044  263L
.OPENW  000043  262L
.PCHL   002264  102E
.POSIT  000047  266L
.PRINT  000003  245L  1484  1486  1488
.RCK    003260  110E
.READ   000004  246L
.REGI   040005  118E
.REGPTR 040035  129E
.RENAM  000051  268L
.RESET  000204  288L  884
.RNB    002331  105E
.RNP    002325  104E
.SCIN   000001  243L
.SCOUT  000002  244L  2067  2416
.SETTP  000052  269L
.SRS    002265  103E
.START  040000  116E
.SYSRES 000012  252L
.TICCNT 040033  128E  1921  1935
.TPERR  002205  101E
.TPERRX 040031  127E
.UIVEC  040037  130E
.VERS   000011  251L
.WNB    003024  108E
.WNP    003017  107E
.WRITE  000005  247L
ABS.COD 000010  537L
ABS.ENT 000006  535L
ABS.ID  000000  531L
ABS.LDA 000002  533L
ABS.LEN 000004  534L
AIO.CGN 041047  805L
AIO.CHA 041116  820L
AIO.CNT 041111  816L
AIO.CSI 041050  806L
AIO.DDA 041041  801E
AIO.DES 041055  810L
AIO.DEV 041057  811L
AIO.DIR 041062  814L
AIO.DTA 041053  809L
AIO.EOF 041113  818L
AIO.EOM 041112  817L
AIO.FLG 041043  802L
AIO.GRT 041044  803L
AIO.LGN 041051  807L
AIO.LSI 041052  808L
```

```
AIO.SPB 041046   804L
AIO.TFP 041114   819L
AIO.UNI 041061   812L   1064   1445   1910
AIO.VEC 041040   800L
BELL    000007    25E   1165
BITS    004373'  1223   1311   1966L
BITS1   005000'  1971L  1973
BKSP    000010    27E
BOOT.F  000001   780E
C.DSYN  000375   226E
C.STX   000002    29E
C.SYN   000026    28E
CAB     001203'   995   1197L
CAB.    001234'  1201   1208   1217L  1927
CAB1    001205'  1201L  1204   1214
CAB2    001216'  1208L  1211
CB.CLI  000100    64E    79
CB.MTL  000040    63E
CB.SPK  000200    65E
CB.SSI  000020    62E
CDB.H84 000001   723E
CDB.H85 000000   722E
CDM     001263'   982   1049   1258L
CDM2    002025'  1266   1271   1275   1277   1285   1318L
CDMA    002031'  1303   1322L
CDMB    002033'  1282   1323L
CDS     004105'  1002   1804L
CDS1    004124'  1821L  1830
CDS2    004136'  1824   1828L
CDS3    004147'  1823   1834L
CDS4    004150'  1836L  1844   1883
CDS4.5  004163'  1840   1842L
CDS5    004174'  1838   1848L
CDS6.   004260'  1828   1842   1887L
CDT     002035'  1107   1338L
CDT1    002040'  1340L  1364
CDT2    002073'  1351   1358L
CDT3    002077'  1345   1356   1362L
CEF     031354    154E
CO.FLG  000001    699E
COF     002106'  1059   1379L
COF1    002135'  1394L  1412
COF2    002167'  1399   1407L
COF3    002203'  1406   1417L
COFA    002212'  1382   1385   1389   1403   1424L
CR      000015    21E
CS.FLG  000200   700E
CSL.CHR 000001   677E
CSL.ECH 000200   675E
CSL.WRP 000002   676E
CTLA    000001    36E
CTLB    000002    37E
CTLC    000003    38E
CTLD    000004    39E
CTLO    000017    40E
CTLP    000020    41E
CTLQ    000021    42E
CTLS    000023    43E
```

```
CTLZ     000032      44E
CTP.2SB  000010     685E
CTP.BKM  000002     686E
CTP.BKS  000200     682E
CTP.MLI  000040     683E
CTP.MLO  000020     684E
CTP.TAB  000001     687E
D.ABORT  040141     628L
D.CDE    040160     633L
D.CON    040110     601L
D.DLY    040235     648L
D.DTS    040163     634L
D.ERRT   040232     647L
D.LPS    040177     638L
D.MAI    040171     636L
D.MAO    040174     637L
D.MOUNT  040133     626L
D.RAM    040240     604L
D.RDB    040202     639L
D.READ   040147     630L
D.READR  040152     631L
D.SDP    040205     640L    1915
D.SDT    040166     635L
D.STS    040210     641L
D.STZ    040213     642L    1916
D.SYDD   040130     625L
D.UDLY   040216     643L
D.VEC    040130     603L     623
D.WNB    040227     646L
D.WRITE  040155     632L
D.WSC    040221     644L
D.WSP    040224     645L
D.XIT    040144     629L
D.XOK    040136     627L
DC.ART   000007     438L    1446
DC.CLO   000006     437L
DC.LOD   000011     440L
DC.MAX   000012     441L
DC.MOV   000010     439L    1526
DC.OPR   000003     434L
DC.OPU   000005     436L
DC.OPW   000004     435L
DC.REA   000000     431L    1653    1760    1812    1878
DC.RER   000002     433L    1452
DC.WRI   000001     432L    1087    1549    1738    1855
DCA      032002     156E
DDF.BOL  000011     501E
DDF.BOO  000000     500L
DDF.LAB  000011     502L    1451
DDF.RGT  000012     503L    1612
DDF.USR  000014     504L
DEBUG    000001       1E     908     945     978    1043    1259    1441    1471    1512    1598    1748    1805
                   1911    2394
DEV.DDA  000004     323L
DEV.DVG  000016     335L
DEV.DVL  000014     334L
DEV.FLG  000006     324L
DEV.JMP  000003     322L
```

