```
000.001                 1  PUBLIC  EQU     1               NOT PUBLIC
                        3  ***     SET - SET SYSTEM PARAMETERS.
                        4  *
                        5  *       JGL, 1/16/78 FOR *HEATH* COMPANY
                        6  *
                        7  *       COPYRIGHT 1978 BY HEATH COMPANY
                        8  *
                        9  *            G. C.,      Copyrisht 1979, 1980 By Heath Co.
                       10  *


                       12  ***     SET - SET OPTIONS.
                       13  *
                       14  *       SET     DEV:    OPTIONS
                       15  *
                       16  *
                       17  *       SET HELP        LIST GENERAL HELP OPTIONS
                       18  *
                       19  *       SET TT: BKS     TERMINAL PROCESSES BACKSPACES
                       20  *       SET TT: FF      Terminal Processes Form-Feed
                       21  *       SET TT: MLI     MAP LOWER CASE TO UPPER ON INPUT
                       22  *       SET TT: MLO     MASK LOWER CASE TO UPPER ON OUTPUT
                       23  *       SET TT: BKM     MAP BKSP TO RUBOUT
                       24  *       SET TT: TAB     SEND TAB CHARACTERS TO TERMINAL
                       25  *       SET TT: FILL CC NN      PAD CHARACTER CC WITH NN NULLS
                       26  *       SET TT: HELP    LIST SET OPTIONS FOR TT:
                       27  *
                       28  *
                       29  *       SET HDOS HELP   LIST HDOS SETTABLE PARAMETERS           /80.06.GC/
                       30  *       SET HDOS STAND-ALONE    ENABLE HDOS TO RUN STAND-ALONE
                       31
                       32
                       33  ***     IN ADDITION TO THE ABOVE 'BUILT-IN' OPTIONS, SET HAS THE
                       34  *       ABILITY TO SET OPTIONS IN DISK-RESIDENT DEVICE DRIVERS. THE
                       35  *       FORMAT FOR THE COMMAND IS:
                       36  *
                       37  *       SET dev: <OPTIONS>
                       38  *
                       39  *       WHERE 'DEV:' IS A DEVICE AND (OPTIONALLY) A UNIT NUMBER.
                       40  *
                       41  *       SET WILL LOAD THE DEVICE DRIVER INTO MEMORY, RELOCATE THE
                       42  *       'SET PREAMBLE' (THAT CODE BELOW DVD.ENT) AND CALL THE
                       43  *       PREAMBLE CODE WITH
                       44  *
                       45  *       (DE) = ADDRESS OF <OPTIONS> STRING
                       46  *       (A) = UNIT NUMBER
                       47  *
                       48  *       THE DEVICE DRIVER PREAMBLE IS RESPONSIBLE FOR ANALYZINT THE
                       49  *       OPTION STRING (WITH THE HELP OF SOME ROUTINES IN *SET*) AND
                       50  *       SETTING ITS OWN OPTION FLAGS. WHEN IT RETURNS TO SET,
                       51  *       *SET* WILL POST THE DRIVER BACK TO THE DISK.
```

```
                        53  ****    ASSEMBLY CONSTANTS
                        54
000.000                 55  CN.DVD  EQU     0       CHANNEL NUMBER FOR READING/WRITING DEVICE DRIVERS
                        56
                        57  ****
```

```
000.000                 60          XTEXT   ASCII

                        62X **      ASCII CHARACTER EQUIVALENCES.
                        63X
000.015                 64X CR      EQU     13              CARRIAGE RETURN
000.012                 65X LF      EQU     10              LINE FEED
000.200                 66X NULL    EQU     200Q            PAD CHARACTER
000.000                 67X NUL2    EQU     0
000.007                 68X BELL    EQU     7               BELL CHARACTER
000.177                 69X RUBOUT  EQU     177Q
000.010                 70X BKSP    EQU     10Q             CTL-H
000.026                 71X C.SYN   EQU     26Q             SYNC
000.002                 72X C.STX   EQU     2               STX
000.047                 73X QUOTE   EQU     47Q
000.011                 74X TAB     EQU     11Q
000.033                 75X ESC     EQU     33Q
000.012                 76X NL      EQU     12Q             NEW LINE (HDOS SYSTEMS)
000.212                 77X ENL     EQU     NL+200Q         NL + END-OF-LINE-FLAG
000.014                 78X FF      EQU     14Q             FORM FEED
000.001                 79X CTLA    EQU     01Q             CTL-A
000.002                 80X CTLB    EQU     02Q             CTL-B
000.003                 81X CTLC    EQU     03Q             CTL-C
000.004                 82X CTLD    EQU     04Q             CTL-D
000.017                 83X CTLO    EQU     17Q             CTL-O
000.020                 84X CTLP    EQU     20Q             CTL-P
000.021                 85X CTLQ    EQU     21Q             CTL-Q
000.023                 86X CTLS    EQU     23Q             CTL-S
000.032                 87X CTLZ    EQU     32Q             CTL-Z
000.000                 88          XTEXT   HOSDEF


                        90X **      HOSDEF - DEFINE HOS PARAMETER.
                        91X *
                        92X
                        93X
000.040                 94X VERS    EQU     2*16+0          VERSION 2.0
                        95X
000.377                 96X SYSCALL EQU     377Q            SYSCALL INSTRUCTION
                        97X
                        98X
000.000                 99X         ORG     0
                       100X
                       101X *       RESIDENT FUNCTIONS
                       102X
000.000                103X .EXIT   DS      1               EXIT (MUST BE FIRST)
000.001                104X .SCIN   DS      1               SCIN
000.002                105X .SCOUT  DS      1               SCOUT
000.003                106X .PRINT  DS      1               PRINT
000.004                107X .READ   DS      1               READ
000.005                108X .WRITE  DS      1               WRITE
000.006                109X .CONSL  DS      1               SET/CLEAR CONSOLE OPTIONS
000.007                110X .CLRCO  DS      1               CLEAR CONSOLE BUFFER
000.010                111X .LOADO  DS      1               LOAD AN OVERLAY
000.011                112X .VERS   DS      1               RETURN HDOS VERSION NUMBER
```

```
000.012                113X .SYSRES DS      1              PRECEDING FUNCTIONS ARE RESIDENT
                       114X
                       115X
                       116X *       *HDOSOVL0.SYS*  FUNCTIONS
                       117X
000.040                118X         ORG     40A
                       119X
000.040                120X .LINK   DS      1              LINK  (MUST BE FIRST)
000.041                121X .CTLC   DS      1              CTL-C
000.042                122X .OPENR  DS      1              OPENR
000.043                123X .OPENW  DS      1              OPENW
000.044                124X .OPENU  DS      1              OPENU
000.045                125X .OPENC  DS      1              OPENC
000.046                126X .CLOSE  DS      1              CLOSE
000.047                127X .POSIT  DS      1              POSITION
000.050                128X .DELET  DS      1              DELETE
000.051                129X .RENAM  DS      1              RENAME
000.052                130X .SETTP  DS      1              SETTOP
000.053                131X .DECODE DS      1              NAME DECODE
000.054                132X .NAME   DS      1              GET FILE NAME FROM CHANNEL
000.055                133X .CLEAR  DS      1              CLEAR CHAN
000.056                134X .CLEARA DS      1              CLEAR ALL CHANS
000.057                135X .ERROR  DS      1              LOOKUP ERROR
000.060                136X .CHFLG  DS      1              CHANGE FLAGS
000.061                137X .DISMT  DS      1              FLAG SYSTEM DISK DISMOUNTED
000.062                138X .LOADD  DS      1              LOAD DEVICE DRIVER
000.063                139X .OPEN   DS      1              Parametrized Open
                       140X
                       141X
                       142X *       *HDOSOVL1.SYS*  FUNCTIONS
                       143X
000.200                144X         ORG     200Q
                       145X
000.200                146X .MOUNT  DS      1              MOUNT  (MUST BE FIRST)
000.201                147X .DMOUN  DS      1              DISMOUNT
000.202                148X .MONMS  DS      1              MOUNT/NO MESSAGE
000.203                149X .DMNMS  DS      1              DISMOUNT/NO MESSAGE
000.204                150X .RESET  DS      1              RESET = DISMOUNT/MOUNT OF UNIT
000.205                151X .CLEAN  DS      1              Clean device
000.206                152X .DAD    DS      1              Dismount All Disks          /80.08.sc/
000.207                153          XTEXT   DIRDEF


                       155X **      DIRECTORY ENTRY FORMAT.
                       156X
000.000                157X         ORG     0
                       158X
                       159X
000.377                160X DF.EMP  EQU     377Q           FLAGS ENTRY EMPTY
000.376                161X DF.CLR  EQU     376Q           FLAGS ENTRY EMPTY, REST OF DIR ALSO CLEAR
                       162X
000.000                163X DIR.NAM DS      8              NAME
000.010                164X DIR.EXT DS      3              EXTENSION
000.013                165X DIR.PRO DS      1              PROJECT
000.014                166X DIR.VER DS      1              VERSION
```

```
000.015                167X DIRIDL  EQU     *               FILE IDENTIFICATION LENGTH
                       168X
000.015                169X DIR.CLU DS      1               CLUSTER FACTOR
000.016                170X DIR.FLG DS      1               FLAGS
000.017                171X         DS      1               RESERVED
000.020                172X DIR.FGN DS      1               FIRST GROUP NUMBER
000.021                173X DIR.LGN DS      1               LAST GROUP NUMBER
000.022                174X DIR.LSI DS      1               LAST SECTOR INDEX (IN LAST GROUP)
000.023                175X DIR.CRD DS      2               CREATION DATE
000.025                176X DIR.ALD DS      2               LAST ALTERATION DATE
                       177X
000.027                178X DIRELEN EQU     *               DIRECTORY ENTRY LENGTH
000.027                179          XTEXT   ECDEF


                       181X **      ERROR CODE DEFINITIONS.
                       182X
000.000                183X         ORG     0
000.000                184X         DS      1               NO ERROR #0
000.001                185X EC.EOF  DS      1               END OF FILE
000.002                186X EC.EOM  DS      1               END OF MEDIA
000.003                187X EC.ILC  DS      1               ILLEGAL SYSCALL CODE
000.004                188X EC.CNA  DS      1               CHANNEL NOT AVAILABLE
000.005                189X EC.DNS  DS      1               DEVICE NOT SUITABLE
000.006                190X EC.IDN  DS      1               ILLEGAL DEVICE NAME
000.007                191X EC.IFN  DS      1               ILLEGAL FILE NAME
000.010                192X EC.NRD  DS      1               NO ROOM FOR DEVICE DRIVER
000.011                193X EC.FNO  DS      1               CHANNEL NOT OPEN
000.012                194X EC.ILR  DS      1               ILLEGAL REQUEST
000.013                195X EC.FUC  DS      1               FILE USAGE CONFLICT
000.014                196X EC.FNF  DS      1               FILE NAME NOT FOUND
000.015                197X EC.UND  DS      1               UNKNOWN DEVICE
000.016                198X EC.ICN  DS      1               ILLEGAL CHANNEL NUMBER
000.017                199X EC.DIF  DS      1               DIRECTORY FULL
000.020                200X EC.IFC  DS      1               ILLEGAL FILE CONTENTS
000.021                201X EC.NEM  DS      1               NOT ENOUGH MEMORY
000.022                202X EC.RF   DS      1               READ FAILURE
000.023                203X EC.WF   DS      1               WRITE FAILURE
000.024                204X EC.WPV  DS      1               WRITE PROTECTION VIOLATION
000.025                205X EC.WP   DS      1               DISK WRITE PROTECTED
000.026                206X EC.FAP  DS      1               FILE ALREADY PRESENT
000.027                207X EC.DDA  DS      1               DEVICE DRIVER ABORT
000.030                208X EC.FL   DS      1               FILE LOCKED
000.031                209X EC.FAO  DS      1               FILE ALREADY OPEN
000.032                210X EC.IS   DS      1               ILLEGAL SWITCH
000.033                211X EC.UUN  DS      1               UNKNOWN UNIT NUMBER
000.034                212X EC.FNR  DS      1               FILE NAME REQUIRED
000.035                213X EC.DIW  DS      1               DEVICE IS NOT WRITABLE (OR WRITE LOCKED)
000.036                214X EC.UNA  DS      1               UNIT NOT AVAILABLE
000.037                215X EC.ILV  DS      1               ILLEGAL VALUE
000.040                216X EC.ILO  DS      1               ILLEGAL OPTION
000.041                217X EC.VPM  DS      1               VOLUME PRESENTLY MOUNTED ON DEVICE
000.042                218X EC.NVM  DS      1               NO VOLUME PRESENTLY MOUNTED
000.043                219X EC.FOD  DS      1               FILE OPEN ON DEVICE
000.044                220X EC.NPM  DS      1               NO PROVISIONS MADE FOR REMOUNTING MORE DISKS
```

```
000.045                 221X EC.DNI   DS      1               DISK NOT INITIALIZED
000.046                 222X EC.DNR   DS      1               DISK IS NOT READABLE
000.047                 223X EC.DSC   DS      1               DISK STRUCTURE IS CORRUPT
000.050                 224X EC.NCV   DS      1               NOT CORRECT VERSION OF HDOS
000.051                 225X EC.NOS   DS      1               NO OPERATING SYSTEM MOUNTED
000.052                 226X EC.IOI   DS      1               ILLEGAL OVERLAY INDEX
000.053                 227X EC.OTL   DS      1               OVERLAY TO LARGE
000.054                 228           XTEXT   IOCDEF


                        230X **       I/O CHANNEL DEFINITIONS.
                        231X
000.000                 232X          ORG     0
                        233X
000.000                 234X IOC.LNK  DS      2               ADDRESS OF NEXT CHANNEL, =0 IF LAST
000.002                 235X IOC.DDA  DS      2               THREAD JUMP TO DEVICE DRIVER (VIA DEV TABLE)
                        236X
000.004                 237X IOC.FLG  DS      1               FILE TYPE FLAGS
000.001                 238X FT.DD    EQU     00000001B       =1 IF DIRECTORY DEVICE
000.002                 239X FT.OR    EQU     00000010B       =1 IF OPEN FOR READ
000.004                 240X FT.OW    EQU     00000100B       =1 IF OPEN FOR WRITE
000.010                 241X FT.OU    EQU     00001000B       =1 IF OPEN FOR UPDATE
000.020                 242X FT.OC    EQU     00010000B       =1 IF OPEN FOR CHARACTER MODE   /80.02.GC/
000.003                 243X IOC.SQL  EQU     *-IOC.DDA       LENGTH OF INFO FOR SEQUENTIAL FILE (FROM IOC)
                        244X
000.005                 245X IOC.GRT  DS      2               ADDRESS OF GROUP RESERVATION TABLE
000.007                 246X IOC.SPG  DS      1               SECTORS PER GROUP, THIS DEVICE
000.010                 247X IOC.CGN  DS      1               CURRENT GROUP NUMBER
000.011                 248X IOC.CSI  DS      1               CURRENT SECTOR INDEX (IN CURRENT GROUP)
000.012                 249X IOC.LGN  DS      1               LAST GROUP NUMBER
000.013                 250X IOC.LSI  DS      1               LAST SECTOR INDEX (IN LAST GROUP)
000.010                 251X IOC.DRL  EQU     *-IOC.FLG       LENGTH OF INFO NORMALLY COPIED BACK TO
                        252X *                                THE CHANNEL TABLE
000.014                 253X IOC.DTA  DS      2               DEVICE TABLE ADDRESS FOR THIS DEVICE
000.016                 254X IOC.DES  DS      2               SECTOR NUMBER OF DIRECTORY ENTRY
000.020                 255X IOC.DEV  DS      2               DEVICE CODE
000.022                 256X IOC.UNI  DS      1               UNIT NUMBER (0-9)
000.021                 257X IOC.DIL  EQU     *-IOC.DDA       LENGTH OF INFO FOR DIRECTORY FILE (FROM IOC)
                        258X
000.023                 259X IOC.DIR  DS      DIRELEN         DIRECTORY ENTRY
                        260X
000.052                 261X IOCELEN  EQU     *               IOC ENTRY LENGTH
                        262X
000.001                 263X IOCCTD   EQU     1               INDEX OF USER CHANNEL #0 IN CHANTAB (FIRST = 0)
000.052                 264           XTEXT   HOSEQU
```

```
                    266X **     HDOS SYSTEM EQUIVALENCES.
