```
000.001                 2  DEBUG   EQU     1               ASSEMBLE IN DEBUG MODE
000.001                 3  .MANUF. EQU     1               ASSEMBLE IN MANUFACTURE MODE
                        4
                        6  ***     SYSCMD - SYSTEM COMMAND PROCESSOR.
                        7  *
                        8  *       JGL, 12/06/1977
                        9  *
                       10  *       FOR HEATH COMPANY.
                       11  *
                       12  *       G. Chandler,    78/09   Maintenence release
                       13  *                       79/05   HDOS Version:  1.5
                       14  *
                       16  **      SYSCMD CAN BE ASSEMBLED IN THREE DIFFERENT MODES:
                       17  *
                       18  *       PRODUCTION MODE
                       19  *               NORMAL SYSCMD
                       20  *
                       21  *       DEBUG MODE
                       22  *               SOME EXTRA COMMANDS FOR SYSTEM DEVELOPMENT
                       23  *
                       24  *       MANUFACTURE MODE
                       25  *               AUTOMATICALLY LINKS TO 'MANDIAG.ABS' UPON ENTRY, UNLESS
                       26  *               40076-40077 CONTAINS 'GL'
                       27
                       28
                       29
000.000                30          XTEXT   FBDEF

                       32X **      FILE BLOCK DEFINITIONS.
                       33X
000.000                34X         ORG     0
000.000                35X FB.CHA  DS      1               CHANNEL NUMBER
000.001                36X FB.FLG  DS      1               FLAGS
000.002                37X FB.FWA  DS      2               BUFFER FWA
000.004                38X FB.PTR  DS      2               BUFFER POINTER
000.006                39X FB.LIM  DS      2               LIMIT OF DATA IN BUFFER (READ OPERATIONS)
000.010                40X FB.LWA  DS      2               LWA OF BUFFER
000.012                41X FB.NAM  DS      4+8+4+1         NAME OF FILE
000.021                42X FB.NAML EQU     *-FB.NAM
000.033                43X FBENL   EQU     *               ENTRY LENGTH
000.033                44          XTEXT   DIFDEF
```

```
                    46X **      DIRECTORY FILE FLAGS.
                    47X
000.200             48X DIF.SYS EQU     10000000B           SYSTEM FILE
000.100             49X DIF.LOC EQU     01000000B           LOCKED FOR CHANGE
000.040             50X DIF.WP  EQU     00100000B           WRITE PROTECTED
000.020             51X DIF.CNT EQU     00010000B           CONTIGUOUS FILE
                    52X
000.033             53          XTEXT   ASCII


                    55X **      ASCII CHARACTER EQUIVALENCES.
                    56X
000.015             57X CR      EQU     13                  CARRIAGE RETURN
000.012             58X LF      EQU     10                  LINE FEED
000.200             59X NULL    EQU     200Q                PAD CHARACTER
000.000             60X NUL2    EQU     0
000.007             61X BELL    EQU     7                   BELL CHARACTER
000.177             62X RUBOUT  EQU     177Q
000.010             63X BKSP    EQU     10Q                 CTL-H
000.026             64X C.SYN   EQU     26Q                 SYNC
000.002             65X C.STX   EQU     2                   STX
000.047             66X QUOTE   EQU     47Q
000.011             67X TAB     EQU     11Q
000.033             68X ESC     EQU     33Q
000.012             69X NL      EQU     12Q                 NEW LINE (HDOS SYSTEMS)
000.212             70X ENL     EQU     NL+200Q             NL + END-OF-LINE-FLAG
000.014             71X FF      EQU     14Q                 FORM FEED
000.001             72X CTLA    EQU     01Q                 CTL-A
000.002             73X CTLB    EQU     02Q                 CTL-B
000.003             74X CTLC    EQU     03Q                 CTL-C
000.004             75X CTLD    EQU     04Q                 CTL-D
000.017             76X CTLO    EQU     17Q                 CTL-O
000.020             77X CTLP    EQU     20Q                 CTL-P
000.021             78X CTLQ    EQU     21Q                 CTL-Q
000.023             79X CTLS    EQU     23Q                 CTL-S
000.032             80X CTLZ    EQU     32Q                 CTL-Z
000.033             81          XTEXT   08250


                    83X **      8250 UART CONTROL AND BIT DEFINITIONS.
                    84X
000.350             85X SC.ACE  EQU     350Q                SYSTEM CONSOLE PORT IF 8250 ACE
000.156             86X AC.DLY  EQU     110                 220 MIL. SEC. DELAY FOR 8250
                    87X
000.000             88X UR.RBR  EQU     0                   RECEIVER BUFFER REGISTER (READ ONLY)
                    89X
000.000             90X UR.THR  EQU     0                   TRANSMITTER HOLDING REGISTER (WRITE ONLY)
                    91X
000.000             92X UR.DLL  EQU     0                   DIVISOR LATCH (LEAST SIGNIFICANT)
                    93X
000.001             94X UR.DLM  EQU     1                   DIVISOR LATCH (MOST SIGNIFICANT)
                    95X
000.001             96X UR.IER  EQU     1                   INTERRUPT ENABLE REGISTER
000.001             97X UC.EDA  EQU     00000001B            ENABLE RECEIVED DATA AVAILABLE INTERRUPT
```

```
000.002                  98X UC.TRE  EQU     00000010B       ENABLE TRANSMIT HOLD REGISTER EMPTY INTERRUPT
000.004                  99X UC.RSI  EQU     00000100B       ENABLE RECEIVE STATUS INTERRUPT
000.010                 100X UC.MSI  EQU     00001000B       ENABLE MODEM STATUS INTERRUPT
                        101X
000.002                 102X UR.IIR  EQU     2               INTERRUPT IDENTIFICATION REGISTER
000.001                 103X UC.IIP  EQU     00000001B        INVERTED INTERRUPT PENDING (0 MEANS PENDING)
000.006                 104X UC.IID  EQU     00000110B        INTERRUPT ID
                        105X
000.003                 106X UR.LCR  EQU     3               LINE CONTROL REGISTER
000.000                 107X UC.5BW  EQU     00000000B        5 BIT WORDS
000.001                 108X UC.6BW  EQU     00000001B        6 BIT WORDS
000.002                 109X UC.7BW  EQU     00000010B        7 BIT WORDS
000.003                 110X UC.8BW  EQU     00000011B        8 BIT WORDS
000.004                 111X UC.2SB  EQU     00000100B        TWO STOP BITS SELECTED
000.010                 112X UC.PEN  EQU     00001000B        PARITY COMPUTATION ENABLED
000.020                 113X UC.EPS  EQU     00010000B        EVEN PARITY SELECT
000.040                 114X UC.SKP  EQU     00100000B        STICK PARITY
000.100                 115X UC.SB   EQU     01000000B        SET BREAK
000.200                 116X UC.DLA  EQU     10000000B        DIVISOR LATCH ACCESS
                        117X
000.004                 118X UR.MCR  EQU     4               MODEM CONTROL REGISTER
000.001                 119X UC.DTR  EQU     00000001B        DATA TERMINAL READY
000.002                 120X UC.RTS  EQU     00000010B        REQUEST TO SEND
000.004                 121X UC.OU1  EQU     00000100B        OUT 1
000.010                 122X UC.OU2  EQU     00001000B        OUT 2
000.020                 123X UC.LOO  EQU     00010000B        LOOP
                        124X
000.005                 125X UR.LSR  EQU     5               LINE STATUS REGISTER
000.001                 126X UC.DR   EQU     00000001B        DATA READY
000.002                 127X UC.OR   EQU     00000010B        OVERRUN
000.004                 128X UC.PE   EQU     00000100B        PARITY ERROR
000.010                 129X UC.FE   EQU     00001000B        FRAMING ERROR
000.020                 130X UC.BI   EQU     00010000B        BREAK INTERRUPT
000.040                 131X UC.THE  EQU     00100000B        TRANSMITTER HOLDING REGISTER EMPTY
000.100                 132X UC.TSE  EQU     01000000B        TRANSMITTER SHIFT REGISTER EMPTY
                        133X
000.006                 134X UR.MSR  EQU     6               MODEM STATUS REGISTER
000.001                 135X UC.DCS  EQU     00000001B        DELTA CLEAR TO SEND
000.002                 136X UC.DDR  EQU     00000010B        DELTA DATA SET READY
000.004                 137X UC.TER  EQU     00000100B        TRAILING EDGE OF RING
000.010                 138X UC.DRL  EQU     00001000B        DELTA RECEIVE LINE SIGNAL DETECT
000.020                 139X UC.CTS  EQU     00010000B        CLEAR TO SEND
000.040                 140X UC.DSR  EQU     00100000B        DATA SET READY
000.100                 141X UC.RI   EQU     01000000B        RING INDICATOR
000.200                 142X UC.RLS  EQU     10000000B        RECEIVED LINE SIGNAL DETECT
000.033                 143          XTEXT   U8251
```

```
                        146X **     8251 USART BIT DEFINITIONS.
                        147X *
                        148X
                        149X **     PORT ADDRESSES
                        150X
000.000                 151X UDR    EQU     0               DATA REGISTER IS EVEN
000.001                 152X USR    EQU     1               STATUS REGISTER IS NEXT
                        153X
000.372                 154X SC.UART EQU    372Q            CONSOLE USART ADDRESS (IFF 8251)
                        155X
                        156X
                        157X **     MODE INSTRUCTION CONTROL BITS.
                        158X
000.100                 159X UMI.1B EQU     01000000B       1 STOP BIT
000.200                 160X UMI.HB EQU     10000000B       1 1/2 STOP BITS
000.300                 161X UMI.2B EQU     11000000B       2 STOP BITS
000.040                 162X UMI.PE EQU     00100000B       EVEN PARITY
000.020                 163X UMI.PA EQU     00010000B       USE PARITY
000.000                 164X UMI.L5 EQU     00000000B       5 BIT CHARACTERS
000.004                 165X UMI.L6 EQU     00000100B       6 BIT CHARACTERS
000.010                 166X UMI.L7 EQU     00001000B       7 BIT CHARACTERS
000.014                 167X UMI.L8 EQU     00001100B       8 BIT CHARACTERS
000.001                 168X UMI.1X EQU     00000001B       CLOCK X 1
000.002                 169X UMI.16X EQU    00000010B       CLOCK X 16
000.003                 170X UMI.64X EQU    00000011B       CLOCK X 64
                        171X
                        172X **     COMMAND INSTRUCTION BITS.
                        173X
000.100                 174X UCI.IR EQU     01000000B       INTERNAL RESET
000.040                 175X UCI.RQ EQU     00100000B       READER-ON CONTROL FLAG
000.020                 176X UCI.ER EQU     00010000B       ERROR RESET
000.004                 177X UCI.RE EQU     00000100B       RECEIVE ENABLE
000.002                 178X UCI.IE EQU     00000010B       ENABLE INTERRUPTS FLAG
000.001                 179X UCI.TE EQU     00000001B       TRANSMIT ENABLE
                        180X
                        181X **     STATUS READ COMMAND BITS.
                        182X
000.040                 183X USR.FE EQU     00100000B       FRAMING ERROR
000.020                 184X USR.OE EQU     00010000B       OVERRUN ERROR
000.010                 185X USR.PE EQU     00001000B       PARITY ERROR
000.004                 186X USR.TXE EQU    00000100B       TRANSMITTER EMPTY
000.002                 187X USR.RXR EQU    00000010B       RECEIVER READY
000.001                 188X USR.TXR EQU    00000001B       TRANSMITTER READY
000.033                 189         XTEXT   MTR
```

```
                        192X **     MTR - PAM/8 EQUIVALENCES.
                        193X *
                        194X *      THIS DECK CONTAINS SYMBOLIC DEFINITIONS USED TO
                        195X *      MAKE USE OF THE PAM/8 CODE AND CONTROL BYTES.


                        197X **     IO PORTS
                        198X
    000.360             199X IP.PAD EQU     360Q            PAD INPUT PORT
    000.360             200X OP.CTL EQU     360Q            CONTROL OUTPUT PORT
    000.360             201X OP.DIG EQU     360Q            DIGIT SELECT OUTPUT PORT
    000.361             202X OP.SEG EQU     361Q            SEGMENT SELECT OUTPUT PORT


                        204X **     FRONT PANEL CONTROL BITS.
                        205X
    000.020             206X CB.SSI EQU     00010000B       SINGLE STEP INTERRUPT
    000.040             207X CB.MTL EQU     00100000B       MONITOR LIGHT
    000.100             208X CB.CLI EQU     01000000B       CLOCK INTERRUPT ENABLE
    000.200             209X CB.SPK EQU     10000000B       SPEAKER ENABLE


                        211X **     MONITOR MODE FLAGS.
                        212X
    000.000             213X DM.MR  EQU     0               MEMORY READ
    000.001             214X DM.MW  EQU     1               MEMORY WRITE
    000.002             215X DM.RR  EQU     2               REGISTER READ
    000.003             216X DM.RW  EQU     3               REGISTER WRITE


                        218X **     USER OPTION BITS.
                        219X *
                        220X *      THESE BITS ARE SET IN CELL .MFLAG.
                        221X
    000.200             222X UO.HLT EQU     10000000B       DISABLE HALT PROCESSING
    000.100             223X UO.NFR EQU     CB.CLI          NO REFRESH OF FRONT PANEL
    000.002             224X UO.DDU EQU     00000010B       DISABLE DISPLAY UPDATE
    000.001             225X UO.CLK EQU     00000001B       ALLOW PRIVATE INTERRUPT PROCESSING


                        227X **     MONITOR IDENTIFICATION FLAGS
                        228X *
                        229X *      THESE BYTES IDENTIFY THE ROM MONITOR.
                        230X *      THEY ARE THE VARIOUS VALUES OF LOCATION .IDENT
                        231X
    000.021             232X M.PAM8 EQU     021Q            'LXI' INSTRUCTION AT 000.000 IN PAM-8
    000.303             233X M.FOX  EQU     303Q            'JMP' INSTRUCTION AT 000.000 IN FOX ROM
```

```
                        235X **     ROUTINE ENTRY POINTS.
                        236X *
                        237X
000.000                 238X .IDENT  EQU     0000A       IDENTIFICATION LOCATION
000.053                 239X .DLY    EQU     0053A       DELAY
001.267                 240X .LOAD   EQU     1267A       TAPE LOAD
001.374                 241X .DUMP   EQU     1374A       TAPE DUMP
002.136                 242X .ALARM  EQU     2136A       ALARM ROUTINE
002.140                 243X .HORN   EQU     2140A       HORN
002.172                 244X .CTC    EQU     2172A       CHECK TAPE CHECKSUM
002.205                 245X .TPERR  EQU     2205A       TAPE ERROR ROUTINE
002.264                 246X .PCHL   EQU     2264A       PCHL INSTRUCTION
002.265                 247X .SRS    EQU     2265A       SCAN RECORD START
002.325                 248X .RNP    EQU     2325A       READ NEXT PAIR
002.331                 249X .RNB    EQU     2331A       READ NEXT BYTE
002.347                 250X .CRC    EQU     2347A       CRC-16 CALCULATOR
003.017                 251X .WNP    EQU     3017A       WRITE NEXT PAIR
003.024                 252X .WNB    EQU     3024A       WRITE NEXT BYTE
003.122                 253X .DOD    EQU     3122A       DECODE FOR OCTAL DISPLAY
003.260                 254X .RCK    EQU     3260A       READ CONSOLE KEYSET
003.356                 255X .DODA   EQU     3356A       SEGMENT CODE TABLE

                        257X **     RAM CELLS USED BY H8MTR.
                        258X *
                        259X
040.000                 260X .START  EQU     40000A      START DUMP ADDRESS
040.002                 261X .IOWRK  EQU     40002A      IN OR OUT INSTRUCTION
040.005                 262X .REGI   EQU     40005A      DISPLAYED REGISTER INDEX
040.006                 263X .DSPROT EQU     40006A      PERIOD FLAG BYTE
040.007                 264X .DSPMOD EQU     40007A      DISPLAY MODE
040.010                 265X .MFLAG  EQU     40010A      USER OPTION BYTE
040.011                 266X .CTLFLG EQU     40011A      PANEL CONTROL BYTE
040.013                 267X .ALEDS  EQU     40013A      ABUSS LEDS
040.021                 268X .DLEDS  EQU     40021A      DBUSS LEDS
040.024                 269X .ABUSS  EQU     40024A      ABUSS REGISTER
040.027                 270X .CRCSUM EQU     40027A      CRCSUM WORD
040.031                 271X .TPERRX EQU     40031A      TAPE ERROR EXIT VECTOR
040.033                 272X .TICCNT EQU     40033A      CLOCK TICK COUNTER
040.035                 273X .REGPTR EQU     40035A      REGISTER POINTER
040.037                 274X .UIVEC  EQU     40037A      USER INTERRUPT VECTORS
000.033                 275          XTEXT   ECDEF

                        277X **     ERROR CODE DEFINITIONS.
                        278X
000.000                 279X         ORG     0
000.000                 280X         DS      1           NO ERROR #0
000.001                 281X EC.EOF  DS      1           END OF FILE
000.002                 282X EC.EOM  DS      1           END OF MEDIA
000.003                 283X EC.ILC  DS      1           ILLEGAL SYSCALL CODE
000.004                 284X EC.CNA  DS      1           CHANNEL NOT AVAILABLE
000.005                 285X EC.DNS  DS      1           DEVICE NOT SUITABLE
000.006                 286X EC.IDN  DS      1           ILLEGAL DEVICE NAME
```

```
000.007                 287X EC.IFN  DS    1           ILLEGAL FILE NAME
000.010                 288X EC.NRD  DS    1           NO ROOM FOR DEVICE DRIVER
000.011                 289X EC.FNO  DS    1           CHANNEL NOT OPEN
000.012                 290X EC.ILR  DS    1           ILLEGAL REQUEST
000.013                 291X EC.FUC  DS    1           FILE USAGE CONFLICT
000.014                 292X EC.FNF  DS    1           FILE NAME NOT FOUND
000.015                 293X EC.UND  DS    1           UNKNOWN DEVICE
000.016                 294X EC.ICN  DS    1           ILLEGAL CHANNEL NUMBER
000.017                 295X EC.DIF  DS    1           DIRECTORY FULL
000.020                 296X EC.IFC  DS    1           ILLEGAL FILE CONTENTS
000.021                 297X EC.NEM  DS    1           NOT ENOUGH MEMORY
000.022                 298X EC.RF   DS    1           READ FAILURE
000.023                 299X EC.WF   DS    1           WRITE FAILURE
000.024                 300X EC.WPV  DS    1           WRITE PROTECTION VIOLATION
000.025                 301X EC.WP   DS    1           DISK WRITE PROTECTED
000.026                 302X EC.FAP  DS    1           FILE ALREADY PRESENT
000.027                 303X EC.DDA  DS    1           DEVICE DRIVER ABORT
000.030                 304X EC.FL   DS    1           FILE LOCKED
000.031                 305X EC.FAO  DS    1           FILE ALREADY OPEN
000.032                 306X EC.IS   DS    1           ILLEGAL SWITCH
000.033                 307X EC.UUN  DS    1           UNKNOWN UNIT NUMBER
000.034                 308X EC.FNR  DS    1           FILE NAME REQUIRED
000.035                 309X EC.DIW  DS    1           DEVICE IS NOT WRITABLE (OR WRITE LOCKED)
000.036                 310X EC.UNA  DS    1           UNIT NOT AVAILABLE
000.037                 311X EC.ILV  DS    1           ILLEGAL VALUE
000.040                 312X EC.ILO  DS    1           ILLEGAL OPTION
000.041                 313X EC.VPM  DS    1           VOLUME PRESENTLY MOUNTED ON DEVICE
000.042                 314X EC.NVM  DS    1           NO VOLUME PRESENTLY MOUNTED
000.043                 315X EC.FOD  DS    1           FILE OPEN ON DEVICE
000.044                 316X EC.NPM  DS    1           NO PROVISIONS MADE FOR REMOUNTING MORE DISKS
000.045                 317X EC.DNI  DS    1           DISK NOT INITIALIZED
000.046                 318X EC.DNR  DS    1           DISK IS NOT READABLE
000.047                 319X EC.DSC  DS    1           DISK STRUCTURE IS CORRUPT
000.050                 320X EC.NCV  DS    1           NOT CORRECT VERSION OF HDOS
000.051                 321X EC.NOS  DS    1           NO OPERATING SYSTEM MOUNTED
000.052                 322X EC.IOI  DS    1           ILLEGAL OVERLAY INDEX
000.053                 323X EC.OTL  DS    1           OVERLAY TO LARGE
000.054                 324          XTEXT OVLDEF


                        326X **      OVERLAY TABLE ENTRYS.