```
DEV.MNU 000011      331L
DEV.NUM 000010      330L      1306
DEV.NAM 000000      314L
DEV.RES 000002      318L      1347
DEV.SPG 000007      329L
DEV.UNT 000012      332L      1295
DEVELEN 000017      337E      1362
DEVNAME 005312'     1303      1482     2525L
DEVTAB  005321'     1294      1305     2528L
DF.CLR  000376      357E      1675     1841     1863
DF.DI   000040      202E
DF.DSO  000002      198E      1224
DF.DS1  000004      199E      1225
DF.DS2  000010      200E      1226
DF.EMP  000377      356E      1672
DF.HD   000001      192E      1203     1210     1929
DF.MO   000020      201E      1227
DF.SD   000010      195E
DF.ST   000100      203E
DF.TO   000002      193E
DF.WB   000001      197E
DF.WP   000004      194E
DF.WR   000200      204E
DIF.CNT 000020      557E
DIF.LOC 000100      555E
DIF.SYS 000200      554E
DIF.WP  000040      556E
DIR.ALD 000025      372L
DIR.CLU 000015      365L
DIR.CRD 000023      371L
DIR.EXT 000010      360L
DIR.FGN 000020      368L      1682
DIR.FLG 000016      366L
DIR.LGN 000021      369L
DIR.LSI 000022      370L
DIR.NAM 000000      359L
DIR.PRO 000013      361L
DIR.VER 000014      362L
DIRELEN 000027      374E       382      419      814
DIRIDL  000015      363E
DIS.ENL 001373      386L      1661     1890
DIS.ENT 000000      381E
DIS.LNK 001376      388L      1634     1643     1869
DIS.SEC 001374      387L      1664     1852
DM.MR   000000       69E
DM.MW   000001       70E
DM.RR   000002       71E
DM.RW   000003       72E
DMO1    000264'     1052      1059L
DMO2    000331'     1066      1078L
DMO4    001017'     1072      1076     1114L
DMOA    000127'      951       955L     1097
DMONMS  000244'      882       949     1042E
DMOUN   000114'      876       944E    1156
DP.DC   000177      190E      1228     1231
DR.IM   000001      319E      1350     1353
DR.PR   000002      320E      1353
DREAD   031256      144E
```

```
DT.CR   000002   326E
DT.CW   000004   327E    1561   1564
DT.DD   000001   325E
DV.EL   000000   315E    1343
DV.NU   000001   316E    1344   1359
DWRITE  031253   142E
EC.CNA  000004   451L
EC.DDA  000027   470L
EC.DIF  000017   462L
EC.DIW  000035   476L
EC.DNI  000045   484L    1573
EC.DNR  000046   485L    1454
EC.DNS  000005   452L    1318
EC.DSC  000047   486L    1746
EC.EOF  000001   448L
EC.EOM  000002   449L
EC.FAO  000031   472L
EC.FAP  000026   469L
EC.FL   000030   471L
EC.FNF  000014   459L
EC.FNO  000011   456L
EC.FNR  000034   475L
EC.FOD  000043   482L    1419
EC.FUC  000013   458L
EC.ICN  000016   461L
EC.IDN  000006   453L
EC.IFC  000020   463L
EC.IFN  000007   454L    2019
EC.ILC  000003   450L     853
EC.ILO  000040   479L
EC.ILR  000012   457L
EC.ILV  000037   478L
EC.IOI  000052   489L
EC.IS   000032   473L
EC.NCV  000050   487L
EC.NEM  000021   464L
EC.NOS  000051   488L
EC.NPM  000044   483L    1116
EC.NRD  000010   455L
EC.NVM  000042   481L    1053   1159
EC.OTL  000053   490L
EC.RF   000022   465L
EC.UNA  000036   477L
EC.UND  000015   460L
EC.UUN  000033   474L
EC.VPM  000041   480L     988
EC.WF   000023   466L
EC.WP   000025   468L    1014
EC.WPV  000024   467L
ENL     000212    34E    1170   2525
ERR.FNO 031344   150E
ERR.ILR 031350   152E
ERRILC  000015   853L
ESC     000033    32E
FF      000014    35E
FFB     032133   158E
FFL     032205   160E
FT.ABS  000000   182E
```