                    267X *
                    268X
024.000             269X S.GRT0 EQU     24000A          SYSTEM AREA FOR  GRT0
025.000             270X S.GRT1 EQU     25000A          SYSTEM AREA FOR  GRT1
026.000             271X S.GRT2 EQU     26000A          SYSTEM AREA FOR  GRT2
                    272X
030.000             273X ROMBOOT EQU    30000A          ROM BOOT ENTRY
                    274X
040.100             275X        ORG     40100A          FREE SPACE FROM PAM-8
                    276X
040.100             277X        DS      8               JUMP TO SYSTEM EXIT
040.110             278X D.CON  DS      16              DISK CONSTANTS
040.130             279X SYDD   EQU     *               SYSTEM DISK ENTRY POINT
040.130             280X D.VEC  DS      24*3            SYSTEM ROM ENTRY VECTORS
040.240             281X D.RAM  DS      31              SYSTEM ROM WORK AREA
040.277             282X S.VAL  DS      36              SYSTEM VALUES
040.343             283X S.INT  DS      115             SYSTEM INTERNAL WORK AREAS
041.126             284X        DS      16
041.146             285X S.SOVR DS      2               STACK OVERFLOW WARNING
041.150             286X        DS      42200A-*        SYSTEM STACK
001.032             287X STACKL EQU     *-S.SOVR        STACK SIZE
                    288X
042.200             289X STACK  EQU     *               LWA+1 SYSTEM STACK
042.200             290X USERFWA EQU    *               USER FWA
042.200             291         XTEXT   EDCON


                    293X **     D.CON DETAILED EQUIVALENCES.
                    294X *
                    295X *      HOSEQU MUST BE MODIFIED WHEN THIS TABLE IS MODIFIED.
                    296X
040.110             297X        ORG     D.CON
                    298X
040.110             299X D.XITA DS      2               SEE SYSTEM ROM FOR DESCRIPTION
040.112             300X D.WRITA DS     1
040.113             301X D.WRITB DS     1
040.114             302X D.WRITC DS     1
040.115             303X D.MAIA DS      1
040.116             304X D.LPSA DS      1
040.117             305X D.SDPA DS      1
040.120             306X D.SDPB DS      1
040.121             307X D.STSA DS      1
040.122             308X D.STSB DS      1
040.123             309X D.WHDA DS      1
040.124             310X D.WNHA DS      1
040.125             311X D.WSCA DS      1
                    312X
040.126             313X D.ERTS DS      2               TRACK AND SECTOR OF LAST DISK ERRORS
040.130             314         XTEXT   ESVAL
```

```
                    316X **      S.VAL - SYSTEM VALUE DEFINTIONS.
                    317X *
                    318X *       THESE VALUES ARE SET AND MAINTAINED BY THE SYSTEM.
                    319X *
                    320X *       THE DECK HOSEQU MUST BE MODIFIED WHEN THIS IS MODIFIED.
                    321X
                    322X
040.277             323X         ORG     S.VAL
                    324X
040.277             325X S.DATE  DS      9               SYSTEM DATE (IN ASCII)
040.310             326X S.DATC  DS      2               CODED DATE
040.312             327X S.TIME  DS      4               TIME FROM MIDNIGHT (IN TICS)
040.316             328X S.HIMEM DS      2               HARDWARE HIGH MEMORY ADRESS+1
                    329X
040.320             330X S.SYSM  DS      2               FWA RESIDENT SYSTEM
                    331X
040.322             332X S.USRM  DS      2               LWA USER MEMORY
                    333X
040.324             334X S.OMAX  DS      2               MAX OVERLAY SIZE FOR SYSTEM
                    335X
                    336X
                    337X **      THE FOLLOWING FIVE CELLS SHOULD BE MODIFIED/READ ONLY VIA THE .CONSL SYSCALL
                    338X
000.200             339X CSL.ECH EQU     10000000B       SUPPRESS ECHO
000.004             340X CSL.RAW EQU     00000100B       Raw Mode I/O                    /80.09.sc/
000.002             341X CSL.WRP EQU     00000010B       WRAP LINES AT WIDTH
000.001             342X CSL.CHR EQU     00000001B       OPERATE IN CHARACTER MODE
                    343X
000.000             344X I.CSLMD EQU     0               S.CSLMD IS FIRST BYTE
040.326             345X S.CSLMD DS      1               CONSOLE MODE
                    346X
000.200             347X CTP.BKS EQU     10000000B       TERMINAL PROCESSES BACKSPACES
000.100             348X CTP.FF  EQU     01000000B       Terminal Processes Form-Feed    /80.09.sc/
000.040             349X CTP.MLI EQU     00100000B       MAP LOWER CASE TO UPPER ON INPUT
000.020             350X CTP.MLO EQU     00010000B       MAP LOWER CASE TO UPPER ON OUTPUT
000.010             351X CTP.2SB EQU     00001000B       TERMINAL NEEDS TWO STOP BITS
000.002             352X CTP.BKM EQU     00000010B       MAP BKSP (UPON INPUT) TO RUBOUT
000.001             353X CTP.TAB EQU     00000001B       TERMINAL SUPPORTS TAB CHARACTERS
                    354X
000.001             355X I.CONTY EQU     1               S.CONTY IS 2ND BYTE
000.000             356X         ERRNZ   *-S.CSLMD-I.CONTY
040.327             357X S.CONTY DS      1               CONSOLE TYPE FLAGS
000.002             358X I.CUSOR EQU     2               S.CUSOR IS 3RD BYTE
000.000             359X         ERRNZ   *-S.CSLMD-I.CUSOR
040.330             360X S.CUSOR DS      1               CURRENT CURSOR POSITION
000.003             361X I.CONWI EQU     3               S.CONWI IS 4TH BYTE
000.000             362X         ERRNZ   *-S.CSLMD-I.CONWI
040.331             363X S.CONWI DS      1               CONSOLE WIDTH
                    364X
000.001             365X CO.FLG  EQU     00000001B       CTL-O FLAG
000.200             366X CS.FLG  EQU     10000000B       CTL-S FLAG
                    367X
000.004             368X I.CONFL EQU     4               S.CONFL IS 5TH BYTE
000.000             369X         ERRNZ   *-S.CSLMD-I.CONFL
040.332             370X S.CONFL DS      1               CONSOLE FLAGS
                    371X
```

```
040.333                372X S.CAADR DS      2              ADDRESS FOR ABORT PROCESSING (>256 IF VALID)
040.335                373X S.CCTAB DS      4              ADDR FOR CTL-A, CTL-B, CTL-C PROCESSING
040.343                374          XTEXT   ESINT


                       376X **      S.INT - SYSTEM INTERNAL WORKAREA DEFINITIONS.
                       377X *
                       378X *       THESE CELLS ARE REFERENCED BY OVERLAYS AND MAIN CODE, AND
                       379X *       MUST THEREFORE RESIDE IN FIXED LOW MEMORY.
                       380X
                       381X
040.343                382X         ORG     S.INT
                       383X
                       384X **      CONSOLE STATUS FLAGS
                       385X
040.343                386X S.CDB   DS      1              CONSOLE DESCRIPTOR BYTE
000.000                387X CDB.H85 EQU     00000000B
000.001                388X CDB.H84 EQU     00000001B      =0 IF H8-5, =1 IF H8-4
040.344                389X S.BAUD  DS      2              [0-14]  H8-4 BAUD RATE, =0 IF H8-5
                       390X *                              [15]    =1 IF BAUD RATE => 2 STOP BITS
                       391X
                       392X **      TABLE ADDRESS WORDS
                       393X
040.346                394X S.DLINK DS      2              ADDRESS OF DATA IN HDOS CODE
040.350                395X S.OFWA  DS      2              FWA  OVERLAY  TABLE
040.352                396X S.CFWA  DS      2              FWA  CHANNEL  TABLE
040.354                397X S.DFWA  DS      2              FWA  DEVICE   TABLE
040.356                398X S.RFWA  DS      2              FWA  RESIDENT HDOS CODE
                       399X
                       400X **      DEVICE DRIVER DELAYED LOAD FLAGS
                       401X
040.360                402X S.DDLDA DS      2              DRIVER LOAD ADDRESS (HIGH BYTE=0 IF NO LOAD PENDING)
040.362                403X S.DDLEN DS      2              CODE LENGTH IN BYTES
040.364                404X S.DDGRP DS      1              GROUP NUMBER FOR DRIVER
040.365                405X         DS      1              HOLD PLACE
                       406X *S.DDSEC        DS      2              SECTOR NUMBER FOR DRIVER ( * OBSOLETE ! * )
040.366                407X S.DDDTA DS      2              DEVICE'S ADDRESS IN DEVLST +DEV.RES
040.370                408X S.DDOPC DS      1              OPEN OPCODE PENDEDING
                       409X
                       410X **      OVERLAY MANAGEMENT FLAGS
                       411X
000.001                412X OVL.IN  EQU     00000001B      IN MEMORY
000.002                413X OVL.RES EQU     00000010B      PERMINANTLY RESIDENT
000.014                414X OVL.NUM EQU     00001100B      OVERLAY NUMBER MASK
000.200                415X OVL.UCS EQU     10000000B      USER CODE SWAPPED FOR OVERLAY
                       416X
040.371                417X S.OVLFL DS      1              OVERLAY FLAG
040.372                418X S.UCSF  DS      2              FWA SWAPPED USER CODE
040.374                419X S.UCSL  DS      2              LENGTH SWAPPED USER CODE
040.376                420X S.OVLS  DS      2              SIZE OF OVERLAY CODE
041.000                421X S.OVLE  DS      2              ENTRY POINT OF OVERLAY CODE
                       422X
041.002                423X S.SSN   DS      2              SWAP AREA SECTOR NUMBER
041.004                424X S.OSN   DS      2              OVERLAY SECTOR NUMBER
```

```
                        425X
                        426X *      SYSCALL PROCESSING WORK AREAS
                        427X
041.006                 428X S.CACC  DS      1           (ACC) UPON SYSCALL
041.007                 429X S.CODE  DS      1           SYSCALL INDEX IN PROGRESS
                        430X
                        431X *      JUMPS TO ROUTINES IN RESIDENT HDOS CODE
                        432X
041.010                 433X S.JUMPS DS      0           START OF DUMP VECTORS
041.010                 434X S.SDD   DS      3           JUMP TO STAND-IN DEVICE DRIVER
041.013                 435X S.FASER DS      3           JUMP TO FATSERR (FATAL SYSTEM ERROR)
041.016                 436X S.DIREA DS      3           JUMP TO DIREAD (DISK FILE READ)
041.021                 437X S.FCI   DS      3           JUMP TO FCI (FETCH CHANNEL INFO)
041.024                 438X S.SCI   DS      3           JUMP TO SCI (STORE CHANNEL INFO)
041.027                 439X S.GUP   DS      3           JUMP TO GUP (GET UNIT POINTER)
                        440X
041.032                 441X S.MOUNT DS      1           <>0 IF THE SYSTEM DISK IS MOUNTED
041.033                 442X S.DCS   DS      1           DEFAULT CLUSTER SIZE-1
                        443X
041.034                 444X S.BOOTF DS      1           BOOT FLAGS
000.001                 445X BOOT.P  EQU     00000001B   EXECUTE PROLOGUE UPON BOOTUP
                        446X
                        447X *      STACK VALUE SAVED FOR OVERLAY SYSCALLS
                        448X
041.035                 449X S.OVSTK DS      2           VALUE OF SP UPON SYSCALLS USING OVERLAY
                        450X
041.037                 451X         DS      1           RESERVED


                        453X **     ACTIVE I/O AREA.
                        454X *
                        455X *      THE AIO.XXX AREA CONTAINS INFORMATION ABOUT THE I/O OPERATION
                        456X *      CURRENTLY BEING PERFORMED. THE INFORMATION IS OBTAINED FROM
                        457X *      THE CHANNEL TABLE, AND WILL BE RESTORED THERE WHEN DONE.
                        458X *
                        459X *      NORMALLY, THE AIO.XXX INFORMATION WOULD BE OBTAINED DIRECTLY
                        460X *      FROM VARIOUS SYSTEM TABLES VIA POINTER REGISTERS. SINCE THE
                        461X *      8080 HAS NO GOOD INDEXED ADDRESSING, THE DATA IS MANUALLY
                        462X *      COPIED INTO THE AIO.XXX CELLS BEFORE PROCESSING, AND
                        463X *      BACKDATED AFTER PROCESSING.
                        464X
041.040                 465X AIO.VEC DS      3           JUMP INSTRUCTION
041.041                 466X AIO.DDA EQU     *-2         DEVICE DRIVER ADDRESS
041.043                 467X AIO.FLG DS      1           FLAG BYTE
041.044                 468X AIO.GRT DS      2           ADDRESS OF GROUP RESERV TABLE
041.046                 469X AIO.SPG DS      1           SECTORS PER GROUP
041.047                 470X AIO.CGN DS      1           CURRENT GROUP NUMBER
041.050                 471X AIO.CSI DS      1           CURRENT SECTOR INDEX
041.051                 472X AIO.LGN DS      1           LAST GROUP NUMBER
041.052                 473X AIO.LSI DS      1           LAST SECTOR INDEX
041.053                 474X AIO.DTA DS      2           DEVICE TABLE ADDRESS
041.055                 475X AIO.DES DS      2           DIRECTORY SECTOR
041.057                 476X AIO.DEV DS      2           DEVICE CODE
041.061                 477X AIO.UNI DS      1           UNIT NUMBER (0-9)
```

```
                    478X
041.062             479X AIO.DIR DS      DIRELEN         DIRECTORY ENTRY
                    480X
041.111             481X AIO.CNT DS      1               SECTOR COUNT
041.112             482X AIO.EOM DS      1               END OF MEDIA FLAG
041.113             483X AIO.EOF DS      1               END OF FILE FLAG
041.114             484X AIO.TFP DS      2               TEMP FILE POINTERS
041.116             485X AIO.CHA DS      2               ADDRESS OF CHANNEL BLOCK (IOC.DDA)



041.120             487X S.BDA   DS      1               Boot Device Address (Setup by ROM) /80.09.sc/
041.121             488X S.SCR   DS      2               SYSTEM SCRATCH AREA ADDRESS
041.123             489          XTEXT   MTRDEF


                    491X **      HDOS MONITOR PRIVATE RAM AREA DEFINITIONS.
                    492X
000.000             493X         ORG     0
000.000             494X M.SYSM  DS      1               SYSCALL ITTERATION COUNT
000.001             495X M.SALO  DS      1               STAND-ALONE FLAG
000.002             496X M.CSLC  DS      1               LINES IN CONSOLE BUFFER
000.003             497X M.CPRE  DS      1               CONSOLE PREVIOUS CHARACTER
000.004             498X M.CRUB  DS      1               CONSOLE RUBOUT FLAG
000.005             499X M.CINT  DS      1               CONSOLE INTERRUPT FLAG
000.006             500X M.CIN   DS      2               CONSOLE CB IN POINTER
000.010             501X M.COUT  DS      2               CONSOLE CB OUT POINTER
000.012             502X M.CFWA  DS      2               CONSOLE CB FWA POINTER
000.014             503X M.CLWA  DS      2               CONSOLE CB LWA POINTER
000.016             504X M.CDLY  DS      1               CONSOLE PAD CHARACTER COUNT
000.017             505X M.CDCA  DS      2               ADDRESS OF CHARACTER BEING PADDED
000.021             506X M.SUNI  DS      1               System Unit Number          /80.05.sc/
000.022             507X M.SYDD  DS      2               Address of Raw System Driver  /80.09.sc/
000.024             508          XTEXT   DDDEF


                    510X **      DEVICE DRIVER COMMUNICATION FLAGS.
                    511X *
                    512X
000.000             513X         ORG     0
                    514X
000.000             515X DC.REA  DS      1               READ
000.001             516X DC.WRI  DS      1               WRITE
000.002             517X DC.RER  DS      1               READ REGARDLESS
000.003             518X DC.OPR  DS      1               OPEN FOR READ
000.004             519X DC.OPW  DS      1               OPEN FOR WRITE
000.005             520X DC.OPU  DS      1               OPEN FOR UPDATE
000.006             521X DC.CLO  DS      1               CLOSE
000.007             522X DC.ABT  DS      1               ABORT
000.010             523X DC.MOU  DS      1               MOUNT DEVICE
000.011             524X DC.LOD  DS      1               LOAD DEVICE DRIVER
000.012             525X DC.RDY  DS      1               Device Ready              /80.04.GC/
```

```
000.013                 526X DC.MAX  DS      1               MAXIMUM ENTRY INDEX
000.014                 527          XTEXT   FLTDEF

                        529X **      FLTDEF - DEFAULT SECTOR DEFINITIONS
                        530X
000.000                 531X         ORG     0
000.000                 532X FLT.CTY DS      1               CONSOLE TYPE FLAGS (FOR S.CONTY)
000.001                 533X FLT.CWI DS      1               CONSOLE WIDTH (FOR S.CONWI)
000.002                 534X FLT.CFC DS      1               CONSOLE FILL CHARACTERS NEEDED
000.003                 535X FLT.CRF DS      1               CONSOLE CHARACTER REQUIRING FILL(377Q IF NONE)
000.004                 536X FLT.MNC DS      1               MAXIMUM NUMBER OF I/O CHANNELS
000.005                 537X         DS      1               Hold Place (Formerly Track Delay) /80.06.sc/
000.006                 538X FLT.CDB DS      1               CONSOLE DEFINITION BYTE
000.007                 539X FLT.CBD DS      2               CONSOLE BAUD RATE
000.011                 540X FLT.BOP DS      1               BOOTUP FLAGS
000.012                 541X FLT.SAL DS      1               STAND-ALONE FLAG( != 0  => CAN GO STAND-ALONE)
                        542X
000.013                 543X FLT.PBO DS      1               Permanent Boot Options          /80.08.sc/
000.001                 544X PBO.DAT EQU     00000001B       No-Date: 0=> No Date            /80.08.sc/
000.014                 545          XTEXT   FILDEF

                        547X **      FILDEF - FILE TYPE DEFINITIONS.