                        327X
000.000                 328X         ORG   0
                        329X
000.000                 330X OVL.COD DS    2           FIRST SECTOR OF OVERLAY CODE
000.002                 331X OVL.SIZ DS    2           OVERLAY SIZE
000.004                 332X OVL.ENT DS    2           OVERLAY ENTRY POINT
000.006                 333X OVL.FLB DS    1           OVERLAY FLAG BYTE
000.007                 334X         DS    1           DUMMY BYTE TO ROUND TABLE SIZE UP TO 8
000.010                 335X OVL.ENS EQU   *           OVERLAY ENTRY SIZE
                        336X
                        337X *       OVERLAY INDICES
                        338X
000.000                 339X         ORG   0
```

```
                    340X
000.000             341X OVL0    DS    1
000.001             342X OVL1    DS    1
000.002             343          XTEXT DDFDEF


                    345X **      DIRECTORY DEVICE FORMAT DEFINITION.
                    346X *
                    347X
                    348X
000.002             349X HOS.SPG EQU   2          2 SECTORS PER GROUP REQUIRED FOR NOW
                    350X
000.000             351X         ORG   0
000.000             352X DDF.BOO DS    9          2K BOOT PROGRAM
000.011             353X DDF.BOL EQU   *          LENGTH OF BOOT
000.011             354X DDF.LAB DS    1          LABEL SECTOR
000.012             355X DDF.RGT DS    2          RESERVED GROUP TABLE
000.014             356X DDF.USR DS    0          BEGINNING OF OPEN SPACE
000.014             357          XTEXT DIRDEF


                    359X **      DIRECTORY ENTRY FORMAT.
                    360X
000.000             361X         ORG   0
                    362X
                    363X
000.377             364X DF.EMP  EQU   377Q         FLAGS ENTRY EMPTY
000.376             365X DF.CLR  EQU   376Q         FLAGS ENTRY EMPTY, REST OF DIR ALSO CLEAR
                    366X
000.000             367X DIR.NAM DS    8          NAME
000.010             368X DIR.EXT DS    3          EXTENSION
000.013             369X DIR.PRO DS    1          PROJECT
000.014             370X DIR.VER DS    1          VERSION
000.015             371X DIRIDL  EQU   *          FILE IDENTIFICATION LENGTH
                    372X
000.015             373X DIR.CLU DS    1          CLUSTER FACTOR
000.016             374X DIR.FLG DS    1          FLAGS
000.017             375X         DS    1          RESERVED
000.020             376X DIR.FGN DS    1          FIRST GROUP NUMBER
000.021             377X DIR.LGN DS    1          LAST GROUP NUMBER
000.022             378X DIR.LSI DS    1          LAST SECTOR INDEX (IN LAST GROUP)
000.023             379X DIR.CRD DS    2          CREATION DATE
000.025             380X DIR.ALD DS    2          LAST ALTERATION DATE
                    381X
000.027             382X DIRELEN EQU   *          DIRECTORY ENTRY LENGTH
000.027             383          XTEXT LABDEF
```

```
                        385X **     DISK LABEL SECTOR FORMATS.
                        386X
000.000                 387X        ORG     0
000.000                 388X LAB.SER DS     1           SERIAL NUMBER OF VOLUME
000.001                 389X LAB.IND DS     2           INITIALIZATION DATE
000.003                 390X LAB.DIS DS     2           SECTOR NUMBER OF 1ST DIRECTORY SECTOR
000.005                 391X LAB.GRT DS     2           INDEX OF GRT SECTOR
000.007                 392X LAB.SPG DS     1           SECTORS PER GROUP
                        393X
000.000                 394X LAB.DAT EQU    0           DATA VOLUME ONLY
000.001                 395X LAB.SYS EQU    1           SYSTEM VOLUME
000.002                 396X LAB.NOD EQU    2           => LAB.NOD MEANS VOLUME HAS NO DIRECTORY
                        397X
000.010                 398X LAB.VLT DS     1           VOLUME TYPE
000.011                 399X LAB.VER DS     1           VERSION OF INIT17 THAT INITED DISK
000.012                 400X        DS      7           UNUSED
000.021                 401X LAB.LAB DS     60          LABEL
000.074                 402X LAB.LBL EQU    *-LAB.LAB   LABEL LENGTH
000.115                 403         XTEXT   DISDEF


                        405X **     DIRECTORY BLOCK FORMAT.
                        406X
000.000                 407X        ORG     0
                        408X
000.000                 409X DIS.ENT EQU    *           FIRST ENTRY ADDRESS
000.000                 410X        DS      22*DIRELEN  22 DIRECTORY ENTRYS PER BLOCK
001.372                 411X        DS      1           0 BYTE = END OF ENTRYS IN THIS BLOCK
                        412X
001.373                 413X        ORG     512-5       AT END OF BLOCK
001.373                 414X DIS.ENL DS     1           LENGTH OF EACH ENTRY (=DIRELEN)
001.374                 415X DIS.SEC DS     2           BLOCK # OF THIS BLOCK,
001.376                 416X DIS.LNK DS     2           BLOCK # OF NEXT BLOCK, =0 IF THIS IS LAST
002.000                 417         XTEXT   FILDEF


                        419X **     FILDEF - FILE TYPE DEFINITIONS.
                        420X *
                        421X *      DB      377Q,FT.XXX
                        422X
                        423X
000.000                 424X FT.ABS  EQU    0           ABSOLUTE BINARY
000.001                 425X FT.PIC  EQU    1           POSITION INDEPENDANT CODE
000.002                 426X FT.REL  EQU    2           RELOCATABLE CODE
000.003                 427X FT.BAC  EQU    3           COMPILED BASIC CODE
002.000                 428         XTEXT   DDDEF
```

```
                    430X **     DEVICE DRIVER COMMUNICATION FLAGS.
                    431X *
                    432X
000.000             433X        ORG     0
                    434X
000.000             435X DC.REA DS      1           READ
000.001             436X DC.WRI DS      1           WRITE
000.002             437X DC.RER DS      1           READ REGARDLESS
000.003             438X DC.OPR DS      1           OPEN FOR READ
000.004             439X DC.OPW DS      1           OPEN FOR WRITE
000.005             440X DC.OPU DS      1           OPEN FOR UPDATE
000.006             441X DC.CLO DS      1           CLOSE
000.007             442X DC.ABT DS      1           ABORT
000.010             443X DC.MOU DS      1           MOUNT DEVICE
000.011             444X DC.LOD DS      1           LOAD DEVICE DRIVER
000.012             445X DC.MAX DS      1           MAXIMUM ENTRY INDEX
000.013             446         XTEXT   HOSDEF


                    448X **     HOSDEF - DEFINE HOS PARAMETER.
                    449X *
                    450X
                    451X
000.026             452X VERS   EQU     1*16+6      VERSION 1.6
                    453X
000.377             454X SYSCALL EQU    377Q        SYSCALL INSTRUCTION
                    455X
                    456X
000.000             457X        ORG     0
                    458X
                    459X *      RESIDENT FUNCTIONS
                    460X
000.000             461X .EXIT  DS      1           EXIT (MUST BE FIRST)
000.001             462X .SCIN  DS      1           SCIN
000.002             463X .SCOUT DS      1           SCOUT
000.003             464X .PRINT DS      1           PRINT
000.004             465X .READ  DS      1           READ
000.005             466X .WRITE DS      1           WRITE
000.006             467X .CONSL DS      1           SET/CLEAR CONSOLE OPTIONS
000.007             468X .CLRCO DS      1           CLEAR CONSOLE BUFFER
000.010             469X .LOADO DS      1           LOAD AN OVERLAY
000.011             470X .VERS  DS      1           RETURN HDOS VERSION NUMBER
000.012             471X .SYSRES DS     1           PRECEDING FUNCTIONS ARE RESIDENT
                    472X
                    473X
                    474X *      *HDOSOVLO.SYS*  FUNCTIONS
                    475X
000.040             476X        ORG     40A
                    477X
000.040             478X .LINK  DS      1           LINK  (MUST BE FIRST)
000.041             479X .CTLC  DS      1           CTL-C
000.042             480X .OPENR DS      1           OPENR
000.043             481X .OPENW DS      1           OPENW
000.044             482X .OPENU DS      1           OPENU
000.045             483X .OPENC DS      1           OPENC
```

```
000.046                 484X .CLOSE  DS    1           CLOSE
000.047                 485X .POSIT  DS    1           POSITION
000.050                 486X .DELET  DS    1           DELETE
000.051                 487X .RENAM  DS    1           RENAME
000.052                 488X .SETTP  DS    1           SETTOP
000.053                 489X .DECODE DS    1           NAME DECODE
000.054                 490X .NAME   DS    1           GET FILE NAME FROM CHANNEL
000.055                 491X .CLEAR  DS    1           CLEAR CHAN
000.056                 492X .CLEARA DS    1           CLEAR ALL CHANS
000.057                 493X .ERROR  DS    1           LOOKUP ERROR
000.060                 494X .CHFLG  DS    1           CHANGE FLAGS
000.061                 495X .DISMT  DS    1           FLAG SYSTEM DISK DISMOUNTED
000.062                 496X .LOADD  DS    1           LOAD DEVICE DRIVER
                        497X
                        498X
                        499X *       *HDOSOVL1.SYS*  FUNCTIONS
                        500X
000.200                 501X         ORG   200Q
                        502X
000.200                 503X .MOUNT  DS    1           MOUNT  (MUST BE FIRST)
000.201                 504X .DMOUN  DS    1           DISMOUNT
000.202                 505X .MONMS  DS    1           MOUNT/NO MESSAGE
000.203                 506X .DMNMS  DS    1           DISMOUNT/NO MESSAGE
000.204                 507X .RESET  DS    1           RESET = DISMOUNT/MOUNT OF UNIT
000.205                 508          XTEXT HOSEQU


                        510X **      HDOS SYSTEM EQUIVALENCES.
                        511X *
                        512X
024.000                 513X S.GRT0  EQU   24000A        SYSTEM AREA FOR  GRT0
025.000                 514X S.GRT1  EQU   25000A        SYSTEM AREA FOR  GRT1
026.000                 515X S.GRT2  EQU   26000A        SYSTEM AREA FOR  GRT2
                        516X
030.000                 517X ROMBOOT EQU   30000A        ROM BOOT ENTRY
                        518X
040.100                 519X         ORG   40100A        FREE SPACE FROM PAM-8
                        520X
040.100                 521X         DS    8           JUMP TO SYSTEM EXIT
040.110                 522X D.CON   DS    16          DISK CONSTANTS
040.130                 523X SYDD    EQU   *           SYSTEM DISK ENTRY POINT
040.130                 524X D.VEC   DS    24*3        SYSTEM ROM ENTRY VECTORS
040.240                 525X D.RAM   DS    31          SYSTEM ROM WORK AREA
040.277                 526X S.VAL   DS    36          SYSTEM VALUES
040.343                 527X S.INT   DS    115         SYSTEM INTERNAL WORK AREAS
041.126                 528X         DS    16
041.146                 529X S.SOVR  DS    2           STACK OVERFLOW WARNING
041.150                 530X         DS    42200A-*    SYSTEM STACK
001.032                 531X STACKL  EQU   *-5.SOVR    STACK SIZE
                        532X
042.200                 533X STACK   EQU   *           LWA+1 SYSTEM STACK
042.200                 534X USERFWA EQU   *           USER FWA
042.200                 535          XTEXT EDCON
```

```
                        537X **     D.CON DETAILED EQUIVALENCES.
                        538X *
                        539X *      HOSEQU MUST BE MODIFIED WHEN THIS TABLE IS MODIFIED.
                        540X
040.110                 541X        ORG     D.CON
                        542X
040.110                 543X D.XITA DS      2           SEE SYSTEM ROM FOR DESCRIPTION
040.112                 544X D.WRITA DS     1
040.113                 545X D.WRITB DS     1
040.114                 546X D.WRITC DS     1
040.115                 547X D.MAIA DS      1
040.116                 548X D.LPSA DS      1
040.117                 549X D.SDFA DS      1
040.120                 550X D.SDFB DS      1
040.121                 551X D.STSA DS      1
040.122                 552X D.STSB DS      1
040.123                 553X D.WHDA DS      1
040.124                 554X D.WNHA DS      1
040.125                 555X D.WSCA DS      1
                        556X
040.126                 557X D.ERTS DS      2           TRACK AND SECTOR OF LAST DISK ERRORS
040.130                 558         XTEXT   EDRAM



                        560X **     EDRAM - DISK RAM WORKAREA DEFINITION.
                        561X *
                        562X *      ZEROED UPON BOOTING UP.
                        563X *
                        564X *      HOSEQU MUST BE CHANGED WHEN THIS DECK IS CHANGED.
                        565X
                        566X
040.240                 567X        ORG     D.RAM
                        568X
040.240                 569X D.TT   DS      1           TARGET TRACK (CURRENT OPERATION)
040.241                 570X D.TS   DS      1           TARGET SECTOR (CURRENT OPERATION)
                        571X
040.242                 572X D.DVCTL DS     1           DEVICE CONTROL BYTE
                        573X
040.243                 574X D.DLYMO DS     1           MOTOR ON DELAY COUNT
040.244                 575X D.DLYHS DS     1           HEAD SETTLE DELAY COUNTER
                        576X
040.245                 577X D.TRKPT DS     2           ADDRESS IN D.DRVTB FOR TRACK NUMBER
040.247                 578X D.VOLPT DS     2           ADDRESS IN D.DRVTB FOR VOLUME NUMBER
                        579X
040.251                 580X D.DRVTB DS     2*4         TRACK NUMBER AND VOLUME NUMBER FOR 4 DRIVES
                        581X
040.261                 582X D.HECNT DS     1           HARD ERROR COUNT
040.262                 583X D.SECNT DS     2           SOFT ERROR COUNT
040.264                 584X D.OECNT DS     1           OPERATION ERROR COUNT
                        585X
                        586X *      GLOBAL DISK ERROR COUNTERS
                        587X
040.265                 588X D.ERR  DS      0           BEGINNING OF ERROR BLOCK
040.265                 589X D.E.MDS DS     1           MISSING DATA SYNC
```

```
040.266  596X D.E.HSY DS   1          MISSING HEADER SYNC
040.267  591X D.E.CHK DS   1          DATA CHECKSUM
040.270  592X D.E.HCK DS   1          HEADER CHECKSUM
040.271  593X D.E.VOL DS   1          WRONG VOLUME NUMBER
040.272  594X D.E.TRK DS   1          BAD TRACK SEEK
040.273  595X D.ERRL  DS   0          LIMIT OF ERROR COUNTERS
         596X
         597X *       I/O OPERATION COUNTS
         598X
040.273  599X D.OPR   DS   2
040.275  600X D.OPW   DS   2
         601X
000.037  602X D.RAML  EQU  *-D.RAM
040.277  603         XTEXT ESVAL


         605X **      S.VAL - SYSTEM VALUE DEFINTIONS.
         606X *
         607X *       THESE VALUES ARE SET AND MAINTAINED BY THE SYSTEM.
         608X *
         609X *       THE DECK HOSEQU MUST BE MODIFIED WHEN THIS IS MODIFIED.
         610X
         611X
040.277  612X         ORG   S.VAL
         613X
040.277  614X S.DATE  DS   9          SYSTEM DATE (IN ASCII)
040.310  615X S.DATC  DS   2          CODED DATE
040.312  616X S.TIME  DS   4          TIME FROM MIDNIGHT (IN TICS)
040.316  617X S.HIMEM DS   2          HARDWARE HIGH MEMORY ADRESS+1
         618X
040.320  619X S.SYSM  DS   2          FWA RESIDENT SYSTEM
         620X
040.322  621X S.USRM  DS   2          LWA USER MEMORY
         622X
040.324  623X S.OMAX  DS   2          MAX OVERLAY SIZE FOR SYSTEM
         624X
         625X
         626X **      THE FOLLOWING FIVE CELLS SHOULD BE MODIFIED/READ ONLY VIA THE .CONSL SYSCALL
         627X
000.200  628X CSL.ECH EQU  10000000B  SUPPRESS ECHO
000.002  629X CSL.WRP EQU  00000010B  WRAP LINES AT WIDTH
000.001  630X CSL.CHR EQU  00000001B  OPERATE IN CHARACTER MODE
         631X
000.000  632X I.CSLMD EQU  0          S.CSLMD IS FIRST BYTE
040.326  633X S.CSLMD DS   1          CONSOLE MODE
         634X
000.200  635X CTP.BKS EQU  10000000B  TERMINAL PROCESSES BACKSPACES
000.040  636X CTP.MLI EQU  00100000B  MAP LOWER CASE TO UPPER ON INPUT
000.020  637X CTP.MLO EQU  00010000B  MAP LOWER CASE TO UPPER ON OUTPUT
000.010  638X CTP.2SB EQU  00001000B  TERMINAL NEEDS TWO STOP BITS
000.002  639X CTP.BKM EQU  00000010B  MAP BKSP (UPON INPUT) TO RUBOUT
000.001  640X CTP.TAB EQU  00000001B  TERMINAL SUPPORTS TAB CHARACTERS
         641X
000.001  642X I.CONTY EQU  1          S.CONTY IS 2ND BYTE
```

```
000.000                 643X            ERRNZ   *-S.CSLMD-I.CONTY
040.327                 644X S.CONTY    DS      1               CONSOLE TYPE FLAGS
000.002                 645X I.CUSOR    EQU     2               S.CUSOR IS 3RD BYTE
000.000                 646X            ERRNZ   *-S.CSLMD-I.CUSOR
040.330                 647X S.CUSOR    DS      1               CURRENT CURSOR POSITION
000.003                 648X I.CONWI    EQU     3               S.CONWI IS 4TH BYTE
000.000                 649X            ERRNZ   *-S.CSLMD-I.CONWI
040.331                 650X S.CONWI    DS      1               CONSOLE WIDTH
                        651X
000.001                 652X CO.FLG     EQU     00000001B       CTL-O FLAG
000.200                 653X CS.FLG     EQU     10000000B       CTL-S FLAG
                        654X
000.004                 655X I.CONFL    EQU     4               S.CONFL IS 5TH BYTE
000.000                 656X            ERRNZ   *-S.CSLMD-I.CONFL
040.332                 657X S.CONFL    DS      1               CONSOLE FLAGS
                        658X
040.333                 659X S.CAADR    DS      2               ADDRESS FOR ABORT PROCESSING (>256 IF VALID)
040.335                 660X S.CCTAB    DS      6               ADDR FOR CTL-A, CTL-B, CTL-C PROCESSING
040.343                 661             XTEXT   ESINT


                        663X **         S.INT - SYSTEM INTERNAL WORKAREA DEFINITIONS.
                        664X *
                        665X *          THESE CELLS ARE REFERENCED BY OVERLAYS AND MAIN CODE, AND
                        666X *          MUST THEREFORE RESIDE IN FIXED LOW MEMORY.