```
FT.BAC  000003      185E
FT.DD   000001      399E
FT.OR   000002      400E
FT.OU   000010      402E
FT.OW   000004      401E
FT.PIC  000001      183E
FT.REL  000002      184E
GETLAB  002215'      915     1440L   1511
GETLAB. 002233'     1095     1449L
HDS.SPG 000002      497E
HDS1    000014'     852L      858
HDS2    000021'     848       857L
HDSOVL2 000006'     846L
HDSVEC  000046'     859       870L    872     875     878     881     884     887
HDSVECL 000005      857       887E
I.CONFL 000004      702E      703
I.CONTY 000001      689E      690
I.CONWI 000003      695E      696
I.CSLMD 000000      679E
I.CUSOR 000002      692E      693
ILDEHL  005201'     2198      2260L
IMM     002254'      918       952     1470L
IMMA    002335'     1483      1494L
IMMB    002344'     1478      1495L
IMMC    002351'     1482      1487    1496L
IOC.CGN 000010      407L
IOC.CSI 000011      408L
IOC.DDA 000002      396L      403     417
IOC.DES 000016      414L
IOC.DEV 000020      415L      1167    1391    1400
IOC.DIL 000021      417E
IOC.DIR 000023      419L
IOC.DRL 000010      411E
IOC.DTA 000014      413L
IOC.FLG 000004      398L      411     1395    1400
IOC.GRT 000005      405L
IOC.LGN 000012      409L
IOC.LNK 000000      395L      1408
IOC.LSI 000013      410L
IOC.SPG 000007      406L
IOC.SQL 000003      403E
IOC.UNI 000022      416L      1167    1391
IOCCTD  000001      423E
IOCELEN 000052      421E
IP.PAD  000360       55E
ISDEHL  005206'     2205      2281L
LAB.DAT 000000      516E
LAB.DIS 000003      512L      1532
LAB.GRT 000005      513L      1538
LAB.IND 000001      511L
LAB.LAB 000021      523L       524    1489
LAB.LBL 000074      524E
LAB.NOD 000002      518E      1521
LAB.SER 000000      510L      1470    1523
LAB.SPG 000007      514L
LAB.SYS 000001      517E
LAB.VER 000011      521L
LAB.VLT 000010      520L      1520
```

LABEL	027000	1450	1470	1489	1520	1523	1532	1538	2531E
LDO	033012	164E							
LF	000012	22E							
M.CDCA	000017	584L							
M.CDLY	000016	583L							
M.CFWA	000012	581L							
M.CIN	000006	579L							
M.CINT	000005	578L							
M.CLWA	000014	582L							
M.COUT	000010	580L							
M.CPRE	000003	576L							
M.CRUB	000004	577L							
M.CSLC	000002	575L							
M.FOX	000303	89E							
M.PAMB	000021	88E							
M.SALO	000001	574L							
M.SYSM	000000	573L							
MND	002365'	998	1511L						
MND.5	003100'	1559	1564L						
MND.6	003102'	1562	1566L						
MND2	003111'	1522	1573L						
MOU1	000166'	984	991L						
MOU2	000233'	999	1001	1003	1014L				
MOU2.5	000241'	996	1019L						
MOU3	000242'	983	1020L						
MOUA	000101'	917	922L						
MOUNNS	000147'	879	912	977L					
MOUNT	000060'	873	907E	1174					
NL	000012	33E	34	1165	1170	1496	2066		
NUL2	000000	24E							
NULL	000200	23E							
OP.CTL	000360	56E							
OP.DIG	000360	57E							
OP.SEG	000361	58E							
OVL.COD	000000	295L							
OVL.ENS	000010	300E	1068	1073					
OVL.ENT	000004	297L							
OVL.FLB	000006	298L	1068						
OVL.IN	000001	747E							
OVL.NUM	000014	749E							
OVL.RES	000002	748E	1070						
OVL.SIZ	000002	296L							
OVL.UCS	000200	750E							
OVL0	000000	306L	1068						
OVL1	000001	307L							
PDI	033145	166E							
PGT	003115'	1000	1597L						
PGT10.	004042'	1613	1624	1759L	1772				
PGT11.	004052'	1768L							
PGT12.	004067'	1769	1778L	1809					
PGT3	003206'	1641L	1711						
PGT4	003261'	1671L	1710						
PGT5	003304'	1687L	1700						
PGT6	003336'	1674	1678	1705L					
PGT7	003354'	1649	1677	1715L					
PGT8	003365'	1720L	1726						
PGT9	003375'	1722	1725L						
PGTA	027000	1623	1637	1687	2533E				