                        548X *
                        549X *       DB      377Q,FT.XXX
                        550X
                        551X
000.000                 552X FT.ABS  EQU     0               ABSOLUTE BINARY
000.001                 553X FT.PIC  EQU     1               POSITION INDEPENDANT CODE
000.002                 554X FT.REL  EQU     2               RELOCATABLE CODE
000.003                 555X FT.BAC  EQU     3               COMPILED BASIC CODE
000.014                 556          XTEXT   ABSDEF

                        558X **      ABS FORMAT EQUIVALENCES.
                        559X
000.000                 560X         ORG     0
                        561X
000.000                 562X ABS.ID  DS      1               377Q = BINARY FILE FLAG
000.001                 563X         DS      1               FILE TYPE (FT.ABS)
000.002                 564X ABS.LDA DS      2               LOAD ADDRESS
000.004                 565X ABS.LEN DS      2               LENGTH OF ENTIRE RECORD
000.006                 566X ABS.ENT DS      2               ENTRY POINT
                        567X
000.010                 568X ABS.COD DS      0               CODE STARTS HERE
000.010                 569          XTEXT   PICDEF
```

```
                    571X **     PIC FORMAT EQUIVALENCES.
                    572X
000.000             573X        ORG     0
                    574X
000.000             575X PIC.ID DS      1           377Q = BINARY FILE FLAG
000.001             576X        DS      1           FILE TYPE (FT.PIC)
000.002             577X PIC.LEN DS     2           LENGTH OF ENTIRE RECORD
000.004             578X PIC.PTR DS     2           INDEX OF START OF PIC TABLE
                    579X
000.006             580X PIC.COD DS     0           CODE STARTS HERE
000.006             581         XTEXT   DVDDEF


                    583X **     DEVICE DRIVER EQUIVALENCES.
                    584X
000.307             585X DVDFLV EQU     307Q        DEVICE DRIVER FLAG VALUE
                    586X
000.006             587X        ORG     PIC.COD     STARTS AT PIC CODE AREA
                    588X
000.006             589X DVD.DVD DS     1           MUST BE DVDFLV, FLAGS TO HDOS AS DRIVER
000.007             590X DVD.CAP DS     1           DEVICE CAPABILITY FLAG
000.010             591X DVD.MUM DS     1           MOUNTED UNIT MASK
000.011             592X DVD.MNU DS     1           MAXIMUM NUMBER OF UNITS
000.012             593X DVD.UFL DS     8           UNIT SUB-CAPABILITY FLAGS FOR UNITS 0-7
000.022             594X DVD.SET DS     1           = DVDFLV IFF DRIVER WILL TAKE SET OPTIONS
000.023             595X DVD.INP DS     2           Pointer to Init Code            /80.07.sc/
000.025             596X        DS      22          RESERVED, MUST BE 0             /80.07.sc/
000.053             597X DVD.STE EQU    *           ENTRY FOR 'SET' INVOCATION
                    598X
002.000             599X DVD.ENT EQU    2000A       DRIVER ENTRY POINT (MUST BE MULT OF 256)
000.053             600         XTEXT   DDFDEF


                    602X **     DIRECTORY DEVICE FORMAT DEFINITION.             /80.09.sc/
                    603X *
                    604X *      Modified:       Sep-80
                    605X *       No longer require 2 sectors per group
                    606X *       Reserved Group Table dynamically allocated
                    607X *
                    608X
000.000             609X        ORG     0
                    610X
000.000             611X DDF.BOO DS     9           2K BOOT PROGRAM
000.011             612X DDF.BOL EQU    *           LENGTH OF BOOT
000.011             613X DDF.LAB DS     1           LABEL SECTOR
000.012             614X DDF.USR DS     0           BEGINNING OF OPEN SPACE
000.012             615         XTEXT   LABDEF
```

```
                        617X **      DISK LABEL SECTOR FORMATS.
                        618X
000.000                 619X         ORG     0
000.000                 620X LAB.SER DS      1              SERIAL NUMBER OF VOLUME
000.001                 621X LAB.IND DS      2              INITIALIZATION DATE
000.003                 622X LAB.DIS DS      2              SECTOR NUMBER OF 1ST DIRECTORY SECTOR
000.005                 623X LAB.GRT DS      2              INDEX OF GRT SECTOR
000.007                 624X LAB.SPG DS      1              SECTORS PER GROUP
                        625X
000.000                 626X LAB.DAT EQU     0              DATA VOLUME ONLY
000.001                 627X LAB.SYS EQU     1              SYSTEM VOLUME
000.002                 628X LAB.NOD EQU     2              => LAB.NOD MEANS VOLUME HAS NO DIRECTORY
                        629X
000.010                 630X LAB.VLT DS      1              VOLUME TYPE
000.011                 631X LAB.VER DS      1              VERSION OF INIT17 THAT INITED DISK
                        632X
000.012                 633X LAB.RGT DS      2              RGT sector number              /80.06.sc/
                        634X
000.014                 635X LAB.VPR EQU     *              Volume dependant data          /80.05.sc/
000.014                 636X LAB.SIZ DS      2               Volume Size (Bytes/256)       /80.05.sc/
000.016                 637X LAB.PSS DS      2               Physical Sector Size          /80.05.sc/
000.020                 638X LAB.VFL DS      1               Volume dependant Flags        /80.09.sc/
000.001                 639X VFL.NSD EQU     00000001B        Number of Sides:  1 => 2     /80.09.sc/
000.005                 640X LAB.VPL EQU     *-LAB.VPR      Length of volume dependant data /80.05.sc/
                        641X
000.000                 642X         ERRMI   5-LAB.VPL                                     /80.05.sc/
000.021                 643X         DS      5-LAB.VPL      Reserved                       /80.05.sc/
                        644X
000.021                 645X LAB.LAB DS      60             LABEL
000.074                 646X LAB.LBL EQU     *-LAB.LAB      LABEL LENGTH
000.115                 647X         DS      2              Reserved for 0 bytes           /80.09.sc/
                        648X
000.117                 649X LAB.AUX EQU     *              Auxiliary Data                 /80.09.sc/
000.117                 650X LAB.SPT DS      1               Sectors per Track             /80.09.sc/
000.001                 651X LAB.AXL EQU     *-LAB.AUX      Length of Aux. Data            /80.09.sc/


042.170                 653          ORG     USERFWA-ABS.COD
042.170  377 000       654          DB      377Q,FT.ABS
042.172  200 042       655          DW      USERFWA        LOAD ADDR
042.174  272 012       656          DW      MEML-USERFWA   LOAD LENGTH
042.176  333 042       657          DW      ENTRY          ENTRY ADDRESS
```

```
                                660  ***     FIXED ADDRESS UTILITY ROUTINES.
                                661  *
                                662  *       THESE VECTORS RESIDE AT A FIXED ADDRESS RELATIVE TO 'USERFWA'
                                663  *       AND ARE CALLED BY DEVICE DRIVER SET CODE.
                                664
                                665
000.000                         666          ERRNZ   STACK-*
042,200  000                    667          DB      0               STACK+1 MUST BE 0
                                668
                                669


                                671  ***     SNA. - SCAN TO NEXT ARGUMENT.
                                672  *
                                673  *       SNA IS CALLED TO SKIP BLANKS UNTIL THE NEXT ARGUMENT.
                                674  *
                                675  *       ENTRY   (BC) = LINE POINTER
                                676  *       EXIT    (BC) UPDATED
                                677  *               'Z' SET IF AT END OF LINE
                                678  *       USES    A,F,B,C
                                679
                                680
042.201  303 114 053            681  SNA.    JMP     SKB             SCAN NEXT ARGUMENT


                                683  ***     DCS. - DELIMIT CHARACTER STRING.
                                684  *
                                685  *       DCS SCANS A CHARACTER STRING (ANYTHING BUT BLANKS) IN
                                686  *       THE LINE.
                                687  *
                                688  *       ENTRY   (BC) = LINE POINTER
                                689  *       EXIT    (BC) UPDATED PAST STRING
                                690  *               (DE) = ADDR FIRST STRING CHARACTER
                                691  *               (HL) = ADDR LAST STRING CHARACTER
                                692  *               (A) = STRING LENGTH
                                693  *               'Z' SET IF STRING EMPTY
                                694  *       USES    ALL
                                695
                                696
042.204  303 212 052            697  DCS.    JMP     DCS


                                699  ***     CNA. - CONVERT NUMERIC ARGUMENT.
                                700  *
                                701  *       CNA. CONVERTS A NUMERIC ARGUMENT IN THE COMMAND LINE
                                702  *       TO A BINARY VALUE.
                                703  *
                                704  *       THE NUMBER MAY USE A 'B', 'O', 'Q', OR 'D' POSTRADIX.
                                705  *
                                706  *       ENTRY   (BC) = LINE POINTER
```

```
                                707  *               (A) = DEFAULT RADIX
                                708  *       EXIT    (BC) UPDATED
                                709  *               (HL) = VALUE
                                710  *               'C' CLEAR IF OK
                                711  *               'C' SET IF ERROR
                                712  *                (A) = ERROR CODE
                                713  *       USES    ALL
                                714
                                715
042.207  303 242 052           716  CNA.    JMP     DNF             DECODE NUMERIC FIELD


                                718  ***     FST. - FIND IN SERIAL TABLE
                                719  *
                                720
042.212  303 223 053           721  FST.    JMP     $FST


                                723  ***     TBLS. - TABLE SEARCH
                                724  *
                                725
042.215  303 166 054           726  TBLS.   JMP     $TBLS


                                728  ***     WTBLS. - WORD TABLE SEARCH
                                729  *
                                730
042.220  303 217 054           731  WTBLS.  JMP     $WTBLS


                                733  ***     LBD. - LOOK UP BAUD RATE
                                734  *
                                735
042.223  303 326 053           736  LBD.    JMP     $LBD


                                738  ***     SOP. - SET OPTIONS
                                739  *
                                740
042.226  303 010 054           741  SOP.    JMP     SOP
```

```
                                743  ***     PBF. - PROCESS BYTE FLAG
                                744  *
                                745
042.231 303 057 054             746  PBF.    JMP     PBF


                                748  ***     PBV. - PROCESS BYTE VALUE
                                749  *
                                750
042.234 303 076 054             751  PBV.    JMP     PBV
042.237                         752          DS      60              RESERVED
```

```
                                755  ***     MAIN ROUTINE.
                                756  *
                                757
                                758
042.333                         759  ENTRY   EQU     *
042.333  041 072 060           760          LXI     H,RMEML
042.336  377 052               761          DB      SYSCALL,.SETTP  SET TOP OF USED MEMORY
042.340  041 000 000           762          LXI     H,0
042.343  071                   763          DAD     SP
042.344  104                   764          MOV     B,H
042.345  115                   765          MOV     C,L             (BC) = COMMAND LINE
042.346  315 354 042           766          CALL    SET1            PERFORM PROCESSING
042.351  303 225 043           767          JMP     EXIT
                                768
                                769  *       GET DEVICE CODE
                                770  *
                                771  *       THIS SECTION IS CALLED FROM ABOVE. A 'RETURN' CAUSES SET TO EXIT
                                772
042.354  315 212 052           773  SET1    CALL    DCS
042.357  312 046 043           774          JZ      ERR.IDS         ILLEGAL DEVICE SPEC
042.362  176                   775          MOV     A,M
042.363  366 200               776          ORI     200Q
042.365  167                   777          MOV     M,A
042.366  041 013 043           778          LXI     H,SETA
042.371  315 223 053           779          CALL    $FST
042.374  302 215 051           780          JNZ     PDF             NOT BUILT-IN, PATCH DISK FILE AND RETURN
042.377  176                   781          MOV     A,M
043.000  315 061 031           782          CALL    $TJMP
043.003  231 043               783          DW      SETTT           TT:
043.005  364 046               784          DW      SETHLP          HELP
043.007  006 050               785          DW      SETVER          VERSION SWITCH
043.011  035 050               786          DW      SETHOS          HDOS PARAMETERS

043.013  045 043               788  SETA    DW      SETAE           END ADDRESS
043.015  001                   789          DB      1               1 BYTE DATA VALUES
                                790
043.016  124 124 272           791          DB      'TT',':'+200Q,0
043.022  124 124 060           792          DB      'TT0',':'+200Q,0
043.027  110 105 114           793          DB      'HEL','P'+200Q,1
043.034  126 105 322           794          DB      'VE','R'+200Q,2
043.040  110 104 117           795          DB      'HDO','S'+200Q,3
                                796
043.045  000                   797  SETAE   DB      0               END OF TABLE
```

```
                                    801  ***     WHEN VARIOUS ERROR CONDITIONS ARE DETECTED, CODE MAY DO A
                                    802  *       'BAIL OUT' JUMP TO ONE OF THE ERROR ABORT ADDRESSES.
                                    803  *
                                    804  *       THE STACK MAY BE UNCLEAN, FILES BE OPEN, ETC.
                                    805  *       THE ABORT ROUTINES WILL CLEAN THE STACK, ISSUE ANY APPROPRIATE MESSAGES,
                                    806  *       AND EXIT, LEAVING THE O/S TO CLEAN UP ANY OPEN FILES.


                                    808  **      ERR.IDS - ILLEGAL DEVICE SPECIFICATION
                                    809
043.046  315 136 031               810  ERR.IDS CALL    $TYPTX
043.051  007 012 111               811          DB      BELL,NL,'Illegal Device Specification',ENL
043.110  303 225 043               812          JMP     EXIT


                                    814  **      ERR.ILO - ILLEGAL OPTION
                                    815
                                    816
043.113  315 136 031               817  ERR.ILO CALL    $TYPTX
043.116  007 012 111               818          DB      BELL,NL,'Illegal Option',ENL
043.137  303 225 043               819          JMP     EXIT


                                    821  **      ERR.IOV - ILLEGAL OPTION VALUE
                                    822
                                    823
043.142  315 136 031               824  ERR.IOV CALL    $TYPTX
043.145  007 012 111               825          DB      BELL,NL,'Illegal Option Value',ENL
043.174  303 225 043               826          JMP     EXIT


                                    828  **      ERROR - HDOS RETURNED ERROR
                                    829
                                    830
043.177  365                       831  ERROR   PUSH    PSW
043.200  315 136 031               832          CALL    $TYPTX
043.203  007 012 105               833          DB      BELL,NL,'Error -',' '+200Q
043.215  361                       834          POP     PSW
043.216  046 212                   835          MVI     H,ENL
043.220  377 057                   836          DB      SYSCALL,.ERROR
043.222  303 225 043               837          JMP     EXIT


                                    839  **      EXIT - EXIT TO HDOS.
                                    840  *
                                    841  *       EXIT EXITS TO HDOS WITH THE 'ABORT' FLAG SET.
                                    842  *       THIS FORCES THE CONSOLE TO BE RECONFIGURED, IN CASE THE SET
                                    843  *       COMMAND CAUSED CONSOLE HANDLING CHANGES.
                                    844
043.225  076 001                   845  EXIT    MVI     A,1             FORCE RESET
043.227  377 000                   846          SCALL   .EXIT
```

```
                                849  **      SETTT - SET TT: OPTIONS
                                850  *
                                851
                                852
043.231  021 036 044           853  SETTT   LXI     D,TTOPRC
043.234  041 246 043           854          LXI     H,TTOTAB
043.237  315 010 054           855          CALL    SOP
043.242  332 177 043           856          JC      ERROR
043.245  311                   857          RET


043.246  035 044                859  TTOTAB  DW      TTOTABE         LWA
043.250  003                    860          DB      3               VALUE BYTES
                                861
043.251  102 113 323            862          DB      'BK','S'+200Q,CTYI,377Q-CTP.BKS,CTP.BKS
043.257  106 306 000            863          DB      'F','F'+200Q,CTYI,377Q-CTP.FF,CTP.FF            /80.09,sc/
043.264  115 114 311            864          DB      'ML','I'+200Q,CTYI,377Q-CTP.MLI,CTP.MLI
043.272  115 114 317            865          DB      'ML','O'+200Q,CTYI,377Q-CTP.MLO,CTP.MLO
043.300  102 113 315            866          DB      'BK','M'+200Q,CTYI,377Q-CTP.BKM,CTP.BKM
043.306  124 101 302            867          DB      'TA','B'+200Q,CTYI,377Q-CTP.TAB,CTP.TAB
043.314  061 123 302            868          DB      '1S','B'+200Q,CTYI,377Q-CTP.2SB,0
043.322  062 123 302            869          DB      '2S','B'+200Q,CTYI,377Q-CTP.2SB,CTP.2SB
043.330  116 117 102            870          DB      'NOBK','S'+200Q,CTYI,377Q-CTP.BKS,0
043.340  116 117 106            871          DB      'NOF','F'+200Q,CTYI,377Q-CTP.FF,0                /80.09,sc/
043.347  116 117 115            872          DB      'NOML','I'+200Q,CTYI,377Q-CTP.MLI,0
043.357  116 117 115            873          DB      'NOML','O'+200Q,CTYI,377Q-CTP.MLO,0
043.367  116 117 102            874          DB      'NOBK','M'+200Q,CTYI,377Q-CTP.BKM,0
043.377  116 117 124            875          DB      'NOTA','B'+200Q,CTYI,377Q-CTP.TAB,0
044.007  127 111 104            876          DB      'WIDT','H'+200Q,WIDI,0,0
044.017  106 111 114            877          DB      'FIL','L'+200Q,FILI,0,0
044.026  110 105 114            878          DB      'HEL','P'+200Q,TTHLPI,0,0
                                879
044.035  000                    880  TTOTABE DB      0               END OF TABLE


044.036                         882  TTOPRC  DS      0
                                883
000.000                         884  CTYI    EQU     *-TTOPRC/2
044.036  046 044                885          DW      STTCTY
                                886
000.001                         887  WIDI    EQU     *-TTOPRC/2
044.040  066 044                888          DW      STTWID
                                889
000.002                         890  FILI    EQU     *-TTOPRC/2
044.042  124 044                891          DW      STTFIL
                                892
000.003                         893  TTHLPI  EQU     *-TTOPRC/2
044.044  224 044                894          DW      TTHLP
```

```
                                896  **      STTCTY - CONSOLE TYPE FLAGS.