                        667X
                        668X
040.343                 669X            ORG     S.INT
                        670X
                        671X **         CONSOLE STATUS FLAGS
                        672X
040.343                 673X S.CDB      DS      1               CONSOLE DESCRIPTOR BYTE
000.000                 674X CDB.H85    EQU     00000000B
000.001                 675X CDB.H84    EQU     00000001B       =0 IF H8-5, =1 IF H8-4
040.344                 676X S.BAUD     DS      2               [0-14]  H8-4 BAUD RATE, =0 IF H8-5
                        677X *                                  [15]    =1 IF BAUD RATE => 2 STOP BITS
                        678X
                        679X **         TABLE ADDRESS WORDS
                        680X
040.346                 681X S.DLINK    DS      2               ADDRESS OF DATA IN HDOS CODE
040.350                 682X S.OFWA     DS      2               FWA  OVERLAY  TABLE
040.352                 683X S.CFWA     DS      2               FWA  CHANNEL  TABLE
040.354                 684X S.DFWA     DS      2               FWA  DEVICE   TABLE
040.356                 685X S.RFWA     DS      2               FWA  RESIDENT HDOS CODE
                        686X
                        687X **         DEVICE DRIVER DELAYED LOAD FLAGS
                        688X
040.360                 689X S.DDLDA    DS      2               DRIVER LOAD ADDRESS (HIGH BYTE=0 IF NO LOAD PENDING)
040.362                 690X S.DDLEN    DS      2               CODE LENGTH IN BYTES
040.364                 691X S.DDGRP    DS      1               GROUP NUMBER FOR DRIVER
040.365                 692X            DS      1               HOLD PLACE
                        693X *S.DDSEC           DS      2               SECTOR NUMBER FOR DRIVER ( * OBSOLETE ! * )
040.366                 694X S.DDDTA    DS      2               DEVICE'S ADDRESS IN DEVLST +DEV.RES
040.370                 695X S.DDOPC    DS      1               OPEN OPCODE PENDEDING
```

```
                        696X
                        697X **     OVERLAY MANAGEMENT FLAGS
                        698X
    000.001             699X OVL.IN  EQU     00000001B       IN MEMORY
    000.002             700X OVL.RES EQU     00000010B       PERMINANTLY RESIDENT
    000.014             701X OVL.NUM EQU     00001100B       OVERLAY NUMBER MASK
    000.200             702X OVL.UCS EQU     10000000B       USER CODE SWAPPED FOR OVERLAY
                        703X
    040.371             704X S.OVLFL DS      1               OVERLAY FLAG
    040.372             705X S.UCSF  DS      2               FWA SWAPPED USER CODE
    040.374             706X S.UCSL  DS      2               LENGTH SWAPPED USER CODE
    040.376             707X S.OVLS  DS      2               SIZE OF OVERLAY CODE
    041.000             708X S.OVLE  DS      2               ENTRY POINT OF OVERLAY CODE
                        709X
    041.002             710X S.SSN   DS      2               SWAP AREA SECTOR NUMBER
    041.004             711X S.OSN   DS      2               OVERLAY SECTOR NUMBER
                        712X
                        713X *      SYSCALL PROCESSING WORK AREAS
                        714X
    041.006             715X S.CACC  DS      1               (ACC) UPON SYSCALL
    041.007             716X S.CODE  DS      1               SYSCALL INDEX IN PROGRESS
                        717X
                        718X *      JUMPS TO ROUTINES IN RESIDENT HDOS CODE
                        719X
    041.010             720X S.JUMPS DS      0               START OF JUMP VECTORS
    041.010             721X S.SDD   DS      3               JUMP TO STAND-IN DEVICE DRIVER
    041.013             722X S.FASER DS      3               JUMP TO FATSERR (FATAL SYSTEM ERROR)
    041.016             723X S.DIREA DS      3               JUMP TO DIREAD (DISK FILE READ)
    041.021             724X S.FCI   DS      3               JUMP TO FCI (FETCH CHANNEL INFO)
    041.024             725X S.SCI   DS      3               JUMP TO SCI (STORE CHANNEL INFO)
    041.027             726X S.GUP   DS      3               JUMP TO GUP (GET UNIT POINTER)
                        727X
    041.032             728X S.MOUNT DS      1               <>0 IF THE SYSTEM DISK IS MOUNTED
    041.033             729X S.DCS   DS      1               DEFAULT CLUSTER SIZE-1
                        730X
    041.034             731X S.BOOTF DS      1               BOOT FLAGS
    000.001             732X BOOT.P  EQU     00000001B       EXECUTE PROLOGUE UPON BOOTUP
                        733X
                        734X *      STACK VALUE SAVED FOR OVERLAY SYSCALLS
                        735X
    041.035             736X S.OVSTK DS      2               VALUE OF SP UPON SYSCALLS USING OVERLAY
                        737X
    041.037             738X         DS      1               RESERVED


                        740X **     ACTIVE I/O AREA.
                        741X *
                        742X *      THE AIO.XXX AREA CONTAINS INFORMATION ABOUT THE I/O OPERATION
                        743X *      CURRENTLY BEING PERFORMED.  THE INFORMATION IS OBTAINED FROM
                        744X *      THE CHANNEL TABLE, AND WILL BE RESTORED THERE WHEN DONE.
                        745X *
                        746X *      NORMALLY, THE AIO.XXX INFORMATION WOULD BE OBTAINED DIRECTLY
                        747X *      FROM VARIOUS SYSTEM TABLES VIA POINTER REGISTERS.  SINCE THE
                        748X *      8080 HAS NO GOOD INDEXED ADDRESSING, THE DATA IS MANUALLY
```

```
                        749X *      COPIED INTO THE AIO.XXX CELLS BEFORE PROCESSING, AND
                        750X *      BACKDATED AFTER PROCESSING.
                        751X
041.040                 752X AIO.VEC DS     3           JUMP INSTRUCTION
041.041                 753X AIO.DDA EQU    *-2         DEVICE DRIVER ADDRESS
041.043                 754X AIO.FLG DS     1           FLAG BYTE
041.044                 755X AIO.GRT DS     2           ADDRESS OF GROUP RESERV TABLE
041.046                 756X AIO.SPG DS     1           SECTORS PER GROUP
041.047                 757X AIO.CGN DS     1           CURRENT GROUP NUMBER
041.050                 758X AIO.CSI DS     1           CURRENT SECTOR INDEX
041.051                 759X AIO.LGN DS     1           LAST GROUP NUMBER
041.052                 760X AIO.LSI DS     1           LAST SECTOR INDEX
041.053                 761X AIO.DTA DS     2           DEVICE TABLE ADDRESS
041.055                 762X AIO.DES DS     2           DIRECTORY SECTOR
041.057                 763X AIO.DEV DS     2           DEVICE CODE
041.061                 764X AIO.UNI DS     1           UNIT NUMBER (0-9)
                        765X
041.062                 766X AIO.DIR DS     DIRELEN     DIRECTORY ENTRY
                        767X
041.111                 768X AIO.CNT DS     1           SECTOR COUNT
041.112                 769X AIO.EOM DS     1           END OF MEDIA FLAG
041.113                 770X AIO.EOF DS     1           END OF FILE FLAG
041.114                 771X AIO.TFP DS     2           TEMP FILE POINTERS
041.116                 772X AIO.CHA DS     2           ADDRESS OF CHANNEL BLOCK (IOC,DDA)



041.120                 774X S.SCR   DS     2           SYSTEM SCRATCH AREA ADDRESS
041.122                 775          XTEXT  DEVDEF



                        777X **     DEVICE TABLE ENTRYS.
                        778X
000.000                 779X        ORG    0
                        780X
000.000                 781X DEV.NAM DS     2           DEVICE NAME
000.000                 782X DV.EL   EQU    00000000B   END OF DEVICE LIST FLAG
000.001                 783X DV.NU   EQU    00000001B   DEVICE ENTRY NOT IN USE
                        784X
000.002                 785X DEV.RES DS     1           DRIVER RESIDENSE CODE
000.001                 786X DR.IM   EQU    00000001B   DRIVER IN MEMORY
000.002                 787X DR.PR   EQU    00000010B   DRIVER PERMINANTLY RESIDENT
                        788X
000.003                 789X DEV.JMP DS     1           JMP TO PROCESSOR
000.004                 790X DEV.DDA DS     2           DRIVER ADDRESS
000.006                 791X DEV.FLG DS     1           FLAG BYTE
000.001                 792X DT.DD   EQU    00000001B   DIRECTORY DEVICE
000.002                 793X DT.CR   EQU    00000010B   CAPABLE OF READ OPERATION
000.004                 794X DT.CW   EQU    00000100B   CAPABLE OF WRITE OPERATION
                        795X
000.007                 796X DEV.SPG DS     1           SECTORS PER GROUP THIS DEVICE
000.010                 797X DEV.MUM DS     1           MOUNTED UNIT MASK
000.011                 798X DEV.MNU DS     1           MAXIMUM NUMBER OF UNITS
```

```
000.012                 799X DEV.UNT DS      2              ADDRESS OF UNIT SPECIFIC DATA TABLE
                        800X
000.014                 801X DEV.DVL DS      2              DRIVER BYTE LENGTH
000.016                 802X DEV.DVG DS      1              DRIVER ROUTINE GROUP ADDRESS
                        803X
000.017                 804X DEVELEN EQU     *              DEVICE TABLE ENTRY LENGTH



                        806X **      UNIT SPECIFIC DEVICE DATA TABLE ENTRIES
                        807X
000.000                 808X         ORG     0
                        809X
000.000                 810X UNT.FLG DS      1              UNIT SPECIFIC *DEV.FLG*
000.001                 811X UNT.GRT DS      2              ADDRESS OF GROUP RESERVATION TABLE (IF DT.DD)
000.003                 812X UNT.GTS DS      2              GRT SECTOR NUMBER
000.005                 813X UNT.DIS DS      2              DIRECTORY FIRST SECTOR NUMBER
                        814X
000.007                 815X UNT.SIZ EQU     *              SIZE OF UNIT SPECIFIC DATA TABLE PER UNIT
000.007                 816          XTEXT   ABSDEF



                        818X **      ABS FORMAT EQUIVALENCES.
                        819X
000.000                 820X         ORG     0
                        821X
000.000                 822X ABS.ID  DS      1              3770 = BINARY FILE FLAG
000.001                 823X         DS      1              FILE TYPE (FT.ABS)
000.002                 824X ABS.LDA DS      2              LOAD ADDRESS
000.004                 825X ABS.LEN DS      2              LENGTH OF ENTIRE RECORD
000.006                 826X ABS.ENT DS      2              ENTRY POINT
                        827X
000.010                 828X ABS.COD DS      0              CODE STARTS HERE
000.010                 829          XTEXT   MTRDEF



                        831X **      HDOS MONITOR PRIVATE RAM AREA DEFINITIONS.
                        832X
000.000                 833X         ORG     0
000.000                 834X M.SYSM  DS      1              SYSCALL ITTERATION COUNT
000.001                 835X M.SALO  DS      1              STAND-ALONE FLAG
000.002                 836X M.CSLC  DS      1              LINES IN CONSOLE BUFFER
000.003                 837X M.CPRE  DS      1              CONSOLE PREVIOUS CHARACTER
000.004                 838X M.CRUB  DS      1              CONSOLE RUBOUT FLAG
000.005                 839X M.CINT  DS      1              CONSOLE INTERRUPT FLAG
000.006                 840X M.CIN   DS      2              CONSOLE CB IN POINTER
000.010                 841X M.COUT  DS      2              CONSOLE CB OUT POINTER
000.012                 842X M.CFWA  DS      2              CONSOLE CB FWA POINTER
000.014                 843X M.CLWA  DS      2              CONSOLE CB LWA POINTER
000.016                 844X M.CDLY  DS      1              CONSOLE PAD CHARACTER COUNT
000.017                 845X M.CDCA  DS      2              ADDRESS OF CHARACTER BEING PADDED
```

```
                                847  **      CODE HEADERS FOR ABSOLUTE MODE
                                848
     042.170                    849          ORG     USERFWA-ABS.COD
     042.170  377 000           850          DB      3770,FT.ABS
     042.172  200 042           851          DW      USERFWA
     042.174  366 010           852          DW      MEML-USERFWA
     042.176  200 042           853          DW      USERFWA          ENTRY
```

```
                                  856  ***     SYSXIT - SYSTEM EXIT PROCESSOR.
                                  857  *
                                  858
                                  859
        042.200                   860  SYSXIT  EQU     *
                                  861
        042.200  062 327 042     862          STA     SYSXITA         SAVE VALUE
        042.203  377 011         863          DB      SYSCALL,.VERS
        042.205  332 312 042     864          JC      SYSX1           ERROR IN GETTING VERSION,
        042.210  376 026         865          CPI     VERS                    PROBABLY NO .VERS CALL
        042.212  302 312 042     866          JNZ     SYSX1           NOT THE CORRECT HDOS VERSION FOR THIS SYSCMD
        042.215  041 347 055     867          LXI     H,RMEML
        042.220  377 052         868          DB      SYSCALL,.SETTP
        042.222  332 222 042     869          JC      *               IF ERROR
        042.225  061 200 042     870          LXI     SP,STACK
        042.230  072 327 042     871          LDA     SYSXITA
        042.233  247             872          ANA     A
        042.234  304 041 043     873          CNZ     PRSCL           PRESET CONSOLE IF SET
        042.237  076 377         874          MVI     A,-1
        042.241  377 055         875          DB      SYSCALL,.CLEAR  CLEAR OVERLAY CHANNEL
        042.243  315 330 042     876          CALL    CCT             CLEAR CHANNEL TABLE
        042.246  072 032 041     877          LDA     S.MOUNT
        042.251  247             878          ANA     A
        042.252  312 213 043     879          JZ      SYSCMD          HDOS IS NOT MOUNTED
        042.255  315 333 042     880          CALL    CDT             CLEAR DEVICE TABLE
        042.260  315 022 043     881          CALL    COT             CLEAR OVERLAY TABLE
        042.263  052 320 040     882          LHLD    S.SYSM
        042.266  353             883          XCHG                    (DE) = FWA RESERVED SYSTEM AREA
        042.267  052 356 040     884          LHLD    S.RFWA          (HL) = FWA RESIDENT SYSTEM
        042.272  315 216 030     885          CALL    $CMEHL
        042.275  312 213 043     886          JE      SYSCMD          ALREADY EMPTY
        042.300  257             887          XRA     A
        042.301  062 371 040     888          STA     S.OVLFL         CLEAR S.OVLFL
        042.304  042 320 040     889          SHLD    S.SYSM          SET MEMORY USAGE
        042.307  303 213 043     890          JMP     SYSCMD          SYSTEM COMMAND
                                  891
                                  892  *       HDOS VERSION NOT CORRECT, OR ERROR UPON RETURN
                                  893
        042.312  076 050         894  SYSX1   MVI     A,EC.NCV        NOT CORRECT VERSION
        042.314  067             895          STC                     SET ERROR FLAG
        042.315  046 012         896          MVI     H,NL
        042.317  377 057         897          DB      SYSCALL,.ERROR
        042.321  315 013 041     898          CALL    S.FASER         FATAL SYSTEM ERROR
        042.324  303 000 030     899          JMP     ROMBOOT         SHOULD NEVER GET HERE!
                                  900
        042.327  000             901  SYSXITA DB      0               PSW VALUE
```

```
                                905  **      CCT - CLEAR CHANNEL TABLE.
                                906  *
                                907  *       CCT CLEARS OUT THE CHANNEL TABLE.
                                908  *
                                909  *       ENTRY   NONE
                                910  *       EXIT    NONE
                                911  *       USES    ALL
                                912
                                913
042,330  377 056               914  CCT     DB      SYSCALL+.CLEARA
042,332  311                   915          RET


                                917  **      CDT - CLEAR DEVICE TABLE.
                                918  *
                                919  *       CDT CLEARS THE DEVICE TABLE.
                                920  *
                                921  *       NON-RESIDENT DEVICE DRIVERS ARE DISCARDED.
                                922  *       DIRECTORY DEVICES ARE ABORTED.
                                923  *
                                924  *       ENTRY   NONE
                                925  *       EXIT    NONE
                                926  *       USES    ALL
                                927
                                928
042,333  052 354 040           929  CDT     LHLD    S.DFWA          (HL) = DEVICE TABLE FWA
042,336  176                   930  CDT1    MOV     A,M
042,337  247                   931          ANA     A
042,340  310                   932          RZ                      END OF TABLE
042,341  345                   933          PUSH    H               SAVE ADDRESS
                               934
                               935  *       HAVE ENTRY
                               936
042,342  043                   937          INX     H
042,343  043                   938          INX     H
000,000                        939          ERRNZ   DEV.RES-2
042,344  176                   940          MOV     A,M             (A) = DEV.RES
042,345  346 002               941          ANI     DR.PR
042,347  302 370 042           942          JNZ     CDT2            PERMINANTLY RESIDENT
042,352  176                   943          MOV     A,M
042,353  346 376               944          ANI     377Q-DR.IM
042,355  167                   945          MOV     M,A             CLEAR IN MEMORY
042,356  043                   946          INX     H
042,357  043                   947          INX     H
000,000                        948          ERRNZ   DEV.DDA-DEV.RES-2       (HL) = #DEV.DDA
042,360  066 010               949          MVI     M,#S.SDD
042,362  043                   950          INX     H
042,363  066 041               951          MVI     M,S.SDD/256     SET SDD ADDRESS
042,365  053                   952          DCX     H
042,366  053                   953          DCX     H
042,367  053                   954          DCX     H
000,000                        955          ERRNZ   DEV.RES-DEV.DDA+2       (HL) = #DEV.RES
042,370  043                   956  CDT2    INX     H
042,371  043                   957          INX     H
```

```
000.000                  958          ERRNZ   DEV.DDA-DEV.RES-2       (HL) = *DEV.DDA
042.372  136            959          MOV     E,M
042.373  043            960          INX     H
042.374  126            961          MOV     D,M                     (DE) = (DEV.DDA)
042.375  043            962          INX     H
000.000                  963          ERRNZ   DEV.FLG-DEV.DDA-2
042.376  176            964          MOV     A,M
000.000                  965          ERRNZ   DT.DD-1
042.377  037            966          RAR
043.000  322 011 043     967          JNC     CDT3                    NOT DIRECTORY DEVICE
                         968
                         969  *       IS DIRECTORY DEVICE, ABORT IT
                         970
043.003  353            971          XCHG
043.004  076 007         972          MVI     A,DC.ABT
043.006  315 021 043     973          CALL    PCHL                    ABORT IT
                         974
                         975  *       ADVANCE TO NEXT ENTRY
                         976
043.011  341            977  CDT3    POP     H                       (HL) = ENTRY FWA
043.012  021 017 000     978          LXI     D,DEVELEN
043.015  031            979          DAD     D
043.016  303 336 042     980          JMP     CDT1                    DO SOME MORE
                         981
043.021  351            982  PCHL    PCHL


                         984  **      COT     - CLEAR OVERLAY TABLE
                         985  *
                         986  *       COT CLEARS THE OVERLAY TABLE
                         987  *
                         988  *       ENTRY:  NONE
                         989  *       EXIT:   NONE
                         990  *       USES:   ALL
                         991  *
                         992
043.022  052 350 040     993  COT     LHLD    S.OFWA                  (HL) = FIRST WORD OF OVERLAY TABLE
043.025  257            994          XRA     A
043.026  001 006 000     995          LXI     B,OVL.FLB
043.031  011            996          DAD     B
043.032  167            997          MOV     M,A                     ZERO FLAG BYTE FOR *HDOSOVL0.SYS*
043.033  001 010 000     998          LXI     B,OVL.ENS
043.036  011            999          DAD     B
043.037  167           1000          MOV     M,A                     ZERO FLAG BYTE FOR *HDOSOVL1.SYS*
043.040  311           1001          RET
```

```
                                1003  **      PRSCL - PRESET CONSOLE.
                                1004  *
                                1005  *       PRSCL PRESETS THE CONSOLE UART, SETS THE DEFAULT CONTROL PARAMETERS,
                                1006  *       AND CLEARS THE TYPE-AHEAD BUFFER.
                                1007  *
                                1008  *       ENTRY   NONE
                                1009  *       EXIT    NONE
                                1010  *       USES    ALL
                                1011
                                1012
        043.041                 1013  PRSCL   EQU     *
        043.041  363            1014          DI                      DISABLE INTERRUPTS WHILE FIXING
        043.042  052 346 040    1015          LHLD    S.DLINK
        043.045  043            1016          INX     H
        043.046  043            1017          INX     H
        000.000                 1018          ERRNZ   M.CSLC-2
        043.047  066 000        1019          MVI     M,0             CLEAR LINE COUNT
        043.051  043            1020          INX     H
        000.000                 1021          ERRNZ   M.CPRE-M.CSLC-1
        043.052  066 000        1022          MVI     M,0             CLEAR PREVIOUS CHARACTER
        043.054  043            1023          INX     H
        000.000                 1024          ERRNZ   M.CRUB-M.CPRE-1
        043.055  066 000        1025          MVI     M,0             CLEAR RUBOUT FLAG
        043.057  043            1026          INX     H
        000.000                 1027          ERRNZ   M.CINT-M.CRUB-1
        043.060  066 000        1028          MVI     M,0             CLEAR INTERRUPT FLAGS
        043.062  043            1029          INX     H
        000.000                 1030          ERRNZ   M.CIN-M.CINT-1
        043.063  345            1031          PUSH    H               SAVE ADDRESS OF M.CIN
        043.064  043            1032          INX     H
        043.065  043            1033          INX     H
        043.066  043            1034          INX     H
        043.067  043            1035          INX     H
        000.000                 1036          ERRNZ   M.CFWA-M.CINT-5
        043.070  136            1037          MOV     E,M
        043.071  043            1038          INX     H
        043.072  126            1039          MOV     D,M             (DE) = BUFFER FWA
        043.073  341            1040          POP     H               (HL) = @M.CIN
        043.074  163            1041          MOV     M,E
        043.075  043            1042          INX     H
        043.076  162            1043          MOV     M,D
        043.077  043            1044          INX     H
        000.000                 1045          ERRNZ   M.COUT-M.CIN-2
        043.100  163            1046          MOV     M,E
        043.101  043            1047          INX     H
        043.102  162            1048          MOV     M,D
        043.103  373            1049          EI                      ALLOW INTERRUPTS NOW
        043.104  315 313 052    1050          CALL    SCU
        043.107  315 310 051    1051          CALL    ECI
        043.112  311            1052          RET
```

```
                        1055  ***     SYSCMD - SYSTEM COMMAND PROCESSOR.