```
PGTB    004032'   1702     1741E
PGTC    004075'   1770     1783L
PGTD    004077'   1666     1701    1784L
PGTE    004101'   1662     1705    1785L
PGTERR  004035'   1697     1746L
PGTERR. 004040'   1753L
PGTF    004102'   1786L
PGTG    004103'   1609     1691    1716    1788L
PIC.COD 000006    549L
PIC.ID  000000    544L
PIC.LEN 000002    546L
PIC.PTR 000004    547L
QUOTE   000047     30E
REL     033177    170E
REL.    033175    168E
RES1    001060'   1144     1155L
RES2    001077'   1158     1164L
RES3    001165'   1160     1174L
RESA    001171'   1145     1177L
RESB    001176'   1151     1178L
RESET   001025'   885      1140E   1146    1152
ROMBOOT 030000    596E
RUBOUT  000177     26E
RUC     033257    174E
S.BAUD  040344    724L
S.BOOTF 041034    779L
S.CAADR 040333    706L     1148    1229
S.CACC  041006    763L      867
S.CCTAB 040335    707L
S.CDB   040343    721L
S.CFWA  040352    731L     1392
S.CODE  041007    764L
S.CONFL 040332    704L
S.CONTY 040327    691L
S.CONWI 040331    697L
S.CSLMD 040326    680L      690     693     696     703
S.CUSOR 040330    694L
S.DATC  040310    662L
S.DATE  040277    661L
S.DCS   041033    777L
S.DDDTA 040366    742L
S.DDGRP 040364    739L
S.DDLDA 040360    737L
S.DDLEN 040362    738L
S.DDOPC 040370    743L
S.DFWA  040354    732L     1293    1338
S.DIREA 041016    771L
S.DLINK 040346    729L
S.FASER 041013    770L
S.FCI   041021    772L
S.GRT0  024000    592E
S.GRT1  025000    593E
S.GRT2  026000    594E     2531    2533
S.GUP   041027    774L     1298
S.HIMEM 040316    664L
S.INT   040343    606L      717
S.JUMPS 041010    768L
S.MOUNT 041032    776L     1104    1109
```

```
S.OFWA  040350   730L   1067
S.OMAX  040324   670L
S.OSN   041004   759L
S.OVLE  041000   756L
S.OVLFL 040371   752L
S.OVLS  040376   755L
S.OVSTK 041035   784L
S.READ  031275   146E
S.RFWA  040356   733L
S.SCI   041024   773L
S.SCR   041120   822L   1632  1641  1659  1779  1850  1867  1882  1888
S.SDD   041010   769L
S.SOVR  041146   608L    610
S.SSN   041002   758L
S.SYSM  040320   666L
S.TIME  040312   663L
S.UCSF  040372   753L
S.UCSL  040374   754L
S.USRM  040322   668L
S.VAL   040277   605L    659
S.WRITE 031330   148E
SB.ORG  047000   135E
SB.OVMX 014000   136E   2554
STACK   042200   612E
STACKL  001032   610E
SYDD    040130   602E   1088  1447  1453  1527  1550  1654  1739  1762  1813  1856  1879
SYSCALL 000377   235E   1484  1486  1488  2067  2416
TAB     000011    31E   2391
TEE     033233   172E
UF.FCT  000100   219E
UF.RDA  000001   216E
UF.ROR  000002   217E
UF.RPE  000004   218E
UF.TBM  000200   220E
UNIT    005320'  1063   1098  1219  1280  1297  1308  1440  1909  2527L
UNT.DIS 000005   346L   1536  1631
UNT.FLG 000000   343L   1555  1567
UNT.GRT 000001   344L   1083  1607  1734
UNT.GTS 000003   345L   1080  1541  1620  1731
UNT.SIZ 000007   348E
UNITAB  005323'  1078   1299  1534  1605  1729  2529L
UO.CLK  000001    81E
UO.DDU  000002    80E
UO.HLT  000200    78E
UO.NFR  000100    79E
UP.DP   000174   210E
UP.FC   000175   211E
UP.SC   000176   213E
UP.SR   000176   214E
UP.ST   000175   212E
USERFWA 042200   613E
VERS    000026   233E
WDO     004273'  1171   1909L
WDO2    004307'  1920L  1941  1944
WDO3    004326'  1927L  1939
WDO4    004345'  1933   1935L
WDOA    004372'  1930   1948L
XCHGBC  005301'  2195   2199  2207  2209  2492L
```

23354 BYTES FREE