                                897  *
                                898
                                899
044.046                         900  STTCTY  EQU     *
044.046  126                   901          MOV     D,M
044.047  043                   902          INX     H
044.050  136                   903          MOV     E,M             (E) = VALUE
044.051  072 327 040           904          LDA     S.CONTY
044.054  242                   905          ANA     D
044.055  263                   906          ORA     E
044.056  062 327 040           907          STA     S.CONTY
044.061  076 000               908          MVI     A,FLT.CTY
044.063  303 341 052           909          JMP     SHO             SET HOS OPTIONS, AND EXIT

                                911  **      STTWID - SET WIDTH OPTION
                                912  *
                                913  *       SET TT: WIDTH NN
                                914
                                915
044.066  076 012               916  STTWID  MVI     A,10            (A) = DEFAULT BASE
044.070  315 242 052           917          CALL    DNF             DECODE NUMERIC FIELD
044.073  332 142 043           918          JC      ERR.IOV
044.076  174                   919          MOV     A,H
044.077  247                   920          ANA     A
044.100  302 142 043           921          JNZ     ERR.IOV         ILLEGAL OPTION VALUE
044.103  175                   922          MOV     A,L
044.104  376 024               923          CPI     20
044.106  332 142 043           924          JC      ERR.IOV         TOO NARROW
044.111  062 331 040           925          STA     S.CONWI
044.114  137                   926          MOV     E,A
044.115  026 000               927          MVI     D,0
044.117  076 001               928          MVI     A,FLT.CWI
044.121  303 341 052           929          JMP     SHO             SET HOS OPTION AND EXIT

                                931  **      STTFIL - FILL OPTION.
                                932  *
                                933  *       SET TT: FILL CC NN
                                934
                                935
044.124  076 012               936  STTFIL  MVI     A,10            (A) = DEFAULT BASE
044.126  315 242 052           937          CALL    DNF             DECODE NUMERIC FIELD
044.131  332 142 043           938          JC      ERR.IOV
044.134  174                   939          MOV     A,H
044.135  264                   940          ORA     H
044.136  302 142 043           941          JNZ     ERR.IOV
044.141  265                   942          ORA     L
044.142  312 142 043           943          JZ      ERR.IOV         ILLEGAL OPTION VALUE
044.145  365                   944          PUSH    PSW             SAVE CHARACTER CODE
044.146  076 012               945          MVI     A,10            (A) = DEFAULT BASE
044.150  315 242 052           946          CALL    DNF             DECODE NUMERIC FIELD
044.153  332 142 043           947          JC      ERR.IOV
```

```
044.156  174            948          MOV     A,H
044.157  267            949          ORA     A
044.160  302 142 043    950          JNZ     ERR.IOV          ILLEGAL VALUE
044.163  105            951          MOV     B,L              (B) = DELAY COUNT
                        952
                        953  *       SET IN MEMORY
                        954
044.164  052 346 040    955          LHLD    S.DLINK
044.167  021 016 000    956          LXI     D,M.CDLY
044.172  031            957          DAD     D
044.173  160            958          MOV     M,B              SET DELAY COUNT
000.000                 959          ERRNZ   M.CDCA-M.CDLY-1
044.174  043            960          INX     H
044.175  315 211 030    961          CALL    $HLIHL           (HL) = ADDRESS FOR CHAR
044.200  130            962          MOV     E,B              (E) = DELAY
044.201  361            963          POP     PSW              (A) = DELAY CHARACTER
044.202  167            964          MOV     M,A
044.203  365            965          PUSH    PSW
044.204  026 000        966          MVI     D,0
044.206  076 002        967          MVI     A,FLT.CFC
044.210  315 341 052    968          CALL    SHO              SET FILL COUNT
044.213  361            969          POP     PSW
044.214  137            970          MOV     E,A
044.215  026 000        971          MVI     D,0
044.217  076 003        972          MVI     A,FLT.CRF
044.221  303 341 052    973          JMP     SHO              SET CHAR REQUIRING FILL AND EXIT


                        976  **      TTHLP - HELP OPTION FOR DEVICE TT:
                        977  *
                        978
044.224  315 136 031    979  TTHLP   CALL    $TYPTX
044.227  012 012 123    980          DB      NL,NL,'SET Options for TT:',NL,NL
044.256  102 113 123    981          DB      'BKS             CRT terminal allows backspace characters',NL
044.334  106 106 011    982          DB      'FF              Terminal processes Form-Feed character',NL
045.007  115 114 111    983          DB      'MLI             Map lower case input to upper case',NL
045.057  115 114 117    984          DB      'MLO             Map lower case output to upper case',NL
045.130  102 113 115    985          DB      'BKM             Treat "BKSP" codes (on input) as "DELETE" ("RUBOUT")',NL
045.222  124 101 102    986          DB      'TAB             Terminal can process tab codes',NL
045.266  012            987          DB      NL
045.267  011 124 150    988          DB      '       The above options can be preceded by "NO" to negate their',NL
045.362  011 105 146    989          DB      '       Effect. (I.E.  SET TT: NOTAB )',NL
046.022  012            990          DB      NL
046.023  061 123 102    991          DB      '1SB             Use One Stop Bit for Console Terminal',NL
046.076  062 123 102    992          DB      '2SB             Use Two Stop Bits for Console Terminal',NL
046.152  127 111 104    993          DB      'WIDTH NN        Set console width to NN characters',NL
046.226  106 111 114    994          DB      'FILL CC NN      Pad occurrences of character CC with NN null',NL
046.316  011 011 143    995          DB      '                characters',NL
046.333  110 105 114    996          DB      'HELP            Type this text',NL
046.360  012 212        997          DB      NL,ENL
046.362  257            998          XRA     A                CLEAR CARRY
046.363  311            999          RET
```

```
                                1002  ***     SETHLP - PRINT HELP TEXT.
                                1003  *
                                1004
                                1005
046.364  315 136 031           1006  SETHLP  CALL    $TYPTX
046.367  012 012 107           1007          DB      NL,NL,'General Command Format:',NL
047.021  012                   1008          DB      NL
047.022  040 040 040           1009          DB      '   SET xx: opt',NL
047.041  040 040 040           1010          DB      '      xx: --  Device Name',NL
047.072  040 040 040           1011          DB      '      opt --  Desired Option',NL
047.126  012 012               1012          DB      NL,NL
047.130  106 157 162           1013          DB      'For HELP with a specific device, type:',NL
047.177  012                   1014          DB      NL
047.200  040 040 040           1015          DB      '   SET xx: HELP',NL
047.220  012 012               1016          DB      NL,NL
047.222  124 157 040           1017          DB      'To determine the version of SET, type:',NL
047.271  012                   1018          DB      NL
047.272  040 040 040           1019          DB      '   SET Ver',NL
047.305  012                   1020          DB      NL
                                1021
047.306  012                   1022          DB      NL
047.307  124 157 040           1023          DB      'To list settable *HDOS* options, type:',NL
047.357  012                   1024          DB      NL
047.360  040 040 040           1025          DB      '   SET HDOS HELP',NL
050.002  012                   1026          DB      NL
                                1027
050.003  212                   1028          DB      ENL
050.004  257                   1029          XRA     A               CLEAR CARRY
050.005  311                   1030          RET
```

```
                                1034  ***     SETVER  - SET VERSION
                                1035  *
                                1036  *       SETVER PRINTS THE VERSION OF THIS SET PROGRAM
                                1037  *
                                1038
050.006  315 136 031            1039  SETVER  CALL    $TYPTX
050.011  123 105 124            1040          DB      'SET';TAB;'Version: '
050.027  062 056 060            1041          DB      VERS/16+'0','.',VERS&00001111B+'0'
050.032  212                    1042          DB      ENL
050.033  257                    1043          XRA     A               CLEAR CARRY
050.034  311                    1044          RET
```

```
                                1047  **      SETHOS  -  SET HDOS PARAMETERS
                                1048  *
                                1049  *       SETHOS PERMITS THE SETTING OF  *HDOS*  PARAMETERS.
                                1050  *
                                1051
050.035  021 150 050           1052  SETHOS  LXI     D,HOSPRC
050.040  041 052 050           1053          LXI     H,HOSTAB
050.043  315 010 054           1054          CALL    SOP
050.046  332 177 043           1055          JC      ERROR
050.051  311                   1056          RET


050.052  147 050               1058  HOSTAB  DW      HOSTABE         END ADDRESS
050.054  004                   1059          DB      4               ONE DATA BYTE
                                1060
050.055  104 101 124           1061          DB      'DAT','E'+200Q,HOSBFI,PBO,DAT,PBO,DAT,FLT,PBO
050.065  116 117 104           1062          DB      'NODAT','E'+200Q,HOSBFI,PBO,DAT,0,FLT,PBO
                                1063
050.077  123 124 101           1064          DB      'STAND-ALON','E'+200Q,HOSSALI,1,0,0
050.116  116 117 123           1065          DB      'NOSTAND-ALON','E'+200Q,HOSSALI,0,0,0
                                1066
050.137  110 105 114           1067          DB      'HEL','P'+200Q,HELPI,0,0,0
                                1068
050.147  000                   1069  HOSTABE DB      0               END OF TABLE


050.150                         1071  HOSPRC  DS      0
050.150                         1072  .       SET     *
                                1073
000.000                         1074  HOSSALI EQU     *-.,2
050.150  170 050               1075          DW      HOSSAL
                                1076
000.001                         1077  HELPI   EQU     *-.,2
050.152  336 050               1078          DW      HOSHLP
                                1079
000.002                         1080  HOSBFI  EQU     *-.,2
050.154  156 050               1081          DW      HOSBF           Process Byte flag


                                1083  **      HOSBF   -  Set  *HDOS*  Byte flags
                                1084  *
                                1085  *       HOSBF sets a flag in HDOS that does not require
                                1086  *       immediate action in memory, such as flags which
                                1087  *       only affect boot parameters.
                                1088  *
                                1089  *       ENTRY:  HL      = Option Vector
                                1090  *
                                1091  *       EXIT:   NONE
                                1092  *
                                1093  *       USES:   ALL
```

```
                        1094  *
                        1095
050.156  176           1096  HOSBF   MOV     A,M
050.157  043           1097          INX     H
050.160  057           1098          CMA
050.161  127           1099          MOV     D,A             D  = Mask
050.162  136           1100          MOV     E,M             E  = Value
050.163  043           1101          INX     H
050.164  176           1102          MOV     A,M             A  = Byte Index
050.165  303 341 052   1103          JMP     SHO



                        1105  **      HOSSAL  -  SET  *HDOS*  STAND ALONE FLAG
                        1106  *
                        1107  *       HOSSAL SETS THE  *HDOS*  STAND-ALONE FLAG ENABLING  *HDOS*
                        1108  *       TO GO STAND ALONE UPON EXITS TO  *SYSCMD.SYS*  WHEN THE SYSTEM
                        1109  *       DISK HAS BEEN DISMOUNTED.
                        1110  *
                        1111  *
                        1112
050.170  176           1113  HOSSAL  MOV     A,M
050.171  021 001 000   1114          LXI     D,M.SALO
050.174  052 346 040   1115          LHLD    S.DLINK
050.177  031           1116          DAD     D
050.200  167           1117          MOV     M,A             SET  *SALONE*  TO 1, ( != 0  =>  SET ).
050.201  137           1118          MOV     E,A             (E) = NEW VALUE
050.202  026 000       1119          MVI     D,0             (D) = CHANGE MASK
050.204  076 012       1120          MVI     A,FLT.SAL       INDEX OF BYTE TO CHANGE
000.001                 1121          IF      PUBLIC
                        1122          ELSE
050.206  365           1123          PUSH    PSW
050.207  173           1124          MOV     A,E
050.210  247           1125          ANA     A
050.211  312 332 050   1126          JZ      HOS1
050.214  315 136 031   1127          CALL    $TYPTX
050.217  012 111 164   1128          DB      NL,'It is Now Pitch Dark.  If You Proceed, You Will Likely '
050.307  106 141 154   1129          DB      'Fall Into a Pit.',NL,BELL,ENL
050.332  361           1130  HOS1    POP     PSW
                        1131          ENDIF
050.333  303 341 052   1132          JMP     SHO             SET  *HDOS*  OPTIONS, AND EXIT



                        1134  **      HOSHLP  -  PROCESS HELP OPTION FOR PSEUDO-DEVICE  *HDOS*
                        1135  *
                        1136  *
                        1137
050.336  315 136 031   1138  HOSHLP  CALL    $TYPTX
050.341  012 012 123   1139          DB      NL,NL,'Set Options for  HDOS',NL,NL
000.001                 1140          IF      1
                        1141          DB      'AUTO-BOOT              Auto-Boot with Hard Boot',NL
                        1142          ENDIF
050.372  104 101 124   1143          DB      'DATE                   Ask User for Date',NL
```

```
000.001                 1144            IF      PUBLIC
                        1145            DB      'STAND-ALONE            Flag Stand-Alone Operation Legal',NL
                        1146            ENDIF
051.023 012             1147            DB      NL
051.024 011 124 150     1148            DB      '       The above options can be preceeded by 'NO' to',NL
051.103 011 156 145     1149            DB      '       negate their effect, (i.e. SET HDOS NODATE).',NL
051.161 012             1150            DB      NL
051.162 110 105 114     1151            DB      'HELP                   Print this Text',NL
051.211 012 212         1152            DB      NL,ENL
051.213 257             1153            XRA     A       CLEAR CARRY
051.214 311             1154            RET
```

```
                                1158  **      PDF - PATCH DISK FILE.
                                1159  *
                                1160  *       PDF IS CALLED TO SET (PATCH) A DEVICE DRIVER FILE ON THE DISK.