                        1056  *
                        1057
                        1058
043.113  365           1059  ERROR   PUSH    PSW             SAVE ERROR CODE
043.114  377 007       1060          DB      SYSCALL,.CLRCO  CLEAR CONSOLE BUFFER AND CTL-O
043.116  361           1061          POP     PSW             (A) = ERROR CODE
043.117  046 007       1062          MVI     H,BELL          ENTER HERE IF ERROR FROM SYSTEM
043.121  377 057       1063          DB      SYSCALL,.ERROR
043.123  257           1064          XRA     A               (A) = 0
043.124  303 200 042   1065          JMP     SYSXIT          MASTER CLEAR SYSTEM
                        1066
043.127  315 136 031   1067  ILLSYN  CALL    $TYPTX
043.132  012 007 111   1068          DB      NL,BELL,'Illegal Command Syntax',' '+200Q
043.163  303 213 043   1069          JMP     SYSCMD
                        1070
043.166  315 136 031   1071  ILLCMD  CALL    $TYPTX
043.171  012 007 111   1072          DB      NL,BELL,'Illegal Command',' '+200Q
                        1073
043.213  041 302 044   1074  SYSCMD  LXI     H,CCHIT
043.216  076 003       1075          MVI     A,CTLC
043.220  377 041       1076          DB      SYSCALL,.CTLC   SETUP CTL-C PROCESSOR
043.222  061 200 042   1077          LXI     SP,STACK
043.225  257           1078          XRA     A
043.226  062 326 040   1079          STA     S.CSLMD         CLEAR SPECIAL CONSOLE MODES
043.231  315 340 050   1080          CALL    $CCO            CLEAR CTL-O
043.234  315 035 052   1081          CALL    $GNL            GUARANTEE NEW LINE
000.001               1082          IF      .MANUF.
                        1083
                        1084  *       LINK TO MANDIAG.ABS UNLESS FLAGED
                        1085
                        1086          LHLD    40076A          SEE IF 'GL'
                        1087          LXI     D,'LG'
                        1088          CALL    $CDEHL
                        1089          JE      MANU1           RUN AS NORMAL
                        1090          LXI     H,MANUA
                        1091          DB      SYSCALL,.LINK
                        1092          JMP     ERROR
                        1093
                        1094  MANUA   DB      'SY0:MANDIAG.ABS',0
                        1095
                        1096  MANU1   EQU     *
                        1097          ENDIF
043.237  072 032 041   1098          LDA     S.MOUNT
043.242  247           1099          ANA     A
043.243  302 262 043   1100          JNZ     SYSCO           SYSTEM IS MOUNTED
043.246  021 001 000   1101          LXI     D,M.SALO
043.251  052 346 040   1102          LHLD    S.DLINK
043.254  031           1103          DAD     D
043.255  176           1104          MOV     A,M
043.256  247           1105          ANA     A
043.257  312 213 047   1106          JZ      BYE             NO SYSTEM, AND NO STAND-ALONE FLAG SET
                        1107
043.262  315 136 031   1108  SYSCO   CALL    $TYPTX
043.265  276           1109          DB      '>'+200Q                PROMPT
043.266  041 157 054   1110          LXI     H,LINE
```

```
043.271  315 246 052  1111          CALL    $RTL.           READ COMMAND LINE (UPPER CASE)
043.274  332 213 043  1112          JC      SYSCMD          CTL-D STRUCK
                      1113
                      1114  *       CRACK COMMAND NAME
                      1115
043.277  076 200      1116          MVI     A,200Q
043.301  062 366 053  1117          STA     VERB-1          REQUIRED BY VERB SCANNING
043.304  021 157 054  1118          LXI     D,LINE
043.307  041 367 053  1119          LXI     H,VERB
043.312  032          1120  SYSC1   LDAX    D
043.313  376 056      1121          CPI     '.'
043.315  312 344 043  1122          JZ      SYSC2           VALID FILE SPECIFICATION CHARACTER
043.320  376 060      1123          CPI     '0'
043.322  332 352 043  1124          JC      SYSC3           < '0' AND NOT '.'
043.325  376 073      1125          CPI     ';'+1
043.327  332 344 043  1126          JC      SYSC2           NUMERIC, OR ':'
043.332  376 101      1127          CPI     'A'
043.334  332 352 043  1128          JC      SYSC3           NOT ALPHA, NOT NUMERIC, NOT ':', NOT '.'
043.337  376 133      1129          CPI     'Z'+1
043.341  322 352 043  1130          JNC     SYSC3           NOT ALPHA
                      1131
                      1132  *       HAVE ALPHA CHARACTER, BUILD INTO COMMAND VERB
                      1133
043.344  167          1134  SYSC2   MOV     M,A
043.345  043          1135          INX     H
043.346  023          1136          INX     D
043.347  303 312 043  1137          JMP     SYSC1
                      1138
                      1139  *       HAVE SPLIT OUT THE VERB. SEE IF WE KNOW IT
                      1140
043.352  325          1141  SYSC3   PUSH    D               SAVE LINE POINTER
043.353  066 000      1142          MVI     M,0             FORCE END OF VERB
043.355  053          1143          DCX     H
043.356  176          1144          MOV     A,M
043.357  356 200      1145          XRI     200Q            SET END OF WORD
043.361  362 127 043  1146          JP      ILLSYN          NULL COMMAND
043.364  167          1147          MOV     M,A
                      1148
                      1149  *       SEE IF WE KNOW THIS COMMAND
                      1150
043.365  021 367 053  1151          LXI     D,VERB
043.370  041 072 044  1152          LXI     H,SYSCC
043.373  315 332 051  1153          CALL    $FST
043.376  302 045 044  1154          JNZ     SYSC5           NOT BUILD-IN COMMAND
044.001  176          1155          MOV     A,M             (A) = INDEX
044.002  315 061 031  1156          CALL    $TJMP           ENTER PROCESSOR
                      1157
```

```
044.005                 1159  SYSCA   DS      0
                        1160
000.000                 1161  I.RUN   EQU     *-SYSCA/2
044.005  314 044        1162          DW      RUN
                        1163
000.001                 1164  I.SYS   EQU     *-SYSCA/2
044.007  213 043        1165          DW      SYSCMD          UNUSED
                        1166
000.002                 1167  I.DMO   EQU     *-SYSCA/2
044.011  233 045        1168          DW      DMOUNT
                        1169
000.003                 1170  I.HEL   EQU     *-SYSCA/2
044.013  355 044        1171          DW      HELP
                        1172
000.004                 1173  I.LIS   EQU     *-SYSCA/2
044.015  022 045        1174          DW      LIST
                        1175
000.005                 1176  I.DEL   EQU     *-SYSCA/2
044.017  067 045        1177          DW      DELETE
                        1178
000.006                 1179  I.REN   EQU     *-SYSCA/2
044.021  121 045        1180          DW      RENAME
                        1181
000.007                 1182  I.MOU   EQU     *-SYSCA/2
044.023  207 045        1183          DW      MOUNT
                        1184
000.010                 1185  I.DAT   EQU     *-SYSCA/2
044.025  040 046        1186          DW      DATE
                        1187
000.011                 1188  I.DIR   EQU     *-SYSCA/2
044.027  157 045        1189          DW      DIR
                        1190
000.012                 1191  I.STA   EQU     *-SYSCA/2
044.031  145 046        1192          DW      STATUS
                        1193
000.013                 1194  I.COP   EQU     *-SYSCA/2
044.033  153 045        1195          DW      COPY
                        1196
000.014                 1197  I.BYE   EQU     *-SYSCA/2
044.035  213 047        1198          DW      BYE
                        1199
000.015                 1200  I.RES   EQU     *-SYSCA/2
044.037  271 045        1201          DW      RESET
                        1202
000.016                 1203  I.VER   EQU     *-SYSCA/2
044.041  322 045        1204          DW      VERSN
                        1205
000.017                 1206  I.LOA   EQU     *-SYSCA/2
044.043  006 046        1207          DW      LOADD
                        1208
000.001                 1209          IF      DEBUG
                        1210
                        1211  I.ROM   EQU     *-SYSCA/2
                        1212          DW      ROMBOOT         REBOOT
                        1213
                        1214  I.TRA   EQU     *-SYSCA/2
```

```
                        1215            DW      TRAP            TRAP TO HBUG
                        1216
                        1217  I.HBU     EQU     *-SYSCA/2
                        1218            DW      HBUG            LOAD HBUG
                        1219
                        1220  I.BUG     EQU     *-SYSCA/2
                        1221            DW      BUG             RUN WITH DEBUG
                        1222
                        1223            ENDIF
                        1224
                        1225  *         CANT FIND COMMAND ON THE MAGIC (BUILT-IN) LIST
                        1226  *         TRY TO LINK TO IT
                        1227
044.045  301            1228  SYSC5     POP     B               (BC) = START OF PARAMETERS
044.046  315 343 047     1229            CALL    PCL             PASS COMMAND LINE ON  STACK
044.051  041 157 054     1230            LXI     H,LINE
044.054  021 064 044     1231            LXI     D,SYSCB
044.057  377 040         1232            DB      SYSCALL,.LINK   LINK TO IT
044.061  303 166 043     1233            JMP     ILLCMD          JUST DONT KNOW THIS GUY
                        1234
044.064  123 131 060     1235  SYSCB     DB      'SY0ABS'        DEFAULT FOR LINK
                        1236
                        1237  **        COMMAND TABLE
                        1238  *
                        1239  *         DATA VALUES ARE INDEXES INTO SYSCA
044.005                 1240  .         SET     SYSCA           REFERENCE SYSCA
                        1241
044.072                 1242  SYSCC     DS      0
044.072  252 044        1243            DW      SYSCC+SYSCCL    TABLE LIMIT
044.074  001            1244            DB      1               DATA BYTES PER ENTRY
                        1245
044.075  122 125 316    1246            DB      'RU','N'+200Q,I.RUN
044.101  104 111 123    1247            DB      'DISMOUN','T'+200Q,I.DMO
044.112  110 105 114    1248            DB      'HEL','P'+200Q,I.HEL
044.117  114 111 123    1249            DB      'LIS','T'+200Q,I.LIS
044.124  124 131 120    1250            DB      'TYP','E'+200Q,I.LIS
044.131  104 105 114    1251            DB      'DELET','E'+200Q,I.DEL
044.140  122 105 116    1252            DB      'RENAM','E'+200Q,I.REN
044.147  115 117 125    1253            DB      'MOUN','T'+200Q,I.MOU
044.155  104 101 124    1254            DB      'DAT','E'+200Q,I.DAT
044.162  104 111 322    1255            DB      'DI','R'+200Q,I.DIR
044.166  103 101 324    1256            DB      'CA','T'+200Q,I.DIR
044.172  111 116 304    1257            DB      'IN','D'+200Q,I.DIR
044.176  111 116 104    1258            DB      'INDE','X'+200Q,I.DIR
044.204  123 124 101    1259            DB      'STATU','S'+200Q,I.STA
044.213  123 124 101    1260            DB      'STA','T'+200Q,I.STA
044.220  103 117 120    1261            DB      'COP','Y'+200Q,I.COP
044.225  102 131 305    1262            DB      'BY','E'+200Q,I.BYE
044.231  122 105 123    1263            DB      'RESE','T'+200Q,I.RES
044.237  126 105 322    1264            DB      'VE','R'+200Q,I.VER
044.243  114 117 101    1265            DB      'LOA','D'+200Q,I.LOA
000.001                 1266            IF      DEBUG
                        1267            DB      'REBOO','T'+200Q,I.ROM
                        1268            DB      'TRA','P'+200Q,I.TRA
                        1269            DB      'HBU','G'+200Q,I.HBU
                        1270            DB      'BU','G'+200Q,I.BUG
```

```
                          1271          ENDIF
044.250  000 000         1272          DB      0,0             END OF TABLE
000.160                  1273  SYSCCL  EQU     $-SYSCC         END OF TABLE
044.252                  1274          DS      24              TABLE EXTENSION PATCH AREA


                         1276  **      CCHIT - CTL-C PROCESSOR.
                         1277  *
                         1278  *       ENTER COMMAND LOOP
                         1279
                         1280
044.302  377 007         1281  CCHIT   DB      SYSCALL,.CLRCO          CLEAR CONSOLE BUFFER
044.304  315 136 031     1282          CALL    $TYPTX
044.307  136 303         1283          DB      '^','C'+2000
044.311  303 213 043     1284          JMP     SYSCMD
```

```
                        1288  ***     RUN - PROCESS RUN COMMAND.
                        1289  *
                        1290  *       RUN FNAME [PARAMETER LIST]
                        1291
                        1292
044,314  341            1293  RUN     POP     H               (HL) = COMMAND LINE ADDRESS
044,315  315 050 053     1294          CALL    $SOB            SKIP LEADING BLANKS
044,320  353            1295          XCHG                    (DE) = PROGRAM NAME ADDRESS
044,321  041 157 054     1296          LXI     H,LINE          COPY BACK OVER SELF AND 'RUN '
044,324  315 355 050     1297          CALL    $CPF            COPY FILE NAME SEPERATE
044,327  102             1298          MOV     B,D
044,330  113             1299          MOV     C,E             (BC) = ARGUMENT LIST
044,331  315 343 047     1300          CALL    PCL             PASS COMMAND LINE
044,334  021 347 044     1301          LXI     D,RUNA          (DE) = DEFAULTS ADDRESS
044,337  041 157 054     1302          LXI     H,LINE          (HL) = PROGRAM NAME
044,342  377 040         1303          DB      SYSCALL,,LINK
044,344  303 113 043     1304          JMP     ERROR           DIDNT MAKE IT
                        1305
044,347  123 131 060     1306  RUNA    DB      'SY0ABS'        DEFAULTS
```

```
                                1310  ***     HELP - TYPE HELP FILE.
                                1311  *
                                1312  *       HELP
                                1313  *
                                1314  *       TYPES THE FILE SYSHELP.DOC
                                1315
                                1316
044.355                         1317  HELP    EQU     *
044.355  315 212 052           1318          CALL    $MOVEL
044.360  025 000 375           1319          DW      HELPAL,HELPA,LINE       SETUP COMMAND LINE
044.366  041 164 054           1320          LXI     H,LINE+5                POINT TO PARAMETER LIST
044.371  343                   1321          XTHL                            SUBSTITUTE FOR OLD LIST
044.372  303 022 045           1322          JMP     LIST                    DO AS IN LIST
                                1323
044.375  114 111 123           1324  HELPA   DB      'LIST SY0:SYSHELP.DOC',0
000.025                         1325  HELPAL  EQU     *-HELPA
```

```
                                1329  ***     LST - LIST FILE CONTENTS TO CONSOLE.
                                1330  *
                                1331  *       LIST FNAME
                                1332
                                1333
        045.022                 1334  LIST    EQU     *
        045.022  315 212 052   1335          CALL    $MOVEL
        045.025  004 000 056   1336          DW      LISTAL,LISTA,LINE               SETUP PIP COMMANDS
        045.033  301           1337          POP     B                       DISCARD OLD PARAMETERS
        045.034  315 313 047   1338          CALL    FEC                     FIND END OF COMMAND LINE
        045.037  001 005 000   1339          LXI     B,LISTBL
        045.042  021 062 045   1340          LXI     D,LISTB
        045.045  315 252 030   1341          CALL    $MOVE                   ADD /SUP
        045.050  001 157 054   1342          LXI     B,LINE
        045.053  303 002 050   1343          JMP     PIP                     EXECUTE PIP
                               1344
        045.056  124 124 072   1345  LISTA   DB      'TT:='
        000.004                1346  LISTAL  EQU     *-LISTA
        045.062  057 123 125   1347  LISTB   DB      '/SUP',0
        000.005                1348  LISTBL  EQU     *-LISTB
```

```
                                1352  ***     DELETE - DELETE FILES
                                1353  *
                                1354  *       DELETE FNAME [;FNAME;...;FNAME]
                                1355
                                1356
045.067                         1357  DELETE  EQU     *
045.067  301                    1358          POP     B
045.070  305                    1359          PUSH    B               SAVE COMMAND ADDRESS
045.071  315 313 047            1360          CALL    FEC             FIND END OF COMMAND LINE
045.074  001 010 000            1361          LXI     B,DELAL
045.077  021 111 045            1362          LXI     D,DELA
045.102  315 252 030            1363          CALL    $MOVE           ADD /DEL COMMAND
045.105  301                    1364          POP     B
045.106  303 002 050            1365          JMP     PIP
                                1366
045.111  057 104 105            1367  DELA    DB      '/DELETE',0
000.010                         1368  DELAL   EQU     *-DELA
```

```
                          1372  ***     RENAME - RENAME FILES.
                          1373  *
                          1374  *       RENAME FILE1=FILE2
                          1375
                          1376
045.121                   1377  RENAME  EQU     *
045.121  301             1378          POP     B
045.122  305             1379          PUSH    B               (BC) = START OF COMMAND
045.123  315 313 047      1380          CALL    FEC             FIND END OF COMMAND
045.126  001 010 000      1381          LXI     B,RENAL
045.131  021 143 045      1382          LXI     D,RENA
045.134  315 252 030      1383          CALL    $MOVE           MOVE IN /REN
045.137  301             1384          POP     B
045.140  303 002 050      1385          JMP     PIP             LINK TO PIP
                          1386
045.143  057 122 105      1387  RENA    DB      '/RENAME',0
000.010                   1388  RENAL   EQU     $-RENA
```

```
                        1392  ***     COPY - COPY FILES.
                        1393  *
                        1394  *       COPY TARG=SOURCE
                        1395
                        1396
045.153                 1397  COPY    EQU     *
045.153  301           1398          POP     B               (BC) = ARG ADDRESS
045.154  303 002 050    1399          JMP     PIP             CALL PIP
```

```
                                1403  ***     DIR - DIRECTORY LIST FOR DEVICE
                                1404  *
                                1405  *       DIR  [DEV:] [NAMES]
                                1406
                                1407
      045.157                   1408  DIR     EQU     *
      045.157  301              1409          POP     B
      045.160  305              1410          PUSH    B
      045.161  315 313 047      1411          CALL    FEC             FIND END OF COMMAND LINE
      045.164  001 006 000      1412          LXI     B,DIRAL
      045.167  021 201 045      1413          LXI     D,DIRA
      045.172  315 252 030      1414          CALL    $MOVE
      045.175  301              1415          POP     B
      045.176  303 002 050      1416          JMP     PIP
                                1417
      045.201  057 114 111      1418  DIRA    DB      '/LIST',0
      000.006                   1419  DIRAL   EQU     *-DIRA
```

```
                                1423  ***     MOUNT - MOUNT DISK.
                                1424  *
                                1425  *       MOUNT DEV:
                                1426
                                1427
045.207                         1428  MOUNT   EQU     *
045.207  341                    1429          POP     H               (HL) = DEVICE NAME ADDRESS
045.210  377 200                1430          DB      SYSCALL,.MOUNT
045.212  332 113 043            1431          JC      ERROR
045.215  303 213 043            1432          JMP     SYSCMD
                                1433
045.220  115 157 165            1434  MOUNTA  DB      'Mounted On',' '+200Q


                                1436  ***     DISMOUNT - DISMOUNT DEV:
                                1437  *
                                1438  *       DISMOUNT DEV:
                                1439
                                1440
045.233                         1441  DMOUNT  EQU     *
045.233  341                    1442          POP     H               (HL) = LINE ADDRESS
045.234  345                    1443          PUSH    H               SAVE IN CASE OF ERROR
045.235  377 201                1444          DB      SYSCALL,.DMOUN
045.237  341                    1445          POP     H
045.240  322 213 043            1446          JNC     SYSCMD
045.243  376 044                1447          CPI     EC.NFM
045.245  312 254 045            1448          JZ      DMO1            NO PROVISION MADE FOR HDOS TO RESIDE,NOT FATAL
045.250  067                    1449          STC
045.251  303 113 043            1450          JMP     ERROR           RESET ERROR FLAG CLEARED BY CPI?
                                1451
045.254  345                    1452  DMO1    PUSH    H               SAVE LINE ADDRESS
045.255  315 324 047            1453          CALL    LOADOV          LOAD OVERLAYS
045.260  341                    1454          POP     H               RESTORE LINE ADDRESS
045.261  377 201                1455          DB      SYSCALL,.DMOUN
045.263  332 113 043            1456          JC      ERROR
045.266  303 213 043            1457          JMP     SYSCMD
```

```
                                1461  ***     RESET   - PROCESS RESET COMMAND
                                1462  *
                                1463  *       IF THE  *SALONE*  FLAG IS NOT SET, THIS COMMAND IS CONSIDERED ILLEGAL,
                                1464  *       WHICH IMPLIES THAT A DIRECTORY SEARCH SHOULD BE DONE.
                                1465  *
                                1466  *       RESET DEV:              RESET DEV:
                                1467  *
                                1468
045.271                         1469  RESET   EQU     *
045.271  315 275 047            1470          CALL    CSA             CHECK STAND ALONE
045.274  312 045 044            1471          JZ      SYSC5           STAND-ALONE NOT SET  =>  COMMAND ILLEGAL,
                                1472  *                                TRY LOOK-UP
045.277  315 324 047            1473          CALL    LOADOV          LOAD BOTH OVERLAYS
045.302  341                    1474          POP     H               (HL) = LINE ADDRESS
045.303  315 050 053            1475          CALL    $SOB            (A) = NEXT CHARACTER
045.306  247                    1476          ANA     A
045.307  312 127 043            1477          JZ      ILLSYN          MUST HAVE AN EXPLICIT DEVICE SPECIFICATION
045.312  377 204                1478          DB      SYSCALL,.RESET
045.314  332 113 043            1479          JC      ERROR
045.317  303 213 043            1480          JMP     SYSCMD


                                1483  ***     VERSN   - VERSION
                                1484  *
                                1485  *       VER                     PRINT THE CURRENT VERSION OF HDOS
                                1486  *
                                1487
045.322                         1488  VERSN   EQU     *
045.322  377 011                1489          DB      SYSCALL,.VERS
045.324  322 331 045            1490          JNC     VERS1
045.327  076 020                1491          MVI     A,1*16+0        IF  ERROR ON GETTING VERSION, MUST BE 1.0
045.331  365                    1492  VERS1   PUSH    PSW             SAVE VERSION
045.332  346 360                1493          ANI     11110000B       MAP OUT HIGH ORDER BCD DIGIT
045.334  017                    1494          RRC
045.335  017                    1495          RRC
045.336  017                    1496          RRC
045.337  017                    1497          RRC
045.340  306 060                1498          ADI     '0'
045.342  062 377 045            1499          STA     VERSA
045.345  361                    1500          POP     PSW
045.346  346 017                1501          ANI     00001111B       MAP OUT LOW ORDER BCD DIGIT
045.350  306 060                1502          ADI     '0'
045.352  062 001 046            1503          STA     VERSB
045.355  315 136 031            1504          CALL    $TYPTX
045.360  110 104 117            1505          DB      'HDOS',TAB,'Version:
045.377  000                    1506  VERSA   DB      0
046.000  056                    1507          DB      '.'
046.001  000                    1508  VERSB   DB      0
046.002  212                    1509          DB      ENL
046.003  303 213 043            1510          JMP     SYSCMD
```

```
                                1514  ***     LOADD   -  LOAD DEVICE DRIVER
                                1515  *
                                1516  *       IF THE  *SALONE*  FLAG IS NOT SET, THIS COMMAND IS CONSIDERED ILLEGAL
                                1517  *       WHICH IMPLIES THAT A DIRECTORY SEARCH SHOULD BE DONE.
                                1518  *
                                1519  *       LOAD DEV:
                                1520  *
                                1521
046.006                         1522  LOADD   EQU     *
                                1523  *       CALL    CSA                                             /79.11.GC/
                                1524  *       JZ      SYSCS                                           /79.11.GC/
046.006  341                    1525          POP     H               (HL) = DEVICE SPECIFICATION
046.007  377 062                1526          DB      SYSCALL+.LOADD
046.011  332 113 043            1527          JC      ERROR
046.014  052 320 040            1528          LHLD    S.SYSM
046.017  042 356 040            1529          SHLD    S.RFWA          MAKE IT PART OF THE RESIDENT SYSTEM
046.022  052 053 041            1530          LHLD    AIO.DTA         DEVICE TABLE ADDRESS
046.025  021 002 000            1531          LXI     D,DEV.RES
046.030  031                    1532          DAD     D
046.031  176                    1533          MOV     A,M
046.032  366 002                1534          ORI     DR.PR           FLAG DEVICE AS PERMANENTLY RESIDENT
046.034  167                    1535          MOV     M,A
046.035  303 213 043            1536          JMP     SYSCMD
```

```
                                1540  ***     DATE - PROCESS DATE COMMAND.
                                1541  *
                                1542  *       DATE                    PRINT DATE
                                1543  *       DATE MM-DDD-YY          SET DATE
                                1544
                                1545
046.040                         1546  DATE    EQU     *
046.040  341                   1547          POP     H
046.041  315 050 053           1548          CALL    $SOB
046.044  176                   1549          MOV     A,M
046.045  247                   1550          ANA     A
046.046  312 125 046           1551          JZ      DATE3           HE JUST WANTS TO KNOW THE DATE
                                1552
                                1553  *       SET NEW DATE
                                1554
046.051  315 130 050           1555          CALL    $CAD            CODE AUGUSTAN DATE
046.054  322 112 046           1556          JNC     DATE2           OK
046.057  315 136 031           1557          CALL    $TYPTX
046.062  007 111 154           1558          DB      BELL,'Illegal Date Format',ENL
046.107  303 125 046           1559          JMP     DATE3
                                1560
046.112  353                   1561  DATE2   XCHG
046.113  042 310 040           1562          SHLD    S.DATC
046.116  353                   1563          XCHG
046.117  041 277 040           1564          LXI     H,S.DATE
046.122  315 025 051           1565          CALL    $DAD            DECODE INTO ASCII
                                1566
                                1567  *       DISPLAY THE CURRENT DATE
                                1568
046.125  046 040               1569  DATE3   MVI     H,' '
046.127  257                   1570          XRA     A
046.130  377 057               1571          DB      SYSCALL,,ERROR  PRINT SYSTEM TYPE
046.132  041 277 040           1572          LXI     H,S.DATE
046.135  076 011               1573          MVI     A,9
046.137  315 066 053           1574          CALL    $TYPCC          TYPE DATE
046.142  303 213 043           1575          JMP     SYSCMD          EXIT
```

```
                                1579  ***     STATUS - PRINT SYSTEM STATUS.
                                1580  *
                                1581  *       STATUS
                                1582
                                1583
046.145                         1584  STATUS  EQU     *
046.145  315 017 051           1585          CALL    $CRLF
046.150  257                   1586          XRA     A
046.151  046 011               1587          MVI     H,TAB
046.153  377 057               1588          DB      SYSCALL,.ERROR          SYSTEM BANNER MESSAGE
046.155  041 277 040           1589          LXI     H,S.DATE
046.160  076 011               1590          MVI     A,9
046.162  315 066 053           1591          CALL    $TYPCC          TYPE DATE
046.165  052 273 040           1592          LHLD    D.OPR
046.170  104                   1593          MOV     B,H
046.171  115                   1594          MOV     C,L
046.172  041 350 046           1595          LXI     H,STATB
046.175  076 005               1596          MVI     A,5
046.177  315 102 053           1597          CALL    $UDDN           UNPACK READ COUNT
046.202  052 275 040           1598          LHLD    D.OPW
046.205  104                   1599          MOV     B,H
046.206  115                   1600          MOV     C,L
046.207  041 365 046           1601          LXI     H,STATC
046.212  076 005               1602          MVI     A,5
046.214  315 102 053           1603          CALL    $UDDN           UNPACK WRITE COUNT
046.217  072 261 040           1604          LDA     D.HECNT
046.222  117                   1605          MOV     C,A
046.223  006 000               1606          MVI     B,0
046.225  076 003               1607          MVI     A,3
046.227  041 025 047           1608          LXI     H,STATD
046.232  315 102 053           1609          CALL    $UDDN           UNPACK HARD COUNT
046.235  052 262 040           1610          LHLD    D.SECNT
046.240  174                   1611          MOV     A,H
046.241  247                   1612          ANA     A
046.242  037                   1613          RAR
046.243  107                   1614          MOV     B,A
046.244  175                   1615          MOV     A,L
046.245  037                   1616          RAR
046.246  117                   1617          MOV     C,A
046.247  076 005               1618          MVI     A,5
046.251  041 046 047           1619          LXI     H,STATE
046.254  315 102 053           1620          CALL    $UDDN           UNPACK SOFT COUNT
046.257  041 335 046           1621          LXI     H,STATA
046.262  377 003               1622          DB      SYSCALL,.PRINT
046.264  072 126 040           1623          LDA     D.ERTS
046.267  247                   1624          ANA     A
046.270  312 332 046           1625          JZ      STAT0           NO RECENT ERROR TO REPORT
046.273  137                   1626          MOV     E,A
046.274  026 000               1627          MVI     D,0
046.276  315 324 030           1628          CALL    $MU10           (HL) = TRACK*10
046.301  072 127 040           1629          LDA     D.ERTS+1
046.304  315 101 030           1630          CALL    $DADA.          (HL) = SECTOR NUMBER
046.307  104                   1631          MOV     B,H
046.310  115                   1632          MOV     C,L             (BC) = SECTOR NUMBER
046.311  041 143 047           1633          LXI     H,STATG
046.314  076 003               1634          MVI     A,3
```

```
046.316  315 102 053  1635          CALL    $UDDN           UNPACK NUMBER
046.321  041 077 047  1636          LXI     H,STATF
046.324  377 003      1637          DB      SYSCALL,.PRINT
046.326  257          1638          XRA     A
046.327  062 126 040  1639          STA     D.ERTS
                      1640
                      1641  *       OUTPUT THE SY: DEVICE INFORMATION
                      1642
046.332               1643  STAT0   EQU     *
                      1644
000.001               1645          IF      1
                      1646
                      1647          CALL    $CRLF           OUTPUT AN EXTRA BLANK LINE FOR AESTHETICS
                      1648          LHLD    S.DFWA
                      1649          CALL    $INDLB
                      1650          DW      DEV.MUM
                      1651          MOV     C,A             C  = MOUNTED UNITS MASK
                      1652
                      1653          CALL    $INDLB
                      1654          DW      DEV.MNU
                      1655          DCR     A               A  = MAXIMUM UNIT NUMBER
                      1656
                      1657  STAT1   PUSH    PSW
                      1658          CALL    STAT2,          OUTPUT THE DEVICE INFORMATION
                      1659          CALL    $TYPTX
                      1660          DB      NL,ENL          OUTPUT THE NEWLINES, ETC.
                      1661          POP     PSW
                      1662          DCR     A
                      1663          JP      STAT1           NOT FINISHED
                      1664
                      1665          JMP     SYSCMD
                      1666
                      1667  *       OUTPUT THE INFORMATION FOR ONE UNIT OF SY:
                      1668
                      1669  STAT2,  EQU     *
                      1670          MOV     B,A
                      1671          STA     AIO.UNI         GET READY FOR THE UNIT LATER
                      1672          ADI     '0'
                      1673          STA     STATK           SET UP UNIT NUMBER IN MESSAGE
                      1674
                      1675          XRA     A
                      1676          CALL    BITS
                      1677          ANA     C
                      1678          JNZ     STAT3           DEVICE IS MOUNTED
                      1679
                      1680  *       OUTPUT MESSAGE FOR UNMOUNTED UNIT
                      1681
                      1682          CALL    $TYPTX
                      1683          DB      'No Diskette Mounted On',' '+200Q
                      1684          MVI     A,STATL
                      1685          LXI     H,STATJ
                      1686          CALL    $TYPCC
                      1687          RET
                      1688
                      1689  *       OUTPUT VOLUME NUMBER, AND LABEL FOR MOUNTED UNIT
                      1690
```

```
                                1691  STAT3   EQU     *
                                1692          PUSH    B
                                1693
                                1694  *       READ THE VOLUME LABEL
                                1695
                                1696          MVI     A,DC.RER
                                1697          LXI     B,256
                                1698          LXI     D,LABEL
                                1699          LXI     H,DDF.LAB
                                1700          CALL    SYDD
                                1701          JC      ERROR           BAD TROUBLE
                                1702
                                1703  *       OUTPUT THE MESSAGE STRINGS
                                1704
                                1705          LDA     LABEL+LAB.SER
                                1706          MOV     C,A
                                1707          MVI     B,0
                                1708          LXI     H,STATI
                                1709          MVI     A,3
                                1710          CALL    $UDD
                                1711          LXI     H,STATH
                                1712          DB      SYSCALL,,PRINT  PRINT THE UNIT, AND VOLUME NUMBER
                                1713
                                1714          LXI     H,LABEL+LAB.LAB
                                1715          CALL    $DTB
                                1716          DCR     A
                                1717          CNZ     $TYPCC          PRINT THE LABEL
                                1718          POP     B
                                1719          RET
                                1720
                                1721          ELSE
                                1722
046.332  303 213 043            1723          JMP     SYSCMD
                                1724
                                1725          ENDIF
                                1726
046.335  012 104 151            1727  STATA   DB      NL,'Disk I/O:  '
046.350  116 116 116            1728  STATB   DB      'NNNNN Reads, '
046.365  116 116 116            1729  STATC   DB      'NNNNN Writes Performed'
047.013  012 105 162            1730          DB      NL,'Errors:           '
047.025  116 116 116            1731  STATD   DB      'NNN Hard Errors ('
047.046  116 116 116            1732  STATE   DB      'NNNNN Recovered Errors)',NL,ENL
047.077  114 141 163            1733  STATF   DB      'Last Hard Error Occurred on Sector #'
047.143  116 116 116            1734  STATG   DB      'NNN',NL,ENL
                                1735
047.150  126 157 154            1736  STATH   DB      'Volume '
047.157  170 170 170            1737  STATI   DB      'xxx, Mounted On '
                                1738
047.177  123 131                1739  STATJ   DB      'SY'
047.201  060 072 012            1740  STATK   DB      '0:',NL
000.005                         1741  STATL   EQU     *-STATJ
                                1742
047.204  114 141 142            1743          DB      'Label:',' '+200Q
```

```
                        1746  ***     BYE
                        1747  *
                        1748  *       BYE DISMOUNTS BOTH DISKS AND REBOOTS THE SYSTEM
                        1749  *
                        1750
047.213  301            1751  BYE     POP     B
                        1752
047.214  315 324 047     1753          CALL    LOADOV          LOAD OVERLAYS
047.217  041 270 047     1754          LXI     H,BYEC
047.222  315 244 047     1755          CALL    BYE.            DISMOUNT SY2:, IF MOUNTED
047.225  041 263 047     1756          LXI     H,BYEB
047.230  315 244 047     1757          CALL    BYE.            DISMOUNT SY1:, IF MOUNTED
047.233  041 256 047     1758          LXI     H,BYEA
047.236  315 244 047     1759          CALL    BYE.            DISMOUNT SY0:, IF MOUNTED
                        1760
047.241  257            1761          XRA     A
047.242  377 000         1762          DB      SYSCALL+.EXIT   RETURN TO RE-BOOT
                        1763
                        1764  *       DISMOUNT DEVICE WITHOUT REGARD TO WHETHER MOUNTED OR NOT
                        1765
047.244  377 201         1766  BYE.    DB      SYSCALL+.DMOUN
047.246  320             1767          RNC                     NO ERROR
047.247  376 042         1768          CPI     EC.NVM
047.251  310             1769          RZ                      ERROR IS ONLY NOT MOUNTED ERROR
047.252  067             1770          STC                     RESTORE ERROR FLAG
047.253  303 113 043     1771          JMP     ERROR
                        1772
047.256  123 131 060     1773  BYEA    DB      'SY0:',0
047.263  123 131 061     1774  BYEB    DB      'SY1:',0
047.270  123 131 062     1775  BYEC    DB      'SY2:',0
000.001                 1776          IF      DEBUG
                        1777          STL     'DEBUG COMMANDS'
                        1778          EJECT
                        1779  **      TRAP - TRAP TO HBUG
                        1780
                        1781  TRAP    RST     2
                        1782          JMP     SYSCMD          ENTER SYSCMD
                        1783  BUG     SPACE   3,10
                        1784  **      BUG - SAME AS RUN, BUT WITH BUG FLAG
                        1785
                        1786  BUG     CALL    LBUG            LOAD HBUG
                        1787          MVI     A,1
                        1788          STA     40077A
                        1789          JMP     RUN
                        1790  HBUG    SPACE   3,10
                        1791  **      HBUG - LOAD HBUG.
                        1792
                        1793  HBUG    CALL    LBUG            LOAD HBUG
                        1794          JMP     160000A         ENTER IT
                        1795  LBUG    SPACE   3,10
                        1796  **      LBUG = LOAD HBUG
                        1797
                        1798  LBUG    LXI     H,HBUGA
                        1799          MVI     A,0
                        1800          DB      SYSCALL+.OPENR
                        1801          JC      ERROR           IF ERROR
```

```
1802            LXI     B,21000A
1803            LXI     D,160000A
1804            XRA     A
1805            DB      SYSCALL,.READ           READ IT IN
1806            XRA     A
1807            DB      SYSCALL,.CLOSE
1808            RET
1809
1810  HBUGA     DB      'SY0:HBUG.BIN',0
1811            ENDIF
```

```
                                1815  **      CSA     -  CHECK STAND-ALONE
                                1816  *
                                1817  *       CHECK THE STAND-ALONE FLAG.
                                1818  *
                                1819  *       ENTRY:  NONE
                                1820  *
                                1821  *       EXIT:   (PSW)   =  'Z' CLEAR IF FLAG IS     SET
                                1822  *                       =  'Z' SET   IF FLAG IS NOT SET
                                1823  *
                                1824  *       USES:   (PSW)
                                1825  *
                                1826
047.275  325                   1827  CSA     PUSH    D
047.276  345                   1828          PUSH    H
047.277  021 001 000           1829          LXI     D,M.SALO
047.302  052 346 040           1830          LHLD    S.DLINK
047.305  031                   1831          DAD     D               (HL) => SALONE
047.306  176                   1832          MOV     A,M
047.307  247                   1833          ANA     A
047.310  341                   1834          POP     H
047.311  321                   1835          POP     D
047.312  311                   1836          RET


                                1838  **      FEC - FIND END OF COMMAND LINE.
                                1839  *
                                1840  *       FEC LOCATES THE END OF  THE CURRENT COMMAND LINE.
                                1841  *
                                1842  *       ENTRY   (BC) = START OF LINE
                                1843  *       EXIT    (HL) = ADDRESS OF TERMINATING 00 BYTE
                                1844  *       USES    A,F,H,L
                                1845
                                1846
047.313  140                   1847  FEC     MOV     H,B
047.314  151                   1848          MOV     L,C
047.315  176                   1849  FEC1    MOV     A,M
047.316  247                   1850          ANA     A
047.317  310                   1851          RZ                      AT END
047.320  043                   1852          INX     H
047.321  303 315 047           1853          JMP     FEC1


                                1855  **      LOADOV  - LOAD OVERLAYS
                                1856  *
                                1857  *       LOADOV LOADS BOTH OVL0, AND OVL1.
                                1858  *
                                1859  *       ENTRY:  NONE
                                1860  *
                                1861  *       EXIT:   IF  ERROR
                                1862  *                 TO  ERROR
                                1863  *               ELSE
                                1864  *                 TO  CALLER
```

```
                                1885  *
                                1886  *       USES:   ALL
                                1887  *
                                1888
047.324  076 000                1889  LOADOV  MVI     A,OVL0
047.326  377 010                1890          DB      SYSCALL,.LOADO
047.330  332 113 043            1891          JC      ERROR
047.333  076 001                1892          MVI     A,OVL1
047.335  377 010                1893          DB      SYSCALL,.LOADO
047.337  332 113 043            1894          JC      ERROR
047.342  311                    1895          RET


                                1897  **      PCL - PASS COMMAND LINEE.
                                1898  *
                                1899  *       PCL PASSES A COMMAND LINE INTO THE STACK, FOR USE BY THE PROGRAM
                                1900  *       WHICH WILL BE 'LINK'ED TO.
                                1901  *
                                1902  *       THE N BYTES ARE PUT IN THE STACK STARTING AT 'STACK-N' TO 'STACK-1'
                                1903  *
                                1904  *       * * NOTE * *
                                1905  *       THIS ROUTINE PLAYS WITH THE STACK.  IT IS ENTERED VIA A CALL,
                                1906  *       BUT IT THEN EMPTIES THE STACK TO SETUP THE COMMAND LINE.  THUS, 'PCL'S CALLER
                                1907  *       MUST NOT TRY TO RETURN TO IT'S CALLER.