                                1161  *
                                1162  *       1) THE DRIVER IS FOUND, AND READ IN
                                1163  *       2) ITS FORMAT IS CHECKED
                                1164  *       3) THE 'SET PREAMBLE' IS RELOCATED
                                1165  *       4) ITS 'SET' CODE AREA IS ENTERED
                                1166  *       5) THE DRIVER, EXCEPT FOR THE SET PREAMBLE (BECAUSE IT WAS RELOCATED)
                                1167  *          IS WRITTEN BACK OUT
                                1168  *
                                1169  *       ENTRY   (DE) = ADDRESS OF 'dev:' SPECIFICATION
                                1170  *       EXIT    TO CALLER OF OK
                                1171  *               TO 'ERR.???' IF ERROR
                                1172  *       USES    ALL
                                1173
                                1174
051.215                         1175  PDF     EQU     *
051.215  032                   1176          LDAX    D               (A) = FIRST CHAR OF DEVNAME
051.216  247                   1177          ANA     A
051.217  372 046 043           1178          JM      ERR.IDS         ILLEGAL DEV SPECIFIED
051.222  062 046 052           1179          STA     PDFB
051.225  023                   1180          INX     D
051.226  032                   1181          LDAX    D               (A) = 2ND CHAR OF DEVNAME
051.227  247                   1182          ANA     A
051.230  372 046 043           1183          JM      ERR.IDS
051.233  062 047 052           1184          STA     PDFB+1
051.236  023                   1185          INX     D
051.237  032                   1186          LDAX    D               (A) = UNIT NUMBER OR ':'
051.240  356 272               1187          XRI     ':'+200Q                ASSUME ':'
051.242  312 261 051           1188          JZ      PDF1            IS :, (A) = UNIT NUMBER = 0
051.245  032                   1189          LDAX    D
051.246  326 060               1190          SUI     '0'             DECODE UNIT NUMBER
051.250  332 046 043           1191          JC      ERR.IDS         NOT A UNIT NUMBER
051.253  376 010               1192          CPI     7+1
051.255  322 046 043           1193          JNC     ERR.IDS         NOT A UNIT NUMBER
051.260  023                   1194          INX     D               POINT TO ':'
                                1195
                                1196  *       (A) = UNIT NUMBER (IN BINARY)
                                1197  *       (DE) = ADDRESS OF ':'
                                1198
051.261  365                   1199  PDF1    PUSH    PSW             SAVE UNIT NUMBER
051.262  032                   1200          LDAX    D
051.263  023                   1201          INX     D
051.264  376 272               1202          CPI     ':'+200Q
051.266  302 046 043           1203          JNE     ERR.IDS         ILLEGAL DEVICE SPECIFICATION
                                1204
                                1205  *       HAVE DEVICE NAME CRACKED OUT. FIND IN DIRECTORY
                                1206
051.271  325                   1207          PUSH    D               SAVE POINTER TO COMMAND LINE
051.272  041 042 052           1208          LXI     H,PDFA
000.000                         1209          ERRNZ   CN.DVD          USING CHANNEL 0
051.275  257                   1210          XRA     A
051.276  377 044               1211          SCALL   .OPENU          OPEN DEVICE DRIVER FILE
051.300  332 177 043           1212          JC      ERROR           REPORT ERROR
                                1213
```

```
                        1214  *       READ IN FIRST BLOCK, SEE IF DEVICE DRIVER, AND HOW LONG
                        1215
051.303  001 000 001    1216          LXI     B,256
051.306  021 072 057    1217          LXI     D,BUFF
000.000                 1218          ERRNZ   CN.DVD          USING CHANNEL 0
051.311  257            1219          XRA     A
051.312  377 004        1220          SCALL   .READ
051.314  332 177 043    1221          JC      ERROR
                        1222
051.317  072 100 057    1223          LDA     BUFF+DVD.DVD
051.322  376 307        1224          CPI     DVDFLV
051.324  312 005 052    1225          JE      PDF3            OK
051.327  315 136 031    1226          CALL    $TYPTX
051.332  007 012 104    1227          DB      BELL,NL,'Disk File DOES Not Have Proper Format',ENL
052.002  303 225 043    1228          JMP     EXIT
                        1229
                        1230  *       IS DEVICE DRIVER. SEE IF SETUP FOR 'SET' USE
                        1231
052.005  072 114 057    1232  PDF3    LDA     BUFF+DVD.SET
052.010  376 307        1233          CPI     DVDFLV
052.012  302 113 043    1234          JNE     ERR.ILO         ANY OPTION IS ILLEGAL, SINCE NOT SETUP
                        1235
                        1236  *       LOAD AND RELOCATE DRIVER
                        1237
052.015  315 055 052    1238          CALL    LDD             LOAD DEVICE DRIVER
                        1239
                        1240  *       CALL DRIVER PREAMBLE CODE
                        1241
052.020  321            1242          POP     D               (DE) = PARAMETER LIST ADDRESS
052.021  361            1243          POP     PSW             (A) = UNIT NUMBER
052.022  315 145 057    1244          CALL    BUFF+DVD.STE    ENTER AT SET POINT
052.025  332 177 043    1245          JC      ERROR           ILLEGAL OPTION
                        1246
                        1247  *       HAVE SUCCESSFULLY SET. WRITE DRIVER BACK
                        1248
052.030  315 155 052    1249          CALL    WDD             WRITE DEVICE DRIVER
000.000                 1250          ERRNZ   CN.DVD
052.033  257            1251          XRA     A               (A) = CN.DVD
052.034  377 046        1252          SCALL   .CLOSE          CLOSE FILE
052.036  332 177 043    1253          JC      ERROR
052.041  311            1254          RET                     ALL DONE
                        1255
                        1256
052.042  123 131 060    1257  PDFA    DB      'SY0:'          DEVICE DRIVER FILE NAME
052.046                 1258  PDFB    DS      2               CODE STORES NAME HERE
052.050  056 104 126    1259          DB      '.DVD',0
```

```
                                1261  **      LDD - LOAD DEVICE DRIVER.
                                1262  *
                                1263  *       LDD LOADS THE REST OF A DEVICE DRIVER INTO MEMORY, AND DOES A
                                1264  *       PARTICAL RELOCATION.
                                1265  *
                                1266  *       LDD IS ENTERED WITH THE FIRST 256 BYTES OF THE DRIVER IN
                                1267  *       'BUFF'. THE DEVICE DRIVER IS OPEN ON CHANNEL 'CN.DVD', AND
                                1268  *       POSITIONED JUST AFTER THE FIRST SECTOR.
                                1269  *
                                1270  *       LDD COMPUTES THE LENGTH OF THE DRIVER, READS IT ALL IN (INCLUDING
                                1271  *       RELOCATION TABLES), AND THEN RELOCATES ANY CODE IN THE SET PREAMBLE.
                                1272  *
                                1273  *       ENTRY   NONE
                                1274  *       EXIT    TO CALLER IF OK
                                1275  *               TO AN ERROR ABORT ADDRESS IF PROBLEMS
                                1276  *       USES    ALL
                                1277
                                1278
052.055  052 074 057           1279  LDD     LHLD    BUFF+PIC.LEN
052.060  001 377 000           1280          LXI     B,255
052.063  011                   1281          DAD     B               ROUND UP
052.064  104                   1282          MOV     B,H
052.065  016 000               1283          MVI     C,0             (BC) = SECTOR COUNT
                                1284
052.067  041 072 057           1285          LXI     H,BUFF
052.072  011                   1286          DAD     B               (HL) = NEW LWA FOR SET AND BUFFERS
052.073  305                   1287          PUSH    B               SAVE (BC)
052.074  377 052               1288          SCALL   .SETTP
052.076  332 177 043           1289          JC      ERROR           NO ROOM
052.101  301                   1290          POP     B
                                1291
052.102  021 072 060           1292          LXI     D,BUFF+256
000.000                         1293          ERRNZ   CN.DVD
052.105  005                   1294          DCR     B               COUNT - 1 FOR SECTOR ALREADY READ
052.106  257                   1295          XRA     A               (A) = CHANNEL NUMBER
052.107  377 004               1296          SCALL   .READ           READ IN REST OF DRIVER
052.111  332 177 043           1297          JC      ERROR           PROBLEMS
                                1298
                                1299  *       RELOCATE ALL REFERENCES WITHIN THE FIRST 512 BYTES
                                1300
052.114  052 076 057           1301          LHLD    BUFF+PIC.PTR    (HL) = ADDRESS OF RELOCATION TABLE
052.117  001 072 057           1302          LXI     B,BUFF          (BC) = RELOCATION CONSTANT
052.122  011                   1303          DAD     B               (HL) = ABS. ADDRESS OF REL. TABLE
052.123  136                   1304  LDD1    MOV     E,M
052.124  043                   1305          INX     H
052.125  126                   1306          MOV     D,M             (DE) = REL ADDRESS OF WORD TO RELOCATE
052.126  043                   1307          INX     H
052.127  172                   1308          MOV     A,D
052.130  263                   1309          ORA     E
052.131  310                   1310          RZ                      DONE
                                1311
                                1312  *       (DE) = REL ADDRESS OF WORD TO RELOCATE. MAKE SURE IS ELIGIBLE
                                1313
000.000                         1314          ERRNZ   DVD.ENT-512
052.132  172                   1315          MOV     A,D
052.133  376 002               1316          CPI     2
```

```
052.135  322 123 052  1317          JNC     LDD1            IS IN DRIVER CODE, DONT RELOCATE
052.140  353          1318          XCHG
052.141  011          1319          DAD     B
052.142  353          1320          XCHG                    (DE) = ABS ADDRESS OF WORD TO RELOCATE
052.143  032          1321          LDAX    D
052.144  201          1322          ADD     C
052.145  022          1323          STAX    D
052.146  023          1324          INX     D
052.147  032          1325          LDAX    D
052.150  210          1326          ADC     B
052.151  022          1327          STAX    D
052.152  303 123 052  1328          JMP     LDD1            RELOCATE NEXT REFERENCE


                      1330  **      WDD - WRITE DEVICE DRIVER.
                      1331  *
                      1332  *       WDD IS CALLED TO WRITE THE MODIFIED PORTION OF THE DEVICE DRIVER
                      1333  *       BACK.
                      1334  *
                      1335  *       THE DEVICE DRIVER IN 'BUFF' IS WRITTEN BACK, EXCEPT FOR THE PREAMBLE,
                      1336  *       WHICH WAS RELOCATED. THE DRIVER ITSELF WAS NOT RELOCATED, NOR WAS THE
                      1337  *       RELOCATION TABLE ITSELF MODIFIED.
                      1338  *
                      1339  *       ENTRY   NONE
                      1340  *       EXIT    NONE
                      1341  *       USES    ALL
                      1342
                      1343
052.155  001 002 000  1344  WDD     LXI     B,DVD.ENT/256   (BC) = SECTOR NUMBER OF START OF CODE
000.000               1345          ERRNZ   *DVD.ENT        MUST BE MULT OF 256
000.000               1346          ERRNZ   CN.DVD
052.160  257          1347          XRA     A               (A) = CN.DVD
052.161  377 047      1348          SCALL   .POSIT          POSITION FILE
052.163  332 177 043  1349          JC      ERROR
                      1350
                      1351  *       WRITE BACK
                      1352
052.166  052 074 057  1353          LHLD    BUFF+PIC.LEN
052.171  001 377 376  1354          LXI     B,255-DVD.ENT
052.174  011          1355          DAD     B
052.175  104          1356          MOV     B,H
052.176  016 000      1357          MVI     C,0             (BC) = LENGTH TO WRITE, ROUNDED TO SECTOR
052.200  021 072 061  1358          LXI     D,BUFF+DVD.ENT
000.000               1359          ERRNZ   CN.DVD
052.203  257          1360          XRA     A
052.204  377 005      1361          SCALL   .WRITE          WRITE DRIVER BACK
052.206  332 177 043  1362          JC      ERROR
052.211  311          1363          RET                     RETURN
```

```
                                1367  **      DCS - DELIMIT CHARACTER STRING.
                                1368  *
                                1369  *       DCS ADVANCES PAST THE NEXT CHARACTER STRING, AND LOCATES
                                1370  *       ITS STARTING AND ENDING ADDRESSES
                                1371  *
                                1372  *       ENTRY   (BC) = LINE POINTER
                                1373  *       EXIT    (BC) ADVANCED
                                1374  *               (DE) = STRING FWA
                                1375  *               (HL) = STRING LWA
                                1376  *               (A) = STRING SIZE
                                1377  *               'Z' SET IF EMPTY
                                1378  *       USES    ALL
                                1379
                                1380
052.212  315 114 053            1381  DCS     CALL    SKB             SKIP BLANKS
052.215  120                    1382          MOV     D,B
052.216  131                    1383          MOV     E,C
052.217  012                    1384  DCS1    LDAX    B
052.220  247                    1385          ANA     A
052.221  312 233 052            1386          JZ      DCS2            END OF LINE
052.224  003                    1387          INX     B
052.225  376 040                1388          CPI     ' '
052.227  302 217 052            1389          JNE     DCS1            NOT END OF STRING
052.232  013                    1390          DCX     B               POINT TO BLANK
                                1391
052.233  140                    1392  DCS2    MOV     H,B
052.234  151                    1393          MOV     L,C             SET LWA
052.235  053                    1394          DCX     H
052.236  175                    1395          MOV     A,L
052.237  223                    1396          SUB     E
052.240  074                    1397          INR     A
052.241  311                    1398          RET


                                1400  **      DNF - DECODE NUMERIC FIELD.
                                1401  *
                                1402  *       DNF CRACKS THE NEXT FIELD AS A NUMBER.
                                1403  *
                                1404  *       ENTRY   (BC) = LINE POINTER
                                1405  *               (A) = DEFAULT BASE
                                1406  *       EXIT    (HL) = VALUE
                                1407  *               (BC) UPDATED
                                1408  *               'C' CLEAR IF OK
                                1409  *               'C' SET IF ERROR
                                1410  *                (A) = ERROR CODE
                                1411  *       USES    ALL
                                1412
                                1413
052.242  365                    1414  DNF     PUSH    PSW             SAVE POSTRADIX
052.243  315 114 053            1415          CALL    SKB             SKIP BLANKS
052.246  361                    1416          POP     PSW             RESTORE POSTRADIX
052.247  140                    1417          MOV     H,B
052.250  151                    1418          MOV     L,C
052.251  315 243 054            1419          CALL    $DNV            DECODE NUMERIC VALUE
```

```
052.254  104           1420          MOV     B,H
052.255  115           1421          MOV     C,L             RESET POINTER
052.256  353           1422          XCHG
052.257  320           1423          RNC                     ALL OK
052.260  076 037       1424          MVI     A,EC.ILV        ILLEGAL VALUE
052.262  311           1425          RET


                       1427  **      PSZ     - Patch Sector Zero                    /80.08.sc/
                       1428  *
                       1429  *       PSZ patches sector zero of the system device.
                       1430  *
                       1431  *       ENTRY:  A       = byte Index to modify
                       1432  *               D       = Mask
                       1433  *               E       = Value
                       1434  *
                       1435  *       EXIT:   NONE
                       1436  *
                       1437  *       USES:   ALL
                       1438  *
                       1439
052.263  365           1440  PSZ     PUSH    PSW
052.264  325           1441          PUSH    D
052.265  021 072 057   1442          LXI     D,BUFF
052.270  315 316 052   1443          CALL    RSZ
052.273  321           1444          POP     D
052.274  361           1445          POP     PSW
                       1446
052.275  041 072 057   1447          LXI     H,BUFF
052.300  315 072 030   1448          CALL    $DADA
052.303  176           1449          MOV     A,M
052.304  242           1450          ANA     D
052.305  263           1451          ORA     E
052.306  167           1452          MOV     M,A             Update the Byte
                       1453
052.307  021 072 057   1454          LXI     D,BUFF
052.312  315 124 053   1455          CALL    WSZ
052.315  311           1456          RET


                       1458  **      RSZ     - Read Sector Zero                     /80.06.sc/
                       1459  *
                       1460  *       RSZ reads sector zero of the system device
                       1461  *
                       1462  *       ENTRY:  DE      = buffer
                       1463  *
                       1464  *       EXIT:   To  *CALLER*  if no errors
                       1465  *               To  *ERROR*   if    errors
                       1466  *
                       1467
052.316  257           1468  RSZ     XRA     A
052.317  062 061 041   1469          STA     AIO.UNI         Zero Unit number
```

```
052.322  001 000 001  1470          LXI     B,256
052.325  041 000 000  1471          LXI     H,0
052.330  076 002      1472          MVI     A,DC.RER
052.332  315 130 040  1473          CALL    SYDD            Read Regardless of Volume Id
052.335  332 177 043  1474          JC      ERROR
052.340  311          1475          RET


                      1477  **      SHO - SET HOS OPTIONS.
                      1478  *
                      1479  *       SHO IS CALLED TO SET AN OPTION FIELD IN THE HDOS.SYS FILE
                      1480  *       ON THE DISK.
                      1481  *
                      1482  *       THIS FILE IS FLAGGED READ ONLY, SO SHO MUST GO THROUGH THE
                      1483  *       SYSTEM DEVICE DRIVER TO DO ITS DIRTY WORK.
                      1484  *
                      1485  *       ENTRY   (A) = INDEX OF BYTE TO CHANGE
                      1486  *               (D) = MASK FOR CHANGE
                      1487  *               (E) = NEW VALUE
                      1488  *       EXIT    DONE
                      1489  *       USES    ALL
                      1490
                      1491
052.341  306 011      1492  SHO     ADI     PIC.COD+3
052.343  325          1493          PUSH    D               SAVE VALUES
052.344  365          1494          PUSH    PSW             SAVE INDEX INTO BINARY
                      1495
052.345  041 070 053  1496          LXI     H,SHOA
052.350  021 105 053  1497          LXI     D,SHOB
052.353  076 000      1498          MVI     A,CN.DVD
052.355  377 042      1499          DB      SYSCALL+.OPENR  OPEN FILE TO GET INFO
052.357  332 177 043  1500          JC      ERROR
                      1501
                      1502  *       GET FIRST BLOCK
                      1503
052.362  052 352 040  1504          LHLD    S.CFWA
000.000               1505          ERRNZ   IOCCTD-1        MUST SKIP CHANNEL FOR #0
052.365  315 211 030  1506          CALL    $HLIHL          (HL) = ADDRESS OF CHANNEL #0
000.000               1507          ERRNZ   IOC.LNK
000.000               1508          ERRNZ   CN.DVD
052.370  021 004 000  1509          LXI     D,IOC.FLG
052.373  031          1510          DAD     D               (HL) = FLAG BYTE FOR THIS CHANNEL
052.374  076 014      1511          MVI     A,FT.OW+FT.OU
052.376  266          1512          ORA     M
052.377  167          1513          MOV     M,A             KLUDGE IT TO OPEN FOR UPDATE
                      1514
053.000  076 000      1515          MVI     A,CN.DVD
053.002  001 000 001  1516          LXI     B,256
053.005  021 072 057  1517          LXI     D,BUFF
053.010  377 004      1518          DB      SYSCALL+.READ   READ IN THE FIRST SECTOR
053.012  332 177 043  1519          JC      ERROR
                      1520
                      1521  *       MODIFY THE SPECIFIED BYTE
                      1522
```

```
053.015  361           1523           POP     PSW             (A) = INDEX
053.016  321           1524           POP     D               (DE) = VALUES
053.017  041 072 057   1525           LXI     H,BUFF
053.022  315 101 030   1526           CALL    $DADA           (HL) = ADDRESS OF BYTE TO CHANGE
053.025  176           1527           MOV     A,M
053.026  242           1528           ANA     D
053.027  263           1529           ORA     E
053.030  167           1530           MOV     M,A             UPDATE
                       1531
                       1532  *        RE-WRITE THE SECTOR
                       1533
053.031  076 000       1534           MVI     A,CN.DVD
053.033  001 000 000   1535           LXI     B,0
053.036  377 047       1536           DB      SYSCALL,.POSIT
053.040  332 177 043   1537           JC      ERROR
                       1538
053.043  076 000       1539           MVI     A,CN.DVD
053.045  001 000 001   1540           LXI     B,256
053.050  021 072 057   1541           LXI     D,BUFF
053.053  377 005       1542           DB      SYSCALL,.WRITE
053.055  332 177 043   1543           JC      ERROR
                       1544
053.060  076 000       1545           MVI     A,CN.DVD
053.062  377 046       1546           DB      SYSCALL,.CLOSE  CLOSE FILE
053.064  332 177 043   1547           JC      ERROR
053.067  311           1548           RET
                       1549
053.070  123 131 060   1550  SHOA     DB      'SY0:HDOS.SYS',0
053.105  000 000 000   1551  SHOB     DB      0,0,0,0,0,0
```

```
                       1553  **       SKB - SKIP BLANKS.