                                1908  *
                                1909  *       ENTRY   (BC) = LINE ADDRESS
                                1910  *       EXIT    TO CALLER
                                1911  *               (SP) = #STACK-N
                                1912  *       USES    ALL
                                1913
                                1914
047.343  341                    1915  PCL     POP     H               (HL) = RETURN ADDRESS
047.344  363                    1916          DI                      NO INTERRUPTS WHILE PLAYING WITH STACK
047.345  042 000 050            1917          SHLD    PCLA            SET RETURN ADDRESS
047.350  021 200 042            1918          LXI     D,STACK
047.353  315 313 047            1919          CALL    FEC             FIND END OF COMMAND
047.356  012                    1920          LDAX    B
047.357  247                    1921          ANA     A
047.360  312 374 047            1922          JZ      PCL2            HAVE NO LINE TO PASS
                                1923
                                1924  *       GOT A LINE.  MOVE INTO STACK AREA
                                1925
047.363  176                    1926  PCL1    MOV     A,M
047.364  033                    1927          DCX     D
047.365  022                    1928          STAX    D               STORE
047.366  175                    1929          MOV     A,L
047.367  271                    1930          CMP     C
047.370  053                    1931          DCX     H
047.371  302 363 047            1932          JNE     PCL1            MORE TO GO
                                1933
047.374  353                    1934  PCL2    XCHG
047.375  371                    1935          SPHL                    SET STACK POINTER BELOW DATA
047.376  373                    1936          EI
047.377  303 377 047            1937          JMP     *               EXIT
```

```
050.000                    1918  PCLA    EQU     *-2


                           1920  **      PIP - ENVOKE 'PIP'
                           1921  *
                           1922  *       PIP IS ENTERED (VIA A JMP) TO CAUSE A LINK TO PIP.
                           1923  *
                           1924  *       ENTRY   (BC) = COMMAND LINE FWA
                           1925  *       EXIT    TO PIP IF LINK IS OK
                           1926  *               TO SYSCMD VIA ERRMSG OTHERWISE
                           1927  *       USES    ALL
                           1928
                           1929
050.002  315 343 047       1930  PIP     CALL    PCL             PASS COMMAND LINE
050.005  041 075 050       1931          LXI     H,PIPA
050.010  377 040           1932          DB      SYSCALL,.LINK
                           1933
                           1934  *       COULDNT LINK TO PIP
                           1935
050.012  315 136 031       1936          CALL    $TYPTX
050.015  012 007 106       1937          DB      NL;BELL;'File SY0:PIP.ABS Required For This Command';ENL
050.072  303 213 043       1938          JMP     SYSCMD
                           1939
050.075  123 131 060       1940  PIPA    DB      'SY0:PIP.ABS',0
```

```
050.111                 1943            XTEXT   BITS


                        1945X **        BITS    -  BIT SET
                        1946X *
                        1947X *         BITS SETS THE SPECIFIED BIT IN THE ACCUMULATOR.
                        1948X *
                        1949X *         ENTRY:  A       = ORIGINAL  A
                        1950X *                 B       = NUMBER OF BIT TO SET ( 7=HIGH,...,0=LOW )
                        1951X *
                        1952X *         EXIT:   A       = ORIGINAL  A  WITH  BIT(B) SET
                        1953X *
                        1954X *         USES:   PSW
                        1955X *
                        1956X
050.111  305           1957X BITS      PUSH    B
                        1958X
050.112  365           1959X           PUSH    PSW
050.113  076 200       1960X           MVI     A,10000000B
050.115  004           1961X           INR     B
050.116  007           1962X BITS1     RLC
050.117  005           1963X           DCR     B
050.120  302 116 050   1964X           JNZ     BITS1
                        1965X
050.123  117           1966X           MOV     C,A
050.124  361           1967X           POP     PSW
050.125  261           1968X           ORA     C
                        1969X
050.126  301           1970X           POP     BC
050.127  311           1971X           RET
050.130                 1972            XTEXT   CAD


                        1974X **        $CAD - CODE AUGUSTAN DATE.
                        1975X *
                        1976X *         $CAD IS CALLED TO CODE AN AUGUSTAN DATE INTO THE FORM:
                        1977X *
                        1978X *
                        1979X *         -----------------------------------
                        1980X *         I  0  I  6 BITS  I  4 BITS  I  5 BITS  I
                        1981X *         -----------------------------------
                        1982X *                 YEAR-70        MON        DAY
                        1983X *                  1-63          1-12       1-31
                        1984X *
                        1985X *         FROM THE FORM:
                        1986X *
                        1987X *         DD-MMM-YY
                        1988X *
                        1989X *         ENTRY   (HL) = ADDRESS OF STRING
                        1990X *         EXIT    'C' CLEAR IF OK
                        1991X *                 (DE) = 15 BIT VALUE
                        1992X *                 (HL) ADVANCED PAST '-YY'
```

```
                                1993X *                 'C' SET IF ERROR
                                1994X *         USES    ALL
                                1995X
                                1996X
050.130  315 212 051            1997X $CAD      CALL    $DDD            DECODE DECIMAL DIGITS
050.133  330                    1998X           RC                      ERROR
050.134  172                    1999X           MOV     A,D
050.135  247                    2000X           ANA     A
050.136  067                    2001X           STC                     ASSUME TOO LARGE
050.137  300                    2002X           RNZ                     TOO LARGE
050.140  173                    2003X           MOV     A,E
050.141  247                    2004X           ANA     A
050.142  067                    2005X           STC
050.143  310                    2006X           RZ                      TOO SMALL FOR DD
050.144  376 040                2007X           CPI     32
050.146  077                    2008X           CMC
050.147  330                    2009X           RC                      TOO LARGE
050.150  353                    2010X           XCHG                    (HL) = DAY
050.151  076 040                2011X           MVI     A,100000B
050.153  205                    2012X           ADD     L
050.154  157                    2013X           MOV     L,A             COUNT 1ST MONTH
050.155  353                    2014X           XCHG                    (DE) = DD*16+1, (HL) = ADDRESS
                                2015X
                                2016X *         DECODE MONTH
                                2017X
050.156  325                    2018X           PUSH    D               SAVE DD*16+1
050.157  176                    2019X           MOV     A,M
050.160  043                    2020X           INX     H
050.161  376 055                2021X           CPI     '-'
050.163  302 225 050            2022X           JNE     CAD2            FORMAT ERROR
050.166  021 273 050            2023X           LXI     D,CADA          (DE) = MONTH TABLE ADDRESS
050.171  001 003 000            2024X CAD1      LXI     B,3
050.174  345                    2025X           PUSH    H               SAVE TEXT ADDRESS, CADA ADDRESS
050.175  325                    2026X           PUSH    D
050.176  315 060 030            2027X           CALL    $COMP           COMPARE
050.201  321                    2028X           POP     D               (DE) = *CADA* ADDRESS
050.202  312 230 050            2029X           JE      CAD3            GOT MONTH
050.205  341                    2030X           POP     H               (HL) = BUFFER ADDRESS OF MMM-YY
050.206  023                    2031X           INX     D
050.207  023                    2032X           INX     D
050.210  023                    2033X           INX     D               TRY NEXT MONTH
050.211  343                    2034X           XTHL
050.212  076 040                2035X           MVI     A,100000B
050.214  315 101 030            2036X           CALL    $DADA.          COUNT MONTH
050.217  343                    2037X           XTHL
050.220  032                    2038X           LDAX    D               (A) = ENTRY IN CADA
050.221  247                    2039X           ANA     A
050.222  302 171 050            2040X           JNZ     CAD1            MORE MONTHS TO GO
                                2041X
                                2042X *         ERROR
                                2043X
050.225  341                    2044X CAD2      POP     H               CLEAR STACK
050.226  067                    2045X           STC
050.227  311                    2046X           RET                     FLAG ERROR
                                2047X
                                2048X *         CRACK -YY
```

```
                          2049X
050.230  301              2050X CAD3    POP     B               DISCARD ADDRESS IF MMM-YY
050.231  176              2051X         MOV     A,M
050.232  376 055          2052X         CPI     '-'
050.234  302 225 050      2053X         JNE     CAD2            NOT -
050.237  043              2054X         INX     H
050.240  315 212 051      2055X         CALL    $DDD            DECODE DECIMAL DIGITS
050.243  332 225 050      2056X         JC      CAD2            IF ERROR
050.246  172              2057X         MOV     A,D
050.247  247              2058X         ANA     A
050.250  302 225 050      2059X         JNZ     CAD2            ERROR
050.253  173              2060X         MOV     A,E             (A) = YEAR
050.254  326 106          2061X         SUI     70              SUBTRACT DISPLACEMENT
050.256  332 225 050      2062X         JC      CAD2            ERROR
050.261  376 077          2063X         CPI     63
050.263  322 225 050      2064X         JNC     CAD2            TOO LARGE
050.266  321              2065X         POP     D               (DE) = MONTH AND DAY
050.267  207              2066X         ADD     A               (A) = YEAR*2
050.270  202              2067X         ADD     D
050.271  127              2068X         MOV     D,A             MERGE WITH REST OF IT
050.272  311              2069X         RET
                          2070X
050.273                   2071X CADA    DS      0               TABLE OF MONTHS
050.273  112 101 116      2072X         DB      'JANFEBMARAPRMAYJUNJULAUGSEPOCTNOVDEC'+0
050.340                   2073          XTEXT   CDEHL


                          2075X **      $CDEHL - COMPARE (DE) TO (HL)
                          2076X *
                          2077X *       $CDEHL COMPARES (DE) TO (HL) FOR EQUALITY.
                          2078X *
                          2079X *       ENTRY   NONE
                          2080X *       EXIT    'Z' SET IF (DE) = (HL)
                          2081X *       USES    A,F
                          2082X
                          2083X
030.216                   2084X $CDEHL  EQU     30216A          IN H17 ROM
050.340                   2085          XTEXT   CCO


                          2087X **      $CCO - CLEAR CONTROL-O
                          2088X *
                          2089X *       $CCO IS CALLED TO CLEAR THE EFFECT OF THE CTL-O CHARACTER.
                          2090X *
                          2091X *       ENTRY   NONE
                          2092X *       EXIT    NONE
                          2093X *       USES    NONE
                          2094X
                          2095X
050.340  315 054 031      2096X $CCO    CALL    $SAVALL         SAVE REGISTERS
050.343  076 004          2097X         MVI     A,I.CONFL
050.345  001 001 000      2098X         LXI     B,CO.FLG        CLEAR CO.FLG
```

```
050.350  377 006       2099X          DB      SYSCALL,.CONSL
050.352  303 047 031   2100X          JMP     $RSTALL          RESTORE REGISTERS AND RETURN
050.355                2101           XTEXT   COMP


                       2103X **       $COMP - COMPARE TWO CHARACTER STRINGS.
                       2104X *
                       2105X *        $COMP COMPARES TWO BYTE STRINGS.
                       2106X *
                       2107X *        ENTRY   (C) = COMPARE COUNT
                       2108X *                (DE) = FWA OF STRING #1
                       2109X *                (HL) = FWA OF STRING #2
                       2110X *        EXIT    'Z' CLEAR, IS MIS-MATCH
                       2111X *                 (C) = LENGTH REMAINING
                       2112X *                (DE) = ADDRESS OF MISMATCH IN STRING#1
                       2113X *                (HL) = ADDRESS OF MISMATCH IN STRING #2
                       2114X *                'Z' SET, HAVE MATCH
                       2115X *                 (C) = 0
                       2116X *                (DE) = (DE) + (OC)
                       2117X *                (HL) = (HL) + (OC)
                       2118X *        USES    A,F,C,D,E,H,L
                       2119X
                       2120X
030.060                2121X $COMP    EQU     30060A           IN H17 ROM
050.355                2122           XTEXT   CPF


                       2124X **       $CPF - COPY FILE NAME
                       2125X *
                       2126X *        $CPF COPIES A FILE NAME FROM ONE LOCATION TO ANOTHER.
                       2127X *
                       2128X *        THE CHARACTERS ARE COPIED UNTIL A DELIMITER (',', ' ', '=', OR 00)
                       2129X *        IS FOUND.
                       2130X *
                       2131X *        THE FILENAME IS THEN TERMINATED WITH A 00 BYTE.
                       2132X *
                       2133X *        ENTRY   (DE) = FROM ADDRESS
                       2134X *                (HL) = TO ADDRESS
                       2135X *        EXIT    'C' CLEAR IF OK
                       2136X *                 (DE) = ADVANCED PAST NAME AND DELIMITER
                       2137X *                 (HL) POINTS TO 00 BYTE OF DESTINATION
                       2138X *                 (A) = DELIMITER
                       2139X *                 'C' SET IF ERROR
                       2140X *        USES    ALL
                       2141X
                       2142X
050.355  006 022       2143X $CPF     MVI     B,FB.NAML+1              SET MAX LENGTH
050.357  032           2144X $CPF1    LDAX    D
050.360  247           2145X          ANA     A
050.361  312 014 051   2146X          JZ      $CPF2            END
050.364  023           2147X          INX     D
050.365  376 054       2148X          CPI     ','
```

```
051.367  312 014 051  2149X          JE      $CPF2
050.372  376 075      2150X          CPI     '='
050.374  312 014 051  2151X          JE      $CPF2
050.377  376 040      2152X          CPI     ' '
051.001  312 014 051  2153X          JE      $CPF2           IS BLANK
051.004  167          2154X          MOV     M,A             COPY
051.005  043          2155X          INX     H
051.006  005          2156X          DCR     B
051.007  302 357 050  2157X          JNZ     $CPF1           IF MORE GO TO
051.012  067          2158X          STC                     OVERFLOW OF AREA
051.013  311          2159X          RET
                      2160X
                      2161X *        DONE.
                      2162X
051.014  066 000      2163X $CPF2    MVI     M,0             TERMINATE
051.016  311          2164X          RET
051.017               2165           XTEXT   CRLF



                      2167X **       $CRLF - TYPE CARRIAGE RETURN/ LINE FEED
                      2168X *
                      2169X *        $CRLF IS USED TO GENERATE PADDED CRLF'S.
                      2170X *
                      2171X *        ENTRY   NONE
                      2172X *        EXIT    (A) = 0
                      2173X *        USES    A,F
                      2174X
                      2175X
051.017  076 012      2176X $CRLF    MVI     A,NL
051.021  377 002      2177X          DB      SYSCALL+.SCOUT
051.023  257          2178X          XRA     A
051.024  311          2179X          RET
051.025               2180           XTEXT   DAD



                      2182X **       $DAD - DECODE AUGUSTAN DATE.
                      2183X *
                      2184X *        $DAD DECODES A 15 BIT DATE CODE OF THE FORMAT:
                      2185X *
                      2186X *        ---------------------------------------
                      2187X *        I  0  I  6 BITS  I  4 BITS  I  5 BITS  I
                      2188X *        ---------------------------------------
                      2189X *                YEAR-70       MON       DAY
                      2190X *                 1-63        1-12      1-31
                      2191X *
                      2192X *        TO THE FORM:
                      2193X *
                      2194X *        DD-MMM-YY
                      2195X *
                      2196X *        ENTRY   (DE) = 15 BIT VALUE
                      2197X *                (HL) = ADDRESS FOR DECODE
                      2198X *        EXIT    'C' CLEAR IF OK
```

```
                                2199X *              (DE) = (DE)+9
                                2200X *              'C' SET IF ERROR
                                2201X *       USES   ALL
                                2202X
                                2203X
051.025  102                    2204X $DAD    MOV    B,D
051.026  113                    2205X         MOV    C,E
051.027  021 040 000            2206X         LXI    D,32
051.032  345                    2207X         PUSH   H              SAVE ADDRESS
051.033  315 106 030            2208X         CALL   $DU66          (DE) = DAY, (HL) = YEAR & MONTH
051.036  343                    2209X         XTHL                  (HL) = ADDRESS
051.037  102                    2210X         MOV    B,D
051.040  113                    2211X         MOV    C,E
051.041  173                    2212X         MOV    A,E
051.042  247                    2213X         ANA    A
051.043  312 143 051            2214X         JZ     DAD1           BAD VALUE
051.046  076 002                2215X         MVI    A,2
051.050  315 157 031            2216X         CALL   $UDD           UNPACK DAY
051.053  066 055                2217X         MVI    M,'-'
051.055  043                    2218X         INX    H
051.056  301                    2219X         POP    B              (BC) = YEAR & MONTH
051.057  021 020 000            2220X         LXI    D,16
051.062  345                    2221X         PUSH   H              SAVE ADDRESS
051.063  315 106 030            2222X         CALL   $DU66
051.066  343                    2223X         XTHL                  (HL) = ADDRESS, ((SP)) = YEAR
051.067  173                    2224X         MOV    A,E
051.070  207                    2225X         ADD    A
051.071  203                    2226X         ADD    E              (A) = 3*MONTH
051.072  312 143 051            2227X         JZ     DAD1           BAD VALUE
051.075  376 047                2228X         CPI    13*3
051.077  322 143 051            2229X         JNC    DAD1           TOO LARGE
051.102  353                    2230X         XCHG                  (DE) = ADDRESS
051.103  041 143 051            2231X         LXI    H,DADB-3
051.106  315 101 030            2232X         CALL   $DADA,         (HL) = ADDRESS OF MONTH
051.111  001 003 000            2233X         LXI    B,3
051.114  353                    2234X         XCHG                  (HL) = BUFFER ADDR, (DE) = ADDR IN 'DADB'
051.115  315 252 030            2235X         CALL   $MOVE          MOVE MONTH IN
051.120  066 055                2236X         MVI    M,'-'
051.122  043                    2237X         INX    H
051.123  301                    2238X         POP    B              (BC) = YEAR
051.124  171                    2239X         MOV    A,C
051.125  306 106                2240X         ADI    70
051.127  376 144                2241X         CPI    100
051.131  077                    2242X         CMC
051.132  330                    2243X         RC                    TOO LARGE
051.133  117                    2244X         MOV    C,A            (BC) = YEAR
051.134  076 002                2245X         MVI    A,2
051.136  315 157 031            2246X         CALL   $UDD           UNPACK YEAR
051.141  247                    2247X         ANA    A
051.142  311                    2248X         RET
                                2249X
                                2250X *       ILLEGAL FORMAT, (NOT ALL ILLEGALS EXIT HERE!)
                                2251X
051.143  341                    2252X DAD1    POP    H              RESTORE STACK
051.144  067                    2253X         STC                   FLAG ERROR
051.145  311                    2254X         RET
```

```
                                2255X
051.146  112 141 156          2256X DADB    DB      'JanFebMarAprMayJunJulAugSepOctNovDec'
051.212                       2257          XTEXT   DADA


                                2259X **      $DADA - PERFORM (H,L) = (H,L) + (0,A)
                                2260X *
                                2261X *       ENTRY   (H,L) = BEFORE VALUE
                                2262X *               (A) = BEFORE VALUE
                                2263X *       EXIT    (H,L) = (H,L) + (0,A)
                                2264X *               'C' SET IF OVERFLOW
                                2265X *       USES    F,H,L
                                2266X
                                2267X
030.072                         2268X $DADA   EQU     30072A          IN H17 ROM
051.212                         2269          XTEXT   DADA2


                                2271X **      $DADA. - ADD (0,A) TO (H,L)
                                2272X *
                                2273X *       ENTRY   NONE
                                2274X *       EXIT    (HL) = (HL) + (0A)
                                2275X *       USES    A,F,H,L
                                2276X
                                2277X
030.101                         2278X $DADA.  EQU     30101A          IN H17 ROM
051.212                         2279          XTEXT   DDD


                                2281X **      $DDD - DECODE DECIMAL DIGITS.
                                2282X *
                                2283X *       $DDD DECODES A STRING OF DECIMAL DIGITS INTO A DECIMAL INTEGER.
                                2284X *
                                2285X *       THE CHARACTERS ARE TAKEN OUT OF MEMORY. CONVERSION STOPS WITH THE
                                2286X *       FIRST NON-DIGIT CHARACTER FOUND.
                                2287X *
                                2288X *       ENTRY   (HL) = ADDRESS OF CHARACTERS
                                2289X *       EXIT    'C' CLEAR IF OK
                                2290X *                (DE) = NUMBER
                                2291X *                (HL) = INDEX OF FIRST NON-DIGIT ENCOUNTERED
                                2292X *               'C' SET IF ERROR
                                2293X *       USES    A,F,D,E,H,L
                                2294X
                                2295X
051.212  021 000 000           2296X $DDD    LXI     D,0             (DE) = ACCUM
                                2297X
051.215  176                    2298X $DDD1   MOV     A,M
051.216  326 060                2299X         SUI     '0'
051.220  077                    2300X         CMC
051.221  320                    2301X         RNC                     TOO SMALL
```

```
051,222  376 012        2302X          CPI     10
051,224  320            2303X          RNC                     TOO LARGE
051,225  043            2304X          INX     H               ADVANCE ADDRESS
051,226  345            2305X          PUSH    H               SAVE (HL)
051,227  315 324 030    2306X          CALL    $MU10           (HL) = ACCUM*10
051,232  353            2307X          XCHG                    (DE) = ACCUM
051,233  341            2308X          POP     H               (HL) = ADDRESS OF STRING
051,234  330            2309X          RC                      OVERFLOW
051,235  203            2310X          ADD     E
051,236  137            2311X          MOV     E,A
051,237  076 000        2312X          MVI     A,0
051,241  212            2313X          ADC     D
051,242  127            2314X          MOV     D,A
051,243  322 215 051    2315X          JNC     $DDD1           NOT OVERFLOW
051,246  311            2316X          RET
051,247                 2317           XTEXT   DTB


                        2319X **       $DTB - DELETE TRAILING BLANKS.
                        2320X *
                        2321X *        $DTB DELETES THE TRAILING BLANKS FROM A CODED LINE.
                        2322X *
                        2323X *        ENTRY   (HL) = LINE FWA
                        2324X *        EXIT    (A) = LENGTH OF RESULT (ENCLUDING 00 TERMINATOR BYTE)
                        2325X *        USES    A,F
                        2326X
                        2327X
051,247  325            2328X $DTB     PUSH    D               SAVE (DE)
051,250  124            2329X          MOV     D,H
051,251  135            2330X          MOV     E,L             (DE) = FWA
051,252  033            2331X          DCX     D               (DE) = FWA-1
051,253  176            2332X $DTB1    MOV     A,M
051,254  043            2333X          INX     H
051,255  247            2334X          ANA     A               FIND END OF LINE
051,256  302 253 051    2335X          JNZ     $DTB1
051,261  053            2336X          DCX     H               (HL) = ADDRESS OF TERMINATING ZERO BYTE
                        2337X
                        2338X *        GOT END OF LINE, DELETE TRAILING BLANKS
                        2339X
051,262  053            2340X $DTB2    DCX     H               BACKUP ONE CHARACTER
051,263  315 216 030    2341X          CALL    $CDEHL
051,266  312 277 051    2342X          JE      $DTB3           GONE PAST FRONT OF LINE, MUST BE ALL BLANKS
051,271  176            2343X          MOV     A,M
051,272  376 040        2344X          CPI     ' '
051,274  312 262 051    2345X          JE      $DTB2           GOT BLANK
                        2346X
                        2347X *        HAVE TRIMED LINE, COMPUTE LENGTH
                        2348X
051,277  043            2349X $DTB3    INX     H
051,300  066 000        2350X          MVI     M,0             TERMINATE LINE
051,302  175            2351X          MOV     A,L
051,303  223            2352X          SUB     E               (A) = LENGTH +1 (FOR 00 BYTE)
051,304  353            2353X          XCHG
051,305  043            2354X          INX     H               (HL) = LINE FWA
```

```
051.306  321          2355X          POP     D               RESTORE (DE)
051.307  311          2356X          RET
051.310               2357           XTEXT   DU66


                      2359X **       $DU66 - UNSIGNED 16 / 16 DIVIDE.
                      2360X *
                      2361X *        (HL) = (BC)/(DE)
                      2362X *
                      2363X *        ENTRY   (BC), (DE) PRESET
                      2364X *        EXIT    (HL) = RESULT
                      2365X *                (DE) = REMAINDER
                      2366X *        USES    ALL
                      2367X
                      2368X
030.106               2369X $DU66    EQU     30106A          IN H17 ROM
051.310               2370           XTEXT   ECI


                      2372X **       ECI - ENABLE CONSOLE INTERRUPTS
                      2373X *
                      2374X *        ENTRY   NONE
                      2375X *        EXIT    NONE
                      2376X *        USES    (PSW)
                      2377X *
                      2378X
051.310  072 343 040  2379X ECI      LDA     S.CDB
051.313  376 001      2380X          CPI     CDB.H84
051.315  312 325 051  2381X          JZ      ECI1                    IF  8250
                      2382X
                      2383X *        HAVE 8251
                      2384X
051.320  076 027      2385X          MVI     A,UCI.RE+UCI.TE+UCI.ER+UCI.IE
051.322  323 373      2386X          OUT     SC.UART+USR
051.324  311          2387X          RET
                      2388X
                      2389X *        HAVE 8250
                      2390X
051.325  076 001      2391X ECI1     MVI     A,UC.EDA
051.327  323 351      2392X          OUT     SC.ACE+UR.IER
051.331  311          2393X          RET
051.332               2394           XTEXT   FST
```

```
                                2396X **      $FST - FIND IN SERIAL TABLE
                                2397X *
                                2398X *       $FST SEARCHES A SERIAL TABLE FOR
                                2399X *       A SPECIFIC KEY
                                2400X *
                                2401X *       ENTRY   (HL) = ADDR. OF TABLE
                                2402X *               (DE) = ADDR. OF SEARCH KEY
                                2403X *       EXIT    (DE) = UNCHANGED
                                2404X *               'Z' CLEARED IF NO MATCH FOUND
                                2405X *                 (HL) = ADDR. OF NEXT AVAILABLE BYTE
                                2406X *               'Z' SET IF MATCH FOUND
                                2407X *                 (HL) = ADDR. OF FIRST DATA BYTE
                                2408X *       USES    A,F,H,L
                                2409X
                                2410X
                                2411X
051.332  305                   2412X $FST    PUSH    B               SAVE REGISTERS
051.333  325                   2413X         PUSH    D
                                2414X
                                2415X *       SAVE TABLE LIMIT AND DATA BYTE COUNT
                                2416X
051.334  136                   2417X         MOV     E,M             GET AND SAVE TABLE LIMIT
051.335  043                   2418X         INX     H               (HL) = 2ND BYTE OF SIZE
051.336  126                   2419X         MOV     D,M
051.337  353                   2420X         XCHG
051.340  042 032 052           2421X         SHLD    $FST.L          SAVE MAX. TABLE SIZE
                                2422X
051.343  353                   2423X         XCHG
051.344  043                   2424X         INX     H               (HL) = # OF BYTES OF DATA/ENTRY
051.345  176                   2425X         MOV     A,M
051.346  062 034 052           2426X         STA     $FST.C
051.351  043                   2427X         INX     H               (HL) = BEGINNING OF DATA
051.352  321                   2428X FST1    POP     D               RESTORE ADDR. TO SEARCH KEY
051.353  325                   2429X         PUSH    D
                                2430X
                                2431X *       CHECK FOR END OF DATA
                                2432X
051.354  176                   2433X         MOV     A,M
051.355  267                   2434X         ORA     A               AT END OF DATA? ((A) = 0)
051.356  302 365 051           2435X         JNZ     FST2            NO, START MATCHING
051.361  074                   2436X         INR     A               CLEAR 'Z'
051.362  321                   2437X         POP     D
051.363  301                   2438X         POP     B               RESTORE REGISTERS
051.364  311                   2439X         RET
                                2440X
051.365  032                   2441X FST2    LDAX    D               (A) = KEY CHAR.
051.366  276                   2442X         CMP     M               COMPARE TO TABLE
051.367  302 003 052           2443X         JNE     FST3            NO MATCH, FIND NEXT KEY
051.372  247                   2444X         ANA     A               END OF KEY?
051.373  372 025 052           2445X         JM      FST4            YES, SET UP FOR EXIT
051.376  043                   2446X         INX     H
051.377  023                   2447X         INX     D
052.000  303 365 051           2448X         JMP     FST2
                                2449X
052.003  176                   2450X FST3    MOV     A,M             SEARCH FOR END OF KEY
052.004  247                   2451X         ANA     A               TEST CHAR.
```

```
052.005  043            2452X          INX     H
052.006  362 003 052     2453X          JP      FST3            CONTINUE SEARCH
052.011  072 034 052     2454X          LDA     $FST.C          (A) = # OF BYTES OF DATA/ENTRY
052.014  205             2455X          ADD     L
052.015  157             2456X          MOV     L,A
052.016  076 000         2457X          MVI     A,0
052.020  214             2458X          ADC     H
052.021  147             2459X          MOV     H,A             (HL) = HEAD OF NEXT KEY
052.022  303 352 051     2460X          JMP     FST1            COMPARE NEXT KEY
                         2461X
052.025  257             2462X FST4     XRA     A               SET 'Z' FOR EXIT
052.026  043             2463X          INX     H               (HL) = FIRST BYTE OF DATA
052.027  321             2464X          POP     D               RESTORE REGISTERS
052.030  301             2465X          POP     B
052.031  311             2466X          RET                     EXIT
                         2467X
                         2468X
052.032                  2469X $FST.L   DS      2
052.034                  2470X $FST.C   DS      1
052.035                  2471           XTEXT   GNL



                         2473X **       $GNL - GUARANTEE NEW LINE.
                         2474X *
                         2475X *        $GNL GUARANTEES THE START OF A NEW LINE BY ISSUING A CRLF
                         2476X *        IF THE CURSOR IS NOT AT COLUMN 1..
                         2477X *
                         2478X *        ENTRY   NONE
                         2479X *        EXIT    NONE
                         2480X *        USES    ALL
                         2481X
                         2482X
052.035  076 002         2483X $GNL     MVI     A,1.CUSOR
052.037  001 000 000     2484X          LXI     B,0
052.042  377 006         2485X          DB      SYSCALL+.CONSL          READ CURSOR
052.044  075             2486X          DCR     A
052.045  310             2487X          RZ                      AT COLUMN 1
052.046  303 017 051     2488X          JMP     $CRLF           NEW LINE
052.051                  2489           XTEXT   ILDEHL



                         2491X **       ILDEHL  -  INDEXED LOAD OF DE FROM HL
                         2492X *
                         2493X *        'DE' GET THE FULL WORD VALUE POINTED TO BY 'HL', AND 'HL' IS
                         2494X *        INCREMENTED BY TWO.
                         2495X *
                         2496X *        ENTRY:  HL      = ADDRESS OF FULL WORD VALUE
                         2497X *
                         2498X *        EXIT:   DE      = (HL)
                         2499X *                HL      = HL + 2
                         2500X *
                         2501X *        USES:   DE
```

```
                        2502X *
                        2503X
052.051  136           2504X ILDEHL  MOV     E,M
052.052  043           2505X         INX     H
052.053  126           2506X         MOV     D,M
052.054  043           2507X         INX     H
052.055  311           2508X         RET
052.056                2509          XTEXT   INDL



                        2511X **      $INDL - INDEXED LOAD.
                        2512X *
                        2513X *       $INDL LOADS DE WITH THE TWO BYTES AT (HL)+DISPLACMENT
                        2514X *
                        2515X *       THIS ACTS AS AN INDEXED FULL WORD LOAD.
                        2516X *
                        2517X *       (DE) = ( (HL) + DSPLACEMENT )
                        2518X *
                        2519X *       ENTRY   ((RET)) = DISPLACMENT (FULL WORD)
                        2520X *               (HL) = TABLE ADDRESS
                        2521X *       EXIT    TO (RET+2)
                        2522X *       USES    A,F,D,E
                        2523X
                        2524X
030.234                 2525X $INDL   EQU     30234A          IN H17 ROM
052.056                 2526          XTEXT   INDXX



                        2528X **      $INDLB  -  INDEXED LOAD BYTE
                        2529X *
                        2530X *       BYTE INDEXED LOAD PRIMITIVE
                        2531X *
                        2532X *       ENTRY:  HL      = BASE ADDRESS
                        2533X *               (RET)   = FULL WORD RELOCATION
                        2534X *
                        2535X *       EXIT:   A       = ( HL + (RET) )
                        2536X *
                        2537X *       USES:   A
                        2538X *
                        2539X
052.056  353           2540X $INDLB  XCHG                    DE = BASE
052.057  343           2541X         XTHL                    SAVE .DE.
052.060  325           2542X         PUSH    D               SAVE  BASE
052.061  305           2543X         PUSH    B               SAVE .BC.
                        2544X
052.062  116           2545X         MOV     C,M
052.063  043           2546X         INX     H
052.064  106           2547X         MOV     B,M             BC = OFFSET
052.065  043           2548X         INX     H               HL =  .RET.
                        2549X
052.066  353           2550X         XCHG                    HL = BASE
052.067  011           2551X         DAD     B               HL = BASE + OFFSET
```

```
052.070  176          2552X          MOV     A,M             A  = ( BASE + OFFSET )
052.071  353          2553X          XCHG                    HL =  .RET.
                      2554X
052.072  301          2555X          POP     B               RESTORE  .BC.
052.073  321          2556X          POP     D               RESTORE  BASE
052.074  343          2557X          XTHL                    HL = .DE. ; (SP) = .RET.
052.075  353          2558X          XCHG                    DE = .DE. ; HL = BASE
052.076  311          2559X          RET


                      2561X **       $INDS   -  INDEXED STORE
                      2562X *
                      2563X *        INDEXED STORE PRIMITIVE.
                      2564X *
                      2565X *        ENTRY:  HL = BASE ADDRESS
                      2566X *                DE = VALUE TO STORE
                      2567X *
                      2568X *        EXIT:   ( HL + (RET) ) = DE
                      2569X *
                      2570X *        USES:   NONE
                      2571X *
                      2572X
052.077  315 155 053  2573X $INDS    CALL    XCHGBC
052.102  343          2574X          XTHL                    SAVE  .BC.
052.103  325          2575X          PUSH    D
052.104  315 051 052  2576X          CALL    ILDEHL          DE = OFFSET
052.107  315 155 053  2577X          CALL    XCHGBC          BC = .RET.
052.112  353          2578X          XCHG                    DE = BASE ; HL = OFFSET
052.113  031          2579X          DAD     D               HL = BASE + OFFSET
052.114  353          2580X          XCHG
052.115  343          2581X          XTHL                    SAVE  BASE
052.116  353          2582X          XCHG                    DE = VALUE
052.117  315 154 052  2583X          CALL    ISDEHL
052.122  341          2584X          POP     H               HL = BASE
052.123  315 155 053  2585X          CALL    XCHGBC
052.126  343          2586X          XTHL                    RESTORE  .BC.
052.127  315 155 053  2587X          CALL    XCHGBC
052.132  311          2588X          RET


                      2590X **       $INDSB  -  INDEXED BYTE STORE
                      2591X *
                      2592X *        INDEXED BYTE STORE.
                      2593X *
                      2594X *        ENTRY:  A       = VALUE TO STORE
                      2595X *                HL      = BASE ADDRESS
                      2596X *                (RET)   = OFFSET
                      2597X *
                      2598X *        EXIT:   NONE
                      2599X *
                      2600X *        USES:   PSW
                      2601X *
```

```
                        2602X
052.133  353          2603X $INDSB  XCHG            DE = BASE
052.134  343          2604X         XTHL            SAVE  .DE.
052.135  325          2605X         PUSH    D       SAVE  BASE
052.136  305          2606X         PUSH    B       SAVE  .BC.
                        2607X
052.137  116          2608X         MOV     C,M
052.140  043          2609X         INX     H
052.141  106          2610X         MOV     B,M     BC = OFFSET
052.142  043          2611X         INX     H       HL =  .RET.
                        2612X
052.143  353          2613X         XCHG            HL = BASE
052.144  011          2614X         DAD     B       HL = BASE + OFFSET
052.145  167          2615X         MOV     M,A     ( BASE + OFFSET ) = A
052.146  353          2616X         XCHG
                        2617X
052.147  301          2618X         POP     B       RESTORE  .BC.
052.150  321          2619X         POP     D       RESTORE  BASE
052.151  343          2620X         XTHL            HL = .DE. ; (SP) = .RET.
052.152  353          2621X         XCHG            DE = .DE. ; HL = BASE
052.153  311          2622X         RET
052.154                 2623         XTEXT   ISDEHL


                        2625X **      ISDEHL  -  INDEXED STORE OF DE AT HL
                        2626X *
                        2627X *       STORE 'DE' AT THE ADDRESS POINTED TO BY 'HL', AND INCREMENT 'HL'
                        2628X *       BY 2.
                        2629X *
                        2630X *       ENTRY:  DE      = VALUE
                        2631X *               HL      = ADDRESS OF VALUE
                        2632X *
                        2633X *       EXIT:   (HL)    = DE
                        2634X *               HL      = HL + 2
                        2635X *
                        2636X *       USES:   HL
                        2637X *
                        2638X
052.154  163          2639X ISDEHL  MOV     M,E
052.155  043          2640X         INX     H
052.156  162          2641X         MOV     M,D
052.157  043          2642X         INX     H
052.160  311          2643X         RET
052.161                 2644         XTEXT   MCU
```

```
                        2646X **      MCU - MAP LOWER CASE TO UPPER CASE.
                        2647X *
                        2648X *       MCU MAPS A LOWER CASE ALPHABETIC TO UPPER
                        2649X *       CASE.
                        2650X *
                        2651X *       ENTRY   (A) = CHARACTER
                        2652X *       EXIT    (A) = CHARACTER RESULT
                        2653X *       USES    A,F
                        2654X
                        2655X
052.161  376 141       2656X $MCU    CPI     'a'
052.163  330           2657X         RC                      NOT LOWER CASE
052.164  376 173       2658X         CPI     'z'+1
052.166  320           2659X         RNC                     NOT LOWER CASE
052.167  326 040       2660X         SUI     'a'-'A'
052.171  311           2661X         RET
052.172                2662          XTEXT   MLU


                        2664X **      MLU - MAP LOWER CASE LINE TO UPPER CASE.
                        2665X *
                        2666X *       MLU MAPS THE LOWER CASE ALPHABETICS IN A LINE TO UPPER CASE.
                        2667X *
                        2668X *       ENTRY   (HL)  = LINE FWA
                        2669X *       EXIT    NONE
                        2670X *       USES    NONE
                        2671X
                        2672X
052.172  365           2673X $MLU    PUSH    PSW             SAVE (PSW)
052.173  345           2674X         PUSH    H               SAVE FWA
052.174  053           2675X         DCX     H               ANTICIPATE INX H
052.175  043           2676X $MLU1   INX     H
052.176  176           2677X         MOV     A,M             (A)= CHARACTER
052.177  315 161 052    2678X         CALL    $MCU            MAP CHAR TO UPPER
052.202  167           2679X         MOV     M,A
052.203  247           2680X         ANA     A
052.204  302 175 052    2681X         JNZ     $MLU1           MORE TO GO
052.207  341           2682X         POP     H               RESTORE (HL)
052.210  361           2683X         POP     PSW             RESTORE (PSW)
052.211  311           2684X         RET
052.212                2685          XTEXT   MOVE


                        2687X **      $MOVE - MOVE DATA
                        2688X *
                        2689X *       $MOVE MOVES A BLOCK OF BYTES TO A NEW MEMORY ADDRESS.
                        2690X *       IF THE MOVE IS TO A LOWER ADDRESS, THE BYTES ARE MOVED FROM
                        2691X *       FIRST TO LAST.
                        2692X *
                        2693X *       IF THE MOVE IS TO A HIGHER ADDRESS, THE BYTES ARE MOVED FROM
                        2694X *       LAST TO FIRST.
                        2695X *
```

```
                        2696X *       THIS IS DONE SO THAT AN OVERLAPED MOVE WILL NOT 'RIPPLE'.
                        2697X *
                        2698X *       ENTRY   (BC) = COUNT
                        2699X *               (DE) = FROM
                        2700X *               (HL) = TO
                        2701X *       EXIT    MOVED
                        2702X *               (DE) = ADDRESS OF NEXT FROM BYTE
                        2703X *               (HL) = ADDRESS OF NEXT *TO* BYTE
                        2704X *               'C' CLEAR
                        2705X *       USES    ALL
                        2706X
                        2707X
    030.252             2708X $MOVE   EQU     30252A          IN H17 ROM
    052.212             2709          XTEXT   MOVEL


                        2711X **      $MOVEL - MOVE DATA
                        2712X *
                        2713X *       $MOVEL MOVES A BLOCK OF BYTES TO A NEW MEMORY ADDRESS.
                        2714X *       IF THE MOVE IS TO A LOWER ADDRESS, THE BYTES ARE MOVED FROM
                        2715X *       FIRST TO LAST.