                       1554  *
                       1555  *        SKB SKIPS BLANKS IN THE LINE.
                       1556  *
                       1557  *        ENTRY   (BC) = LINE POINTER
                       1558  *        EXIT    (BC) UPDATE
                       1559  *                'Z' SET IFF END OF LINE
                       1560  *        USES    A,F,B,C
                       1561
                       1562
053.113  003           1563  SKB1     INX     B
053.114  012           1564  SKB      LDAX    B
053.115  376 040       1565           CPI     ' '
053.117  312 113 053   1566           JE      SKB1            IF BLANK
053.122  247           1567           ANA     A
053.123  311           1568           RET
```

```
                                1570  **      WSZ     - Write Sector Zero                             /80.06.sc/
                                1571  *
                                1572  *       WSZ writes sector zero of the system device
                                1573  *
                                1574  *       ENTRY:  DE      = address of sector 0
                                1575  *
                                1576  *       EXIT:   To  *CALLER*  if no errors
                                1577  *               To  *ERROR*   if    errors
                                1578  *
                                1579
053.124  257                   1580  WSZ     XRA     A
053.125  062 061 041           1581          STA     AIO.UNI         Zero the unit number
                                1582
053.130  325                   1583          PUSH    D
053.131  041 000 000           1584          LXI     H,0
053.134  076 010               1585          MVI     A,DC.MOU
053.136  315 130 040           1586          CALL    SYDD            Mount SY0: with volume zero
053.141  321                   1587          POP     D
053.142  332 177 043           1588          JC      ERROR
                                1589
053.145  001 000 001           1590          LXI     B,256           Write One sector
053.150  041 000 000           1591          LXI     H,0             Sector = 0
053.153  076 001               1592          MVI     A,DC.WRI
053.155  315 130 040           1593          CALL    SYDD            Write Sector 0
053.160  332 177 043           1594          JC      ERROR
                                1595
053.163  001 000 001           1596          LXI     B,256
053.166  021 072 055           1597          LXI     D,LABEL
053.171  041 011 000           1598          LXI     H,DDF.LAB
053.174  076 002               1599          MVI     A,DC.RER
053.176  315 130 040           1600          CALL    SYDD            Read Label
053.201  332 177 043           1601          JC      ERROR
                                1602
053.204  072 072 055           1603          LDA     LABEL+LAB.SER
053.207  157                   1604          MOV     L,A
053.210  046 000               1605          MVI     H,0             HL = Volume Number
053.212  076 010               1606          MVI     A,DC.MOU
053.214  315 130 040           1607          CALL    SYDD            Re-mount the Volume with real volume number
053.217  332 177 043           1608          JC      ERROR
                                1609
053.222  311                   1610          RET
```

```
                                1613
                                1614
053.223                         1615            XTEXT   FST


                                1617X **        $FST - FIND IN SERIAL TABLE
                                1618X *
                                1619X *         $FST SEARCHES A SERIAL TABLE FOR
                                1620X *         A SPECIFIC KEY
                                1621X *
                                1622X *         ENTRY   (HL) = ADDR. OF TABLE
                                1623X *                 (DE) = ADDR. OF SEARCH KEY
                                1624X *         EXIT    (DE) = UNCHANGED
                                1625X *                 'Z' CLEARED IF NO MATCH FOUND
                                1626X *                   (HL) = ADDR. OF NEXT AVAILABLE BYTE
                                1627X *                 'Z' SET IF MATCH FOUND
                                1628X *                   (HL) = ADDR. OF FIRST DATA BYTE
                                1629X *         USES    A,F,H,L
                                1630X
                                1631X
                                1632X
053.223  305                   1633X $FST      PUSH    B               SAVE REGISTERS
053.224  325                   1634X           PUSH    D
                                1635X
                                1636X *         SAVE TABLE LIMIT AND DATA BYTE COUNT
                                1637X
053.225  136                   1638X           MOV     E,M             GET AND SAVE TABLE LIMIT
053.226  043                   1639X           INX     H               (HL) = 2ND BYTE OF SIZE
053.227  126                   1640X           MOV     D,M
053.230  353                   1641X           XCHG
053.231  042 323 053           1642X           SHLD    $FST.L          SAVE MAX. TABLE SIZE
                                1643X
053.234  353                   1644X           XCHG
053.235  043                   1645X           INX     H               (HL) = # OF BYTES OF DATA/ENTRY
053.236  176                   1646X           MOV     A,M
053.237  062 325 053           1647X           STA     $FST.C
053.242  043                   1648X           INX     H               (HL) = BEGINNING OF DATA
053.243  321                   1649X FST1      POP     D               RESTORE ADDR. TO SEARCH KEY
053.244  325                   1650X           PUSH    D
                                1651X
                                1652X *         CHECK FOR END OF DATA
                                1653X
053.245  176                   1654X           MOV     A,M
053.246  267                   1655X           ORA     A               AT END OF DATA? ((A) = 0)
053.247  302 256 053           1656X           JNZ     FST2            NO, START MATCHING
053.252  074                   1657X           INR     A               CLEAR 'Z'
053.253  321                   1658X           POP     D
053.254  301                   1659X           POP     B               RESTORE REGISTERS
053.255  311                   1660X           RET
                                1661X
053.256  032                   1662X FST2      LDAX    D               (A) = KEY CHAR.
053.257  276                   1663X           CMP     M               COMPARE TO TABLE
053.260  302 274 053           1664X           JNE     FST3            NO MATCH, FIND NEXT KEY
053.263  247                   1665X           ANA     A               END OF KEY?
```

```
053.264  372 316 053  1666X          JM      FST4            YES, SET UP FOR EXIT
053.267  043         1667X          INX     H
053.270  023         1668X          INX     D
053.271  303 256 053 1669X          JMP     FST2
                     1670X
053.274  176         1671X FST3     MOV     A,M             SEARCH FOR END OF KEY
053.275  247         1672X          ANA     A               TEST CHAR.
053.276  043         1673X          INX     H
053.277  362 274 053 1674X          JP      FST3            CONTINUE SEARCH
053.302  072 325 053 1675X          LDA     $FST.C          (A) = # OF BYTES OF DATA/ENTRY
053.305  205         1676X          ADD     L
053.306  157         1677X          MOV     L,A
053.307  076 000     1678X          MVI     A,0
053.311  214         1679X          ADC     H
053.312  147         1680X          MOV     H,A             (HL) = HEAD OF NEXT KEY
053.313  303 243 053 1681X          JMP     FST1            COMPARE NEXT KEY
                     1682X
053.316  257         1683X FST4     XRA     A               SET'Z' FOR EXIT
053.317  043         1684X          INX     H               (HL) = FIRST BYTE OF DATA
053.320  321         1685X          POP     D               RESTORE REGISTERS
053.321  301         1686X          POP     B
053.322  311         1687X          RET                     EXIT
                     1688X
                     1689X
053.323              1690X $FST.L   DS      2
053.325              1691X $FST.C   DS      1
053.326              1692           XTEXT   LBD


                     1694X ***      $LBD - LOOKUP BAUDRATE DIVISOR.
                     1695X *
                     1696X *        $LBD TRANSLATES A BAUD RATE INTO THE PROPER DIVISOR FOR THE
                     1697X *        8250 CHIPS ON THE H8-4 SERIAL CARD.
                     1698X *
                     1699X *        NOTE THAT $LBD DOES NOT ACTUALLY COMPUTE THE TRANSFORMATION, BUT
                     1700X *        SIMPLY LOOKS UP THE VALUE IN A TABLE. THIS IS DONE TO DETECT TYPOS
                     1701X *        IN THE USER SUPPLIED BAUD RATE.
                     1702X *
                     1703X *        ENTRY   (DE) = BAUD RATE (AS A BINARY NUMBER)
                     1704X *        EXIT    'Z' SET IF VALID BAUD RATE
                     1705X *                 (HL) = DIVISOR
                     1706X *                'Z' CLEAR IF NOT VALID BAUD RATE
                     1707X *        USES    A,F,D,E,H,L
                     1708X
                     1709X
053.326  172         1710X $LBD     MOV     A,D
053.327  263         1711X          ORA     E               (A) = CODE VALUE
053.330  041 343 053 1712X          LXI     H,LBDA          (HL) = LOOKUP TABLE
053.333  315 217 054 1713X          CALL    $WTBLS          WORD TABLE LOOKUP
053.336  176         1714X          MOV     A,M
053.337  043         1715X          INX     H
053.340  146         1716X          MOV     H,M
053.341  157         1717X          MOV     L,A
053.342  311         1718X          RET                     RETURN WITH CONDITION CODE FROM $WTBLS
```

```
                                1719X
                                1720X
                                1721X **      BAUD RATE VS 8250 DIVISOR TABLE.
                                1722X *
                                1723X *       KEY IS BAUD RATE SQUEEZED INTO ONE BYTE
                                1724X
053.343                         1725X LBDA    DS      0
053,343  151                    1726X         DB      2400/256!$2400
053.344  060 000                1727X         DW      000060A
053,346  245                    1728X         DB      9600/256!$9600
053.347  014 000                1729X         DW      000014A
053,351  132                    1730X         DB      600/256!$600
053.352  300 000                1731X         DW      000300A
053,354  113                    1732X         DB      19200/256!$19200
053.355  006 000                1733X         DW      000006A
053,357  322                    1734X         DB      4800/256!$4800
053.360  030 000                1735X         DW      000030A
053,362  264                    1736X         DB      1200/256!$1200
053.363  140 000                1737X         DW      000140A
053,365  055                    1738X         DB      300/256!$300
053.366  200 001                1739X         DW      001200A
053,370  074                    1740X         DB      7200/256!$7200
053.371  020 000                1741X         DW      000020A
053,373  036                    1742X         DB      3600/256!$3600
053.374  040 000                1743X         DW      000040A
053,376  017                    1744X         DB      1800/256!$1800
053.377  100 000                1745X         DW      000100A
054,001  156                    1746X         DB      110/256!$110
054.002  027 204                1747X         DW      204027A
054,004  226                    1748X         DB      150/256!$150
054.005  000 003                1749X         DW      003000A
054,007  000                    1750X         DB      0               END OF TABLE
054.010                         1751          XTEXT   SOP
```

```
                                1753X **      SOP - SET OPTIONS
                                1754X *
                                1755X *       PROCESS OPTION SET VIA OPTION TABLE, AND PROCESSOR TABLE.
                                1756X *
                                1757X *       OPTION TABLE FORMAT:  P(1)=PROCESSOR INDEX, P(2,...,N)=PARAMETERS
                                1758X *
                                1759X *               DW      <END OF TABLE>
                                1760X *               DB      N
                                1761X *               DB      '<SEARCH STRIN','G>'+200Q,P,P(1),...,P(N)
                                1762X *               .       .                       .
                                1763X *               .       .                       .
                                1764X *       <EOT>   DB      0                       END OF TABLE
                                1765X *
                                1766X *       PROCESSOR TABLE FORMAT:
                                1767X *
                                1768X *               DW      PROC.0
                                1769X *               DW      PROC.1
                                1770X *               .       .
                                1771X *               .       .
                                1772X *               DW      PROC.N
                                1773X *
                                1774X *
                                1775X *
                                1776X *       ENTRY:  (BC)    = LINE POINTER
                                1777X *               (DE)    = JUMP TABLE ADDRESS
                                1778X *               (HL)    = OPTION TABLE ADDRESS
                                1779X *
                                1780X *       EXIT:   (RET)   = TO PROCESSOR IF NO ERROR
                                1781X *                       = 'C' SET IF ERROR
                                1782X *                          (A) = ERROR CODE
                                1783X *               (BC)    = LINE POINTER UPDATED
                                1784X *               (HL)    = ADDRESS OF NEXT AVAILABLE DATA BYTE
                                1785X *
                                1786X *       USES:   ALL
                                1787X
                                1788X
054.010  325                    1789X SOP     PUSH    D
054.011  345                    1790X         PUSH    H
054.012  315 212 052            1791X         CALL    DCS             (DE) = FWA, (HL) = LWA
054.015  312 051 054            1792X         JZ      SOP1
054.020  176                    1793X         MOV     A,M
054.021  366 200                1794X         ORI     200Q
054.023  167                    1795X         MOV     M,A
054.024  341                    1796X         POP     H               (HL) = OPT. TABLE ADDR.
054.025  315 223 053            1797X         CALL    $FST
054.030  302 052 054            1798X         JNZ     SOP2
054.033  353                    1799X         XCHG                    (DE) = ADDR. OF FIRST DATA BYTE
054.034  341                    1800X         POP     H               (HL) = JUMP TABLE ADDR.
054.035  032                    1801X         LDAX    D               (A)  = PROCESSOR INDEX
054.036  007                    1802X         RLC                     X 2
054.037  023                    1803X         INX     D
054.040  315 101 030            1804X         CALL    $DADA.
054.043  315 211 030            1805X         CALL    $HLIHL          (HL) = PROCESSOR ADDRESS
054.046  345                    1806X         PUSH    H
054.047  353                    1807X         XCHG                    (HL) = NEXT DATA BYTE ADDRESS
054.050  311                    1808X         RET                     ENTER PROCESSOR
```

```
                        1809X
054,051  341           1810X SOP1   POP     H
054.052  321           1811X SOP2   POP     D
054.053  076 040       1812X        MVI     A,EC.ILO        ILLEGAL OPTION SPECIFICATION
054.055  067           1813X        STC
054.056  311           1814X        RET
054.057                1815         XTEXT   PBF


                       1817X **      PBF - PROCESS BYTE FLAG
                       1818X *
                       1819X *       PROCESS BYTE FLAG OPTIONS, THE FORMAT FOR TABLE ENTRIES IS:
                       1820X *
                       1821X *             <MASK>,<VALUE>,<LOW ADDR.>,<HIGH ADDR.>
                       1822X *
                       1823X *
                       1824X *       ENTRY:  (HL)    = ADDRESS OF TABLE VECTOR
                       1825X *
                       1826X *       EXIT:   (RET)   = 'C' CLEAR IF OK
                       1827X *                       = 'C' SET IF ERROR
                       1828X *                         (A) = ERROR CODE
                       1829X *
                       1830X *       USES:   ALL
                       1831X *
                       1832X
054.057  176           1833X PBF    MOV     A,M             (A) = MASK
054.060  043           1834X        INX     H
054.061  365           1835X        PUSH    PSW
054.062  246           1836X        ANA     M               MASK UNUSED BITS OUT OF VALUE
054.063  127           1837X        MOV     D,A             (D) = VALUE
054.064  043           1838X        INX     H
054.065  315 211 030   1839X        CALL    $HLIHL          (HL) = ADDRESS TO STORE BYTE
054.070  361           1840X        POP     PSW
054.071  057           1841X        CMA
054.072  246           1842X        ANA     M               MASK OUT PREVIOUS VALUE
054.073  262           1843X        ORA     D               SET NEW FLAGS
054.074  167           1844X        MOV     M,A             PATCH IT
054.075  311           1845X        RET
054.076                1846         XTEXT   PBV


                       1848X **      PBV - PROCESS BYTE VALUES
                       1849X *
                       1850X *       PROCESS BYTE VALUE OPTIONS.  THE FORMAT FOR TABLE ENTRIES
                       1851X *       IS:
                       1852X *
                       1853X *             <DEFAULT RADIX>,<MIN.>,<MAX.>,<LOW. ADDR.>,<HIGH ADDR.>
                       1854X *
                       1855X *
                       1856X *       ENTRY:  (BC)    = NEXT TEXT CHARACTER ADDRESS
                       1857X *               (HL)    = TABLE VECTOR ADDRESS
                       1858X *
```

```
                                1859X *       EXIT:   (BC)    = UPDATED
                                1860X *               'C' CLEAR IF OK
                                1861X *               'C' SET IF ERROR
                                1862X *                (A) = ERROR CODE
                                1863X *
                                1864X *       USES:   ALL
                                1865X *
                                1866X
054.076  176                    1867X PBV     MOV     A,M             (A)  = DEFAULT RADIX
054.077  043                    1868X         INX     H
054.100  345                    1869X         PUSH    H               SAVE VECTOR ADDRESS
054.101  315 242 052            1870X         CALL    DNF             (HL) = VALUE
054.104  332 147 054            1871X         JC      PBV2
054.107  174                    1872X         MOV     A,H
054.110  247                    1873X         ANA     A
054.111  302 147 054            1874X         JNZ     PBV2
054.114  321                    1875X         POP     D
054.115  353                    1876X         XCHG                    (HL) = NEXT TABLE ADDRESS, (E) = VALUE
054.116  305                    1877X         PUSH    B               SAVE TEXT POINTER
054.117  106                    1878X         MOV     B,M             (B)  = MIN.
054.120  043                    1879X         INX     H
054.121  116                    1880X         MOV     C,M             (C)  = MAX.
054.122  043                    1881X         INX     H
054.123  315 211 030            1882X         CALL    $HLIHL          (HL) = BYTE  VALUE ADDRESS
054.126  173                    1883X         MOV     A,E
054.127  270                    1884X         CMP     B
054.130  332 151 054            1885X         JC      PBV3            (A) < MIN.
054.133  014                    1886X         INR     C
054.134  312 143 054            1887X         JZ      PBV1            IGNORE COMPARE IF C=377Q
054.137  271                    1888X         CMP     C
054.140  322 151 054            1889X         JNC     PBV3            (A) >= MAX. + 1
054.143  301                    1890X PBV1    POP     B               RESTORE TEXT ADDR.
054.144  167                    1891X         MOV     M,A             PATCH IT
054.145  257                    1892X         XRA     A               CLEAR CARRY
054.146  311                    1893X         RET
                                1894X
054.147  341                    1895X PBV2    POP     H
054.150  305                    1896X         PUSH    B
054.151  301                    1897X PBV3    POP     B
054.152  076 037                1898X         MVI     A,EC.ILV        ILLEGAL VALUE SPECIFICATION
054.154  067                    1899X         STC
054.155  311                    1900X         RET
054.156                         1901          XTEXT   TJMP


                                1903X **      $TJMP - TABLE JUMP.
                                1904X *
                                1905X *       USAGE
                                1906X *
                                1907X *       CALL      $TJMP           (A) = INDEX
                                1908X *       DW        ADDR1
                                1909X *       .           .
                                1910X *       .           .
                                1911X *       .           .
```

```
                        1912X *       DW        ADDRN
                        1913X *
                        1914X *       ENTRY     (A) = INDEX
                        1915X *       EXIT      TO PROCESSOR
                        1916X *                 (A) = INDEX*2
                        1917X *       USES      NONE.
                        1918X
                        1919X
    031.061             1920X $TJMP   EQU     31061A          IN H17 ROM, (A) = INDEX*2
                        1921X
    031.062             1922X $TJMP.  EQU     31062A          IN H17 ROM
    054.156             1923          XTEXT   MU86



                        1925X **      $MU86 - MULTIPLY 8X16 UNSIGNED.
                        1926X *
                        1927X *       $MU86 MULTIPLIES A 16 BIT VALUE BY A 8
                        1928X *       BIT VALUE.
                        1929X *
                        1930X *       ENTRY   (A) = MULTIPLIER
                        1931X *               (DE) = MULTIPLICAND
                        1932X *       EXIT    (HL) = RESULT
                        1933X *               'Z' SET IF NOT OVERFLOW
                        1934X *       USES    A,F,H,L
                        1935X
                        1936X
    031.007             1937X $MU86   EQU     31007A          IN H17 ROM
    054.156             1938          XTEXT   DADA



                        1940X **      $DADA - PERFORM (H,L) = (H,L) + (0,A)
                        1941X *
                        1942X *       ENTRY   (H,L) = BEFORE VALUE
                        1943X *               (A) = BEFORE VALUE
                        1944X *       EXIT    (H,L) = (H,L) + (0,A)
                        1945X *               'C' SET IF OVERFLOW
                        1946X *       USES    F,H,L
                        1947X
                        1948X
    030.072             1949X $DADA   EQU     30072A          IN H17 ROM
    054.156             1950          XTEXT   DADA2



                        1952X **      $DADA. - ADD (0,A) TO (H,L)
                        1953X *
                        1954X *       ENTRY   NONE
                        1955X *       EXIT    (HL) = (HL) + (0A)
                        1956X *       USES    A,F,H,L
                        1957X
                        1958X
```

```
030.101                 1959X $DADA.  EQU     30101A          IN H17 ROM
054.156                 1960          XTEXT   HLIHL

                        1962X **      $HLIHL - LOAD HL INDIRECT THROUGH HL.
                        1963X *
                        1964X *       (HL) = ((HL))
                        1965X *
                        1966X *       ENTRY   NONE
                        1967X *       EXIT    NONE
                        1968X *       USES    A,H,L
                        1969X
030.211                 1970X $HLIHL  EQU     30211A          IN H17 ROM
054.156                 1971          XTEXT   CVD

                        1973X **      $CVD - CHECK FOR VALID DIGIT.
                        1974X *
                        1975X *       CVD EXAMINES A DIGIT TO SEE IF IT IS A VALID DECIMAL DIGIT.
                        1976X *
                        1977X *       ENTRY   (HL) = ADDRESS OF CHARACTER
                        1978X *       EXIT    'C' SET IF ILLEGAL
                        1979X *                (A) = VALUE
                        1980X *       USES    A,F
                        1981X
                        1982X
054.156  176           1983X $CVD    MOV     A,M             (A) = CHARACTER
054.157  326 060       1984X $CVD.   SUI     '0'
054.161  330           1985X         RC                      ILLEGAL
054.162  376 012       1986X         CPI     9+1
054.164  077           1987X         CMC
054.165  311           1988X         RET
054.166                 1989          XTEXT   TBLS

                        1991X **      $TBLS - TABLE SEARCH
                        1992X *
                        1993X *       TABLE FORMAT
                        1994X *
                        1995X *       DB      KEY1,VAL1,
                        1996X *       .       .
                        1997X *       .       .
                        1998X *       DB      KEYN,VALN
                        1999X *       DB      0
                        2000X *
                        2001X *       ENTRY   (A) = PATTERN
                        2002X *               (H,L) = TABLE FWA
                        2003X *       EXIT    (A) = PATTERN IF FOUND
                        2004X *               'Z' SET IF FOUND
                        2005X *               'Z' CLEAR IF NOT FOUND OR PATTERN=0          /78.10.GC/
```

```
                        2006X *       USES    A,F,H,L
                        2007X
                        2008X
054,166  305           2009X $TBLS   PUSH    B
054,167  376 000       2010X         CPI     0                                  /78.10.GC/
054,171  312 213 054   2011X         JZ      TBL2                               /78.10.GC/
054,174  107           2012X         MOV     B,A
054,175  176           2013X TBL1    MOV     A,M             (A) = CHARACTER
054,176  043           2014X         INX     H
054,177  270           2015X         CMP     B
054,200  312 215 054   2016X         JZ      TBL3            IF MATCH
054,203  247           2017X         ANA     A
054,204  043           2018X         INX     H               SKIP PAST
054,205  302 175 054   2019X         JNZ     TBL1            IF NOT END OF TABLE
054,210  053           2020X         DCX     H
054,211  053           2021X         DCX     H
054,212  257           2022X         XRA     A               SET TO ZERO FOR OLD USERS  /78.10.GC/
054,213  376 001       2023X TBL2    CPI     1               CLEAR ZERO                 /78.10.GC/
                        2024X
                        2025X *       DONE
                        2026X
054,215  301           2027X TBL3    POP     B
054,216  311           2028X         RET
054,217                2029          XTEXT   WTBLS


                        2031X **      $WTBLS - TABLE SEARCH
                        2032X *
                        2033X *       $WTBLS LOOKS UP WORD VALUES IN A TABLE, USING A ONE-BYTE
                        2034X *       KEY.
                        2035X *
                        2036X *       TABLE FORMAT
                        2037X *
                        2038X *       DB      KEY1
                        2039X *       DW      VAL1
                        2040X *       .       .
                        2041X *       .       .
                        2042X *       DB      KEYN
                        2043X *       DW      VALN
                        2044X *       DB      0
                        2045X *
                        2046X *       ENTRY   (A) = PATTERN
                        2047X *               (H,L) = TABLE FWA
                        2048X *       EXIT    (A) = PATTERN IF FOUND
                        2049X *               'Z' SET IF FOUND
                        2050X *       USES    A,F,H,L
                        2051X
                        2052X
054.217  305           2053X $WTBLS  PUSH    B
054.220  107           2054X         MOV     B,A
054.221  176           2055X $WTBL1  MOV     A,M             (A) = CHARACTER
054.222  043           2056X         INX     H
054.223  270           2057X         CMP     B
054.224  312 241 054   2058X         JZ      $WTBL2          IF MATC
```

```
054.227  247            2059X          ANA     A
054.230  043            2060X          INX     H
054.231  043            2061X          INX     H                SKIP PAST
054.232  302 221 054     2062X          JNZ     $WTBL1           IF NOT END OF TABLE
054.235  053            2063X          DCX     H
054.236  053            2064X          DCX     H
054.237  053            2065X          DCX     H
054.240  264            2066X          ORA     H                CLEAR 'Z'
                        2067X
                        2068X *        DONE
                        2069X
054.241  301            2070X $WTBL2   POP     B
054.242  311            2071X          RET
054.243                 2072           XTEXT   DNV



                        2074X **       $DNV - DECODE NUMERIC VALUE.
                        2075X *
                        2076X *        $DNV DECODES A NUMERIC VALUE (IN THE FORM OF AN ASCII STRING)
                        2077X *        INTO A BINARY NUMBER. THE MAXIMUM MAGNITUDE IS
                        2078X *        65535D.
                        2079X *
                        2080X *        THE NUMBER MAY CONTAIN A POSTRADIX OF 'B' (BINARY)
                        2081X *        'O' OR 'Q' (OCTAL) OR 'D' (DECIMAL)
                        2082X *
                        2083X *        ENTRY   (HL) = ADDRESS OF FIRST BYTE OF NUMBER
                        2084X *                (A) = DEFAULT BASE (2 FOR BINARY, 10 FOR DECIMAL, ETC.)
                        2085X *        EXIT    'C' CLEAR IF OK
                        2086X *                 (HL) ADVANCED PAST NUMBER (AND POSTRADIX)
                        2087X *                 (DE) = VALUE
                        2088X *                'C' SET IF ERROR
                        2089X *        USES    ALL
                        2090X
                        2091X
054.243  062 360 054     2092X $DNV     STA     $DNVA            SET DEFAULT BASE
054.246  104            2093X          MOV     B,H
054.247  115            2094X          MOV     C,L              (BC) = TEXT ADDRESS
                        2095X
                        2096X *        SCAN FOR POSTRADIX
                        2097X
054.250  176            2098X $DNV1    MOV     A,M
054.251  315 157 054     2099X          CALL    $CVD.            CHECK FOR VALID DECIMAL DIGIT
054.254  043            2100X          INX     H
054.255  322 250 054     2101X          JNC     $DNV1            MORE TO GO
054.260  053            2102X          DCX     H                REMOVE EXTRA INCREMENT
054.261  171            2103X          MOV     A,C
054.262  275            2104X          CMP     L                SEE IF THERE WERE ANY NUMBERS
054.263  067            2105X          STC                      ASSUME NOT
054.264  310            2106X          RE                       ERROR
                        2107X
                        2108X *        OUT OF NUMBERS. SEE IF POSTRADIX FOLLOWS
                        2109X
054.265  176            2110X          MOV     A,M              (A) = PROPOSED POSTRADIX
054.266  345            2111X          PUSH    H                SAVE END ADDRESS
```

```
054.267  041 361 054  2112X          LXI     H,$DNVB
054.272  247          2113X          ANA     A
054.273  312 313 054  2114X          JZ      $DNV2           NO POSTRADIX
054.276  315 166 054  2115X          CALL    $IBLS
054.301  176          2116X          MOV     A,M
054.302  302 313 054  2117X          JNE     $DNV2           NOT POSTRADIX
054.305  341          2118X          POP     H
054.306  043          2119X          INX     H               SKIP POSTRADIX
054.307  345          2120X          PUSH    H
054.310  062 360 054  2121X          STA     $DNVA           SET NEW POSTRADIX
054.313  021 000 000  2122X $DNV2     LXI     D,0             (DE) = ACCUMULATOR
                      2123X
                      2124X *        BUILD NUMBER
                      2125X
054.316  072 360 054  2126X $DNV3     LDA     $DNVA           (A) = BASE
054.321  365          2127X          PUSH    PSW             SAVE BASE
054.322  315 007 031  2128X          CALL    $MU86           MULTIPLY
054.325  321          2129X          POP     D               (D) = BASE
054.326  332 356 054  2130X          JC      $DNV4           OVERFLOW
054.331  012          2131X          LDAX    B               (A) = DIGIT
054.332  326 060      2132X          SUI     '0'
054.334  003          2133X          INX     B
054.335  272          2134X          CMP     D               COMPARE TO BASE
054.336  077          2135X          CMC
054.337  332 356 054  2136X          JC      $DNV4           TOO LARGE A DIGIT
054.342  315 101 030  2137X          CALL    $DADA,          ADD TO VALUE
054.345  353          2138X          XCHG                    (DE) = VALUE
054.346  012          2139X          LDAX    B
054.347  315 157 054  2140X          CALL    $CVD.
054.352  322 316 054  2141X          JNC     $DNV3           MORE TO GO
054.355  247          2142X          ANA     A               CLEAR CARRY
054.356  341          2143X $DNV4     POP     H               RESTORE POINTER
054.357  311          2144X          RET                     EXIT
                      2145X
054.360  000          2146X $DNVA     DB      0               DEFAULT BASE
054.361  102 002      2147X $DNVB     DB      'B',2           POSTRADIX TABLE
054.363  117 010      2148X          DB      'O',8
054.365  121 010      2149X          DB      'Q',8
054.367  104 012      2150X          DB      'D',10
054.371  000          2151X          DB      0
054.372               2152           XTEXT   TYPTX


                      2154X **       $TYPTX - TYPE TEXT.
                      2155X *
                      2156X *        $TYPTX IS CALLED TO TYPE A BLOCK OF TEXT ON THE SYSTEM CONSOLE.
                      2157X *
                      2158X *        IMBEDDED ZERO BYTES INDICATE A CARRIAGE RETURN LINE FEED,
                      2159X *        A BYTE WITH THE 2000 BIT SET IS THE LAST BYTE IN THE MESSAGE.
                      2160X *
                      2161X *        ENTRY   (RET) = TEXT
                      2162X *        EXIT    TO (RET+LENGTH)
                      2163X *        USES    A,F
                      2164X
```

```
                        2165X
031.136                 2166X $TYPTX  EQU     31136A          IN H17 ROM
                        2167X
031.144                 2168X $TYPTX. EQU     31144A          IN H17 ROM
```

```
                        2171
        054.372         2172  PATCH   DS      64              PATCH AREA
                        2173
        055.072         2174  MEML    EQU     *               LOAD MEMORY LENGTH
                        2175
                        2176  **      WORK BUFFER.
                        2177  *
                        2178  *       'BUFF' IS A WORK BUFFER, AT LEAST 256 BYTES LONG.
                        2179  *       'BUFF' MAY BE EXTENDED BY ROUTINES BY ISSUING THE '.SETTP' REQUEST.
                        2180  *       THUS, NO DATA AREAS MAY FOLLOW 'BUFF'
                        2181
                        2182
        055.072         2183  LABEL   DS      256             Label buffer                    /80.06.sc/
                        2184
        056.072         2185  SECT    DS      256             Sector Buffer                   /80.06.sc/
                        2186
        057.072         2187  BUFF    DS      256             WORK BUFFER
                        2188
        060.072         2189  RMEML   EQU     *               INITIAL RUNNING LIMIT
                        2190
        060.072         2191          END                     (MUST IMMEDIATELY FOLLOW *BUFF* )
ASSEMBLY COMPLETE
 2191 STATEMENTS
    0 ERRORS DETECTED
11772 BYTES FREE
```

```
$CVD    054156  1983L
$CVD.   054157  1984L  2099  2140
$DADA   030072  1448   1949E
$DADA.  030101  1526   1804  1959E  2137
$DNV    054243  1419   2092L
$DNV1   054250  2098L  2101
$DNV2   054313  2114   2117  2122L
$DNV3   054316  2126L  2141
$DNV4   054356  2130   2136  2143L
$DNVA   054360  2092   2121  2126  2146L
$DNVB   054361  2112   2147L
$FST    053223   721    779  1633L  1797
$FST.C  053325  1647   1675  1691L
$FST.L  053323  1642   1690L
$HLIHL  030211   961   1506  1805  1839  1882  1970E
$LBD    053326   736   1710L
$MU86   031007  1937E  2128
$TBLS   054166   726   2009L  2115
$TJMP   031061   782   1920E
$TJMP.  031062  1922E
$TYPTX  031136   810    817   824   832   979  1006  1039  1127  1138  1226  2166E
$TYPTX. 031144  2168E
$WTBL1  054221  2055L  2062
$WTBL2  054241  2058   2070L
$WTBLS  054217   731   1713  2053L
.       050150  1072S  1074  1077  1080
.CHFLG  000060   136L
.CLEAN  000205   151L
.CLEAR  000055   133L
.CLEARA 000056   134L
.CLOSE  000046   126L  1252  1546
.CLRCO  000007   110L
.CONSL  000006   109L
.CTLC   000041   121L
.DAD    000206   152L
.DECODE 000053   131L
.DELET  000050   128L
.DISMT  000061   137L
.DMNMS  000203   149L
.DMOUN  000201   147L
.ERROR  000057   135L   836
.EXIT   000000   103L   846
.LINK   000040   120L
.LOADD  000062   138L
.LOADO  000010   111L
.MONMS  000202   148L
.MOUNT  000200   146L
.NAME   000054   132L
.OPEN   000063   139L
.OPENC  000045   125L
.OPENR  000042   122L  1499
.OPENU  000044   124L  1211
.OPENW  000043   123L
.POSIT  000047   127L  1348  1536
.PRINT  000003   106L
.READ   000004   107L  1220  1296  1518
.RENAM  000051   129L
.RESET  000204   150L
```

```
.SCIN   000001      104L
.SCOUT  000002      105L
.SETIP  000052      130L      761    1288
.SYSRES 000012      113L
.VERS   000011      112L
.WRITE  000005      108L     1361    1542
ABS.COD 000010      568L      653
ABS.ENT 000006      566L
ABS.ID  000000      562L
ABS.LDA 000002      564L
ABS.LEN 000004      565L
AIO.CGN 041047      470L
AIO.CHA 041116      485L
AIO.CNT 041111      481L
AIO.CSI 041050      471L
AIO.DDA 041041      466E
AIO.DES 041055      475L
AIO.DEV 041057      476L
AIO.DIR 041062      479L
AIO.DTA 041053      474L
AIO.EOF 041113      483L
AIO.EOM 041112      482L
AIO.FLG 041043      467L
AIO.GRT 041044      468L
AIO.LGN 041051      472L
AIO.LSI 041052      473L
AIO.SPG 041046      469L
AIO.TFP 041114      484L
AIO.UNI 041061      477L     1469    1581
AIO.VEC 041040      465L
BELL    000007       68E      811     818     825     833    1129    1227
BKSP    000010       70E
BOOT.P  000001      445E
BUFF    057072     1217     1223    1232    1244    1279    1285    1292    1301    1302    1353    1358    1442
                   1447     1454    1517    1525    1541    2187L
C.STX   000002       72E
C.SYN   000026       71E
CDB.H84 000001      388E
CDB.H85 000000      387E
CN.DVD  000000       55E     1209    1218    1250    1293    1346    1359    1498    1508    1515    1534    1539
                    1545
CNA.    042207      716L
CO.FLG  000001      365E
CR      000015       64E
CS.FLG  000200      366E
CSL.CHR 000001      342E
CSL.ECH 000200      339E
CSL.RAW 000004      340E
CSL.WRP 000002      341E
CTLA    000001       79E
CTLB    000002       80E
CTLC    000003       81E
CTLD    000004       82E
CTLO    000017       83E
CTLP    000020       84E
CTLQ    000021       85E
CTLS    000023       86E
CTLZ    000032       87E
```

```
CTP.2SB 000010   351E   868   869   869
CTP.BKM 000002   352E   866   866   874
CTP.BKS 000200   347E   862   862   870
CTP.FF  000100   348E   863   863   871
CTP.MLI 000040   349E   864   864   872
CTP.MLO 000020   350E   865   865   873
CTP.TAB 000001   353E   867   867   875
CTYI    000000   862    863   864   865   866   867   868   869   870   871   872   873
                 874    875   884E
D.CON   040110   278L   297
D.ERTS  040126   313L
D.LPSA  040116   304L
D.MAIA  040115   303L
D.RAM   040240   281L
D.SDPA  040117   305L
D.SDPB  040120   306L
D.STSA  040121   307L
D.STSB  040122   308L
D.VEC   040130   280L
D.WHDA  040123   309L
D.WNHA  040124   310L
D.WRITA 040112   300L
D.WRITB 040113   301L
D.WRITC 040114   302L
D.WSCA  040125   311L
D.XITA  040110   299L
DC.ABT  000007   522L
DC.CLO  000006   521L
DC.LOD  000011   524L
DC.MAX  000013   526L
DC.MOU  000010   523L   1585  1606
DC.OPR  000003   518L
DC.OPU  000005   520L
DC.OPW  000004   519L
DC.RDY  000012   525L
DC.REA  000000   515L
DC.RER  000002   517L   1472  1599
DC.WRI  000001   516L   1592
DCS     052212   697    773   1381L 1791
DCS.    042204   697L
DCS1    052217   1384L  1389
DCS2    052233   1386   1392L
DDF.BOL 000011   612E
DDF.BOO 000000   611L
DDF.LAB 000011   613L   1598
DDF.USR 000012   614L
DF.CLR  000376   161E
DF.EMP  000377   160E
DIR.ALD 000025   176L
DIR.CLU 000015   169L
DIR.CRD 000023   175L
DIR.EXT 000010   164L
DIR.FGN 000020   172L
DIR.FLG 000016   170L
DIR.LGN 000021   173L
DIR.LSI 000022   174L
DIR.NAM 000000   163L
DIR.PRO 000013   165L
```

```
DIR.VER 000014   166L
DIRELEN 000027   178E   259   479
DIRIDL  000015   167E
DNF     052242   716   917   937   946   1414L  1870
DVD.CAP 000007   590L
DVD.DVD 000006   589L  1223
DVD.ENT 002000   599E  1314  1344  1345  1354  1358
DVD.INP 000023   595L
DVD.MNU 000011   592L
DVD.MUM 000010   591L
DVD.SET 000022   594L  1232
DVD.STE 000053   597E  1244
DVD.UFL 000012   593L
DVDFLV  000307   585E  1224  1233
EC.CNA  000004   188L
EC.DDA  000027   207L
EC.DIF  000017   199L
EC.DIW  000035   213L
EC.DNI  000045   221L
EC.DNR  000046   222L
EC.DNS  000005   189L
EC.DSC  000047   223L
EC.EOF  000001   185L
EC.EOM  000002   186L
EC.FAO  000031   209L
EC.FAP  000026   206L
EC.FL   000030   208L
EC.FNF  000014   196L
EC.FNO  000011   193L
EC.FNR  000034   212L
EC.FOD  000043   219L
EC.FUC  000013   195L
EC.ICN  000016   198L
EC.IDN  000006   190L
EC.IFC  000020   200L
EC.IFN  000007   191L
EC.ILC  000003   187L
EC.ILD  000040   216L  1812
EC.ILR  000012   194L
EC.ILV  000037   215L  1424  1898
EC.IOI  000052   226L
EC.IS   000032   210L
EC.NCV  000050   224L
EC.NEM  000021   201L
EC.NOS  000051   225L
EC.NPM  000044   220L
EC.NRD  000010   192L
EC.NVM  000042   218L
EC.OTL  000053   227L
EC.RF   000022   202L
EC.UNA  000036   214L
EC.UND  000015   197L
EC.UUN  000033   211L
EC.VPM  000041   217L
EC.WF   000023   203L
EC.WP   000025   205L
EC.WPV  000024   204L
ENL     000212   77E   811   818   825   835   997   1028  1042  1129  1152  1227
```

ENTRY	042333	657	759E											
ERR.IDS	043046	774	810L	1178	1183	1191	1193	1203						
ERR.ILO	043113	817L	1234											
ERR.IOV	043142	824L	918	921	924	938	941	943	947	950				
ERROR	043177	831L	856	1055	1212	1221	1245	1253	1289	1297	1349	1362	1474	
		1500	1519	1537	1543	1547	1588	1594	1601	1608				
ESC	000033	75E												
EXIT	043225	767	812	819	826	837	845L	1228						
FF	000014	78E												
FILI	000002	877	890E											
FLT.BOP	000011	540L												
FLT.CBD	000007	539L												
FLT.CDB	000006	538L												
FLT.CFC	000002	534L	967											
FLT.CRF	000003	535L	972											
FLT.CTY	000000	532L	908											
FLT.CWI	000001	533L	928											
FLT.MNC	000004	536L												
FLT.PBO	000013	543L	1061	1062										
FLT.SAL	000012	541L	1120											
FST.	042212	721L												
FST1	053243	1649L	1681											
FST2	053256	1656	1662L	1669										
FST3	053274	1664	1671L	1674										
FST4	053316	1666	1683L											
FT.ABS	000000	552E	654											
FT.BAC	000003	555E												
FT.DD	000001	238E												
FT.OC	000020	242E												
FT.OR	000002	239E												
FT.OU	000010	241E	1511											
FT.OW	000004	240E	1511											
FT.PIC	000001	553E												
FT.REL	000002	554E												
HELPI	000001	1067	1077E											
HOS1	050332	1126	1130L											
HOSBF	050156	1081	1096L											
HOSBFI	000002	1061	1062	1080E										
HOSHLP	050336	1078	1138L											
HOSPRC	050150	1052	1071L											
HOSSAL	050170	1075	1113L											
HOSSALI	000000	1064	1065	1074E										
HOSTAB	050052	1053	1058L											
HOSTABE	050147	1058	1069L											
I.CONFL	000004	368E	369											
I.CONTY	000001	355E	356											
I.CONWI	000003	361E	362											
I.CSLMD	000000	344E												
I.CUSOR	000002	358E	359											
IOC.CGN	000010	247L												
IOC.CSI	000011	248L												
IOC.DDA	000002	235L	243	257										
IOC.DES	000016	254L												
IOC.DEV	000020	255L												
IOC.DIL	000021	257E												
IOC.DIR	000023	259L												
IOC.DRL	000010	251E												
IOC.DTA	000014	253L												

```
IOC.FLG 000004   237L   251   1509
IOC.GRT 000005   245L
IOC.LGN 000012   249L
IOC.LNK 000000   234L   1507
IOC.LSI 000013   250L
IOC.SPG 000007   246L
IOC.SQL 000003   243E
IOC.UNI 000022   256L
IOCCTD  000001   263E   1505
IOCELEN 000052   261E
LAB.AUX 000117   649E   651
LAB.AXL 000001   651E
LAB.DAT 000000   626E
LAB.DIS 000003   622L
LAB.GRT 000005   623L
LAB.IND 000001   621L
LAB.LAB 000021   645L   646
LAB.LBL 000074   646E
LAB.NOD 000002   628E
LAB.PSS 000016   637L
LAB.RGT 000012   633L
LAB.SER 000000   620L   1603
LAB.SIZ 000014   636L
LAB.SPG 000007   624L
LAB.SPT 000117   650L
LAB.SYS 000001   627E
LAB.VER 000011   631L
LAB.VFL 000020   638L
LAB.VLT 000010   630L
LAB.VPL 000005   640E   642   643
LAB.VPR 000014   635E   640
LABEL   055072   1597   1603   2183L
LBD.    042223    736L
LBDA    053343   1712   1725L
LBD     052055   1238   1279L
LBD1    052123   1304L  1317   1328
LF      000012     65E
M.CDCA  000017    505L   959
M.CDLY  000016    504L   956   959
M.CFWA  000012    502L
M.CIN   000006    500L
M.CINT  000005    499L
M.CLWA  000014    503L
M.COUT  000010    501L
M.CPRE  000003    497L
M.CRUB  000004    498L
M.CSLC  000002    496L
M.SALO  000001    495L   1114
M.SUNI  000021    506L
M.SYDD  000022    507L
M.SYSM  000000    494L
MEML    055072    656   2174E
NL      000012     76E     77    811    818    825    833    980    980    980    980    981    982
                   983     984    985    986    987    988    989    990    991    992    993    994    995
                   996     997   1007   1007   1007   1008   1009   1010   1011   1012   1012   1013   1014
                  1015    1016   1016   1017   1018   1019   1020   1022   1023   1024   1025   1026   1128
                  1129    1139   1139   1139   1139   1143   1147   1148   1149   1150   1151   1152   1227
NUL2    000000     67E
```

```
NULL     000200      66E
OVL.IN   000001     412E
OVL.NUM  000014     414E
OVL.RES  000002     413E
OVL.UCS  000200     415E
PATCH    054372    2172L
PBF      054057     746    1833L
PBF.     042231     746L
PBO.DAT  000001     544E   1061   1061   1062
PBV      054076     751    1867L
PBV.     042234     751L
PBV1     054143    1887    1890L
PBV2     054147    1871    1874   1895L
PBV3     054151    1885    1889   1897L
PDF      051215     780    1175E
PDF1     051261    1188    1199L
PDF3     052005    1225    1232L
PDFA     052042    1208    1257L
PDFB     052046    1179    1184   1258L
PIC.COD  000006     580L    587   1492
PIC.ID   000000     575L
PIC.LEN  000002     577L   1279   1353
PIC.PTR  000004     578L   1301
PSZ      052263    1440L
PUBLIC   000001       1E   1121   1144
QUOTE    000047      73E
RMEML    060072     760   2189E
ROMBOOT  030000     273E
RSZ      052316    1443    1468L
RUBOUT   000177      69E
S.BAUD   040344     389L
S.BDA    041120     487L
S.BOOTF  041034     444L
S.CAADR  040333     372L
S.CACC   041006     428L
S.CCTAB  040335     373L
S.CDB    040343     386L
S.CFWA   040352     396L   1504
S.CODE   041007     429L
S.CONFL  040332     370L
S.CONTY  040327     357L    904    907
S.CONWI  040331     363L    925
S.CSLMD  040326     345L    356    359    362    369
S.CUSOR  040330     360L
S.DATC   040310     326L
S.DATE   040277     325L
S.DCS    041033     442L
S.DDDTA  040366     407L
S.DDGRP  040364     404L
S.DDLDA  040360     402L
S.DDLEN  040362     403L
S.DDOPC  040370     408L
S.DFWA   040354     397L
S.DIREA  041016     436L
S.DLINK  040346     394L    955   1115
S.FASER  041013     435L
S.FCI    041021     437L
S.GRTO   024000     269E
```

S.GRT1 025000 270E
S.GRT2 026000 271E
S.GUP 041027 439L
S.HIMEM 040316 328L
S.INT 040343 283L 382
S.JUMPS 041010 433L
S.MOUNT 041032 441L
S.OFWA 040350 395L
S.OMAX 040324 334L
S.OSN 041004 424L
S.OVLE 041000 421L
S.OVLFL 040371 417L
S.OVLS 040376 420L
S.OVSTK 041035 449L
S.RFWA 040356 398L
S.SCI 041024 438L
S.SCR 041121 488L
S.SDD 041010 434L
S.SOVR 041146 285L 287
S.SSN 041002 423L
S.SYSM 040320 330L
S.TIME 040312 327L
S.UCSF 040372 418L
S.UCSL 040374 419L
S.USRM 040322 332L
S.VAL 040277 282L 323
SECT 056072 2185L
SET1 042354 766 773L
SETA 043013 778 788L
SETAE 043045 788 797L
SETHLP 046364 784 1006L
SETHOS 050035 786 1052L
SETTT 043231 783 853L
SETVER 050006 785 1039L
SHO 052341 909 929 968 973 1103 1132 1492L
SHOA 053070 1496 1550L
SHOB 053105 1497 1551L
SKB 053114 681 1381 1415 1564L
SKB1 053113 1563L 1566
SNA. 042201 681L
SOP 054010 741 855 1054 1789L
SOP. 042226 741L
SOP1 054051 1792 1810L
SOP2 054052 1798 1811L
STACK 042200 289E 666
STACKL 001032 287E
STTCTY 044046 885 900E
STTFIL 044124 891 936L
STTWID 044066 888 916L
SYDD 040130 279E 1473 1586 1593 1600 1607
SYSCALL 000377 96E 761 836 1499 1518 1536 1542 1546
TAB 000011 74E 1040
TBL1 054175 2013L 2019
TBL2 054213 2011 2023L
TBL3 054215 2016 2027L
TBLS. 042215 726L
TTHLP 044224 894 979L
TTHLPI 000003 878 893E

TTOPRC	044036	853	882L	884	887	890	893
TTOTAB	043246	854	859L				
TTOTABE	044035	859	880L				
USERFWA	042200	290E	653	655	656		
VERS	000040	94E	1041	1041			
VFL.NSD	000001	639E					
WDD	052155	1249	1344L				
WIDI	000001	876	887E				
WSZ	053124	1455	1580L				
WTBLS.	042220	731L					

24094 BYTES FREE