                        2716X *
                        2717X *       IF THE MOVE IS TO A HIGHER ADDRESS, THE BYTES ARE MOVED FROM
                        2718X *       LAST TO FIRST.
                        2719X *
                        2720X *       THIS IS DONE SO THAT AN OVERLAPED MOVE WILL NOT 'RIPPLE'.
                        2721X *
                        2722X *       CALL    $MOVEL
                        2723X *       DW      COUNT
                        2724X *       DW      FROM
                        2725X *       DW      TO
                        2726X *
                        2727X *       ENTRY   ((SP)) = RET
                        2728X *               (RET+0) = COUNT (WORD VALUE)
                        2729X *               (RET+2) = FROM
                        2730X *               (RET+4) = TO
                        2731X *       EXIT    TO (RET+6)
                        2732X *               (DE) = ADDRESS OF NEXT FROM BYTE
                        2733X *               (HL) = ADDRESS OF NEXT *TO* BYTE
                        2734X *               'C' CLEAR
                        2735X *       USES    ALL
                        2736X
                        2737X
    052.212  341        2738X $MOVEL  POP     H               (HL) = RET
    052.213  116        2739X         MOV     C,M
    052.214  043        2740X         INX     H
    052.215  106        2741X         MOV     B,M             (BC) = COUNT
    052.216  043        2742X         INX     H
    052.217  136        2743X         MOV     E,M
    052.220  043        2744X         INX     H
    052.221  126        2745X         MOV     D,M             (DE) = FROM
    052.222  043        2746X         INX     H
    052.223  325        2747X         PUSH    D               ((SP)) = FROM
    052.224  136        2748X         MOV     E,M
```

```
052.225  043          2749X          INX     H
052.226  126          2750X          MOV     D,M             (DE) = TO
052.227  043          2751X          INX     H
052.230  343          2752X          XTHL                    ((SP)) = RET, (HL) = FROM
052.231  353          2753X          XCHG                    (DE) = FROM , (HL) = TO
052.232  303 252 030  2754X          JMP     $MOVE           MOVE IT
052.235               2755           XTEXT   MU10


                      2757X **       $MU10 - MULTIPLY UNSIGNED 16 BIT QUANTITY BY 10.
                      2758X *
                      2759X *        (HL) = (DE)*10
                      2760X *
                      2761X *        ENTRY   (DE) = MULTIPLIER
                      2762X *        EXIT    'C' CLEAR IF OK
                      2763X *                 (HL) = PRODUCT
                      2764X *                'C' SET IF ERROR
                      2765X *        USES    D,E,H,L,F
                      2766X
                      2767X
030.324               2768X $MU10    EQU     30324A          IN H17 ROM
052.235               2769           XTEXT   RCHAR


                      2771X **       $RCHAR - READ SINGLE CHARACTER FROM CONSOLE.
                      2772X *
                      2773X *        ENTRY   NONE
                      2774X *        EXIT    (A) = CHARACTER
                      2775X *        USES    A,F
                      2776X
                      2777X
052.235  377 001      2778X $RCHAR   DB      SYSCALL,.SCIN
052.237  332 235 052  2779X          JC      $RCHAR          NOT READY
052.242  311          2780X          RET
                      2781X
052.243  377 002      2782X $WCHAR   DB      SYSCALL,.SCOUT
052.245  311          2783X          RET
052.246               2784           XTEXT   RTL


                      2786X **       $RTL - READ TEXT LINE.
                      2787X *
                      2788X *        $RTL READS A LINE FROM THE TERMINAL.
                      2789X *
                      2790X *        CHARACTER ARE ACCEPTED FROM THE TERMINAL, RUBOUT AND BACKSPACE
                      2791X *        CHARACTERS ARE PROCESSED. WHEN A CARRIAGE RETURN IS ENTERED,
                      2792X *        $RTL RETURNS.
                      2793X *
                      2794X *        ENTRY   (HL) = BUFFER FWA
                      2795X *        EXIT    'C' CLEAR IF OK
```

```
                                2796X *                 DATA IN BUFFER
                                2797X *                 (A) = TEXT LENGTH
                                2798X *                 'C' SET IF CTL-D STRUCK
                                2799X *         USES    A,F
                                2800X
                                2801X
052.246  315 255 052            2802X $RTL.     CALL    $RTL            $RTL IN UPPER CASE
052.251  330                    2803X           RC                      CTL-D
052.252  303 172 052            2804X           JMP     $MLU            MAP LINE TO UPPER CASE
                                2805X
052.255                         2806X $RTL      EQU     *
052.255  345                    2807X           PUSH    H               SAVE FWA
052.256  315 235 052            2808X $RTL1     CALL    $RCHAR
052.261  376 004                2809X           CPI     CTLD
052.263  312 310 052            2810X           JE      $RTL2           CTL-D STRUCK
052.266  167                    2811X           MOV     M,A
052.267  043                    2812X           INX     H
052.270  376 012                2813X           CPI     NL
052.272  302 256 052            2814X           JNE     $RTL1
052.275  053                    2815X           DCX     H
052.276  066 000                2816X           MVI     M,0
052.300  043                    2817X           INX     H
                                2818X
                                2819X *         ALL DONE, COMPUTE LENGTH
                                2820X
052.301  353                    2821X           XCHG                    (DE) = LWA+1
052.302  343                    2822X           XTHL                    (HL) = FWA
052.303  173                    2823X           MOV     A,E
052.304  225                    2824X           SUB     L               (A) = LENGTH
052.305  247                    2825X           ANA     A               CLEAR CARRY
052.306  321                    2826X           POP     D               RESTORE (DE)
052.307  311                    2827X           RET
                                2828X
                                2829X *         CTL-D STRUCK
                                2830X
052.310  341                    2831X $RTL2     POP     H               (HL) = FWA
052.311  067                    2832X           STC
052.312  311                    2833X           RET
052.313                         2834            XTEXT   SAVALL


                                2836X **        $RSTALL - RESTORE ALL REGISTERS.
                                2837X *
                                2838X *         $RSTALL RESTORES ALL THE REGISTERS OFF THE STACK, AND
                                2839X *         RETURNS TO THE PREVIOUS CALLER.
                                2840X *
                                2841X *         ENTRY   (SP) = PSW
                                2842X *                 (SP+2) = BC
                                2843X *                 (SP+4) = DE
                                2844X *                 (SP+6) = HL
                                2845X *                 (SP+8) = RET
                                2846X *         EXIT    TO *RET*, REGISTERS RESTORED
                                2847X *         USES    ALL
                                2848X
                                2849X
```

```
031.047                 2850X $RSTALL EQU     31047A          IN H17 ROM

                        2852X **      $SAVALL - SAVE ALL REGISTERS ON STACK.
                        2853X *
                        2854X *       $SAVALL SAVES ALL THE REGISTERS ON THE STACK.
                        2855X *
                        2856X *       ENTRY   NONE
                        2857X *       EXIT    (SP) = PSW
                        2858X *               (SP+2) = BC
                        2859X *               (SP+4) = DE
                        2860X *               (SP+6) = HL
                        2861X *       USES    H,L
                        2862X
                        2863X
031.054                 2864X $SAVALL EQU     31054A          IN H17 ROM
052.313                 2865          XTEXT   SCU



                        2867X **      SCU - SETUP CONSOLE USART.
                        2868X *
                        2869X *       SCU CONFIGURES THE CONSOLE USART.
                        2870X *
                        2871X *       IF  8250
                        2872X *         THEN  PORT = 372-3Q
                        2873X *         ELSE  PORT = 340-7Q
                        2874X *
                        2875X *
                        2876X *       ENTRY   NONE
                        2877X *       EXIT    NONE
                        2878X *       USES    A,F,(BC),(HL)
                        2879X
                        2880X
052.313  072 343 040   2881X SCU     LDA     S.CDB
052.316  376 001       2882X         CPI     CDB.H84
052.320  312 363 052   2883X         JZ      SCU1                    IF  8250
                        2884X
                        2885X *       PRESET 8251
                        2886X
052.323  076 201       2887X         MVI     A,201Q
052.325  323 373       2888X         OUT     SC.UART+USR             GET USART IN KNOWN STATE
052.327  323 373       2889X         OUT     SC.UART+USR
052.331  323 373       2890X         OUT     SC.UART+USR
052.333  323 373       2891X         OUT     SC.UART+USR
052.335  076 100       2892X         MVI     A,UCI.IR                RESET
052.337  323 373       2893X         OUT     SC.UART+USR
052.341  072 327 040   2894X         LDA     S.CONTY
052.344  346 010       2895X         ANI     CTP.2SB
000.000                 2896X         ERRNZ   CTP.2SB*16+UMI.1B-UMI.2B
052.346  007           2897X         RLC
052.347  007           2898X         RLC
052.350  007           2899X         RLC
052.351  007           2900X         RLC
```

```
052.352  366 116      2901X          ORI     UMI.1B+UMI.LB+UMI.16X
052.354  323 373      2902X          OUT     SC.UART+USR
052.356  076 025      2903X          MVI     A,UCI.ER+UCI.RE+UCI.TE
052.360  323 373      2904X          OUT     SC.UART+USR
052.362  311          2905X          RET
                      2906X
                      2907X *        IS 8250
                      2908X
052.363  333 355      2909X SCU1     IN      SC.ACE+UR.LSR                                  /80.01.GC/
052.365  346 100      2910X          ANI     UC.TSE          CHECK FOR SHIFT EMPTY          /80.01.GC/
052.367  312 363 052  2911X          JZ      SCU1                                           /80.01.GC/
                      2912X
052.372  257          2913X          XRA     A                                              /79.01.GC/
052.373  323 351      2914X          OUT     SC.ACE+UR.IER   TURN OFF ANY INTERRUPTS        /79.01.GC/
052.375  076 020      2915X          MVI     A,UC.LOD                                       /79.01.GC/
052.377  323 354      2916X          OUT     SC.ACE+UR.MCR                                  /79.01.GC/
053.001  052 344 040  2917X          LHLD    S.BAUD
053.004  076 200      2918X          MVI     A,UC.DLA
053.006  323 353      2919X          OUT     SC.ACE+UR.LCR   ACCESS DIVISOR LATCHES
053.010  175          2920X          MOV     A,L
053.011  323 350      2921X          OUT     SC.ACE+UR.DLL   SET LEAST SIGNIFICANT
053.013  174          2922X          MOV     A,H
053.014  346 177      2923X          ANI     177Q            TRIM STOP BITS
053.016  323 351      2924X          OUT     SC.ACE+UR.DLM   SET MOST SIGNIFICANT
053.020  072 327 040  2925X          LDA     S.CONTY
053.023  346 010      2926X          ANI     CTP.2SB
053.025  017          2927X          RRC
000.000               2928X          ERRNZ   CTP.2SB/2-UC.2SB
000.000               2929X          ERRNZ   UC.2SB-4        (A) = UC.2SB IF 2 STOP BITS
053.026  366 003      2930X          ORI     UC.8BW          8 BIT WORDS
053.030  323 353      2931X          OUT     SC.ACE+UR.LCR
053.032  076 156      2932X          MVI     A,AC.DLY                                       /79.01.GC/
053.034  315 053 000  2933X          CALL    .DLY                                           /79.01.GC/
053.037  333 350      2934X          IN      SC.ACE+UR.RBR   GOBBLE ANY TRASH              /79.01.GC/
053.041  333 354      2935X          IN      SC.ACE+UR.MCR                                  /79.01.GC/
053.043  346 357      2936X          ANI     377Q-UC.LOO                                    /79.01.GC/
053.045  323 354      2937X          OUT     SC.ACE+UR.MCR                                  /79.01.GC/
053.047  311          2938X          RET
053.050               2939           XTEXT   SOB


                      2941X **       $SOB - SKIP OVER BLANKS.
                      2942X *
                      2943X *        $SOB IS CALLED TO SKIP AN ARBITRARILY LONG STRING OF BLANKS AND TABS.
                      2944X *
                      2945X *        ENTRY   (HL) = FWA OF (POSSIBLE) BLANK STRING
                      2946X *        EXIT    (HL) = LWA+1 OF BLANK STRING (UNCHANGED IF NO BLANKS)
                      2947X *                (A) = FIRST NON-BLANK, NON-TAB CHARACTER EEN
                      2948X *        USES    A,F,H,L
                      2949X
                      2950X
053.050  053          2951X $SOB     DCX     H               PRE-DECREMENT
053.051  043          2952X $SOB1    INX     H
053.052  176          2953X          MOV     A,M
```

```
053.053  376 040      2954X          CPI     ' '
053.055  312 051 053  2955X          JE      $SOB1           GOT BLANK
053.060  376 011      2956X          CPI     TAB
053.062  312 051 053  2957X          JE      $SOB1           GOT TAB
053.065  311          2958X          RET
053.066               2959           XTEXT   TJMP


                      2961X **       $TJMP - TABLE JUMP.
                      2962X *
                      2963X *        USAGE
                      2964X *
                      2965X *        CALL    $TJMP           (A) = INDEX
                      2966X *        DW      ADDR1
                      2967X *        .         .
                      2968X *        .         .
                      2969X *        .         .
                      2970X *        DW      ADDRN
                      2971X *
                      2972X *        ENTRY   (A) = INDEX
                      2973X *        EXIT    TO PROCESSOR
                      2974X *                (A) = INDEX*2
                      2975X *        USES    NONE.
                      2976X
                      2977X
031.061               2978X $TJMP    EQU     31061A          IN H17 ROM, (A) = INDEX*2
                      2979X
031.062               2980X $TJMP.   EQU     31062A          IN H17 ROM
053.066               2981           XTEXT   TYPCC


                      2983X **       $TYPCC - TYPE A CHARACTER STRING BY COUNT.
                      2984X *
                      2985X *        $TYPCC TYPES A STRING OF CHARACTERS. THE CALLER SUPPLIES
                      2986X *        THE CHARACTER ADDRESS AND COUNT.
                      2987X *
                      2988X *        ENTRY   (HL) = ADDRESS
                      2989X *                (A) = COUNT
                      2990X *        EXIT    (HL) = LAST CHARACTER ADDRESS+1
                      2991X *        USES    A,F,H,L
                      2992X
                      2993X
053.066               2994X $TYPCC   EQU     *
053.066  247          2995X          ANA     A
053.067  310          2996X          RZ                      NOTHING TO TYPE
053.070  365          2997X          PUSH    PSW             SAVE COUNT
053.071  176          2998X          MOV     A,M             (A) = CHARACTER
053.072  043          2999X          INX     H
053.073  377 002      3000X          DB      SYSCALL,.SCOUT
053.075  361          3001X          POP     PSW
053.076  075          3002X          DCR     A
053.077  303 066 053  3003X          JMP     $TYPCC
```

```
053.102                3004          XTEXT   TYPT2


                       3006X **      $TYPTX - TYPE TEXT.
                       3007X *
                       3008X *       $TYPTX IS CALLED TO TYPE A BLOCK OF TEXT ON THE SYSTEM CONSOLE.
                       3009X *
                       3010X *       IMBEDDED ZERO BYTES INDICATE A CARRIAGE RETURN LINE FEED,
                       3011X *       A BYTE WITH THE 200Q BIT SET IS THE LAST BYTE IN THE MESSAGE.
                       3012X *
                       3013X *       ENTRY   (RET) = TEXT
                       3014X *       EXIT    TO (RET+LENGTH)
                       3015X *       USES    A,F
                       3016X
                       3017X
031.136                3018X $TYPTX  EQU     31136A          IN H17 ROM
                       3019X
031.144                3020X $TYPTX. EQU     31144A          IN H17 ROM
053.102                3021          XTEXT   UDD


                       3023X **      $UDD - UNPACK DECIMAL DIGITS.
                       3024X *
                       3025X *       UDD CONVERTS A 16 BIT VALUE INTO A SPECIFIED NUMBER OF
                       3026X *       DECIMAL DIGITS. THE RESULT IS ZERO FILLED.
                       3027X *
                       3028X *       ENTRY   (B,C) = ADDRESS VALUE
                       3029X *               (A) = DIGIT COUNT
                       3030X *               (H,L) = MEMORY ADDRESS
                       3031X *       EXIT    (HL) = (HL) + (A)
                       3032X *       USES    ALL
                       3033X
                       3034X
031.157                3035X $UDD    EQU     31157A          IN H17 ROM
053.102                3036          XTEXT   UDDN


                       3038X **      $UDDN - UNPACK DECIMAL DIGITS.
                       3039X *
                       3040X *       UDDN CONVERTS A 16 BIT VALUE INTO A SPECIFIED NUMBER OF
                       3041X *       DECIMAL DIGITS. THE RESULT IS NULL FILLED TO THE LEFT.
                       3042X *
                       3043X *       ENTRY   (B,C) = ADDRESS VALUE
                       3044X *               (A) = DIGIT COUNT
                       3045X *               (H,L) = MEMORY ADDRESS
                       3046X *       EXIT    (HL) = (HL) + (A)
                       3047X *       USES    ALL
                       3048X
                       3049X
053.102                3050X $UDDN   EQU     *
```

```
053.102  315 072 030  3051X          CALL    $DADA
053.105  345          3052X          PUSH    H               SAVE FINAL (H,L) VALUE
                       3053X
053.106  365          3054X UDDN1    PUSH    PSW
053.107  345          3055X          PUSH    H
053.110  021 012 000  3056X          LXI     D,10
053.113  315 106 030  3057X          CALL    $DD66           (H,L) = VALUE/10
053.116  104          3058X          MOV     B,H
053.117  115          3059X          MOV     C,L             (BC) = QUOTIENT
053.120  341          3060X          POP     H
053.121  076 060      3061X          MVI     A,'0'
053.123  203          3062X          ADD     E               ADD REMAINDER
053.124  053          3063X          DCX     H
053.125  167          3064X          MOV     M,A             STORE DIGIT
053.126  170          3065X          MOV     A,B
053.127  261          3066X          ORA     C
053.130  312 142 053  3067X          JZ      UDDN2           ALL ZEROS
053.133  361          3068X          POP     PSW
053.134  075          3069X          DCR     A
053.135  302 106 053  3070X          JNZ     UDDN1           IF MORE TO GO
                       3071X
                       3072X *        ALL DONE, EXIT
                       3073X
053.140  341          3074X UDDN1.5  POP     H               RESTORE H
053.141  311          3075X          RET                       RETURN
                       3076X
                       3077X *        DIGITS LEADING THIS ONE ARE ZERO, STORE NULLS INSTEAD.
                       3078X
053.142  361          3079X UDDN2    POP     PSW
053.143  075          3080X UDDN3    DCR     A
053.144  312 140 053  3081X          JE      UDDN1.5         ALL DONE
053.147  053          3082X          DCX     H
053.150  066 000      3083X          MVI     M,0
053.152  303 143 053  3084X          JMP     UDDN3
053.155               3085           XTEXT   XCHGBC
```

```
                       3087X **       XCHGBC  -  XCHG BC
                       3088X *
                       3089X *        EXCHANGE THE 'BC' REGISTER PAIR WITH THE 'HL' REGISTER PAIR.
                       3090X *
                       3091X *        ENTRY:  BC      = ORIGINAL BC
                       3092X *                HL      = ORIGINAL HL
                       3093X *
                       3094X *        EXIT:   BC      = ORIGINAL HL
                       3095X *                HL      = ORIGINAL BC
                       3096X *
                       3097X *        USES:   BC,HL
                       3098X *
                       3099X
053.155  365          3100X XCHGBC   PUSH    PSW
053.156  170          3101X          MOV     A,B
053.157  104          3102X          MOV     B,H
053.160  147          3103X          MOV     H,A
```

```
053.161  171  3104X  MOV  A,C
053.162  115  3105X  MOV  C,L
053.163  157  3106X  MOV  L,A
053.164  361  3107X  POP  PSW
053.165  311  3108X  RET
```

```
                        3111  **      BUFFERS
                        3112
      053.186           3113  MEML    EQU     *               LOAD IMAGE LWA
                        3114
      053.186           3115          DS      128             PATCH AREA
                        3116
      053.386           3117          DS      1               SET TO 200Q FOR PROCESSING 'VERB'
      053.387           3118  VERB    DS      120             VERB BUFFER
                        3119
      054.157           3120  LINE    DS      120             LINE BUFFER
                        3121
      054.347           3122  LABEL   DS      256             LABEL BUFFER
                        3123
      055.347           3124  RMEML   EQU     *               RUNNING LIMIT
                        3125
      055.347           3126          END
ASSEMBLY COMPLETE
 3126 STATEMENTS
    0 ERRORS DETECTED
10098 BYTES FREE
```

```
$CAD    050130   1555    1997L
$CCO    050340   1080    2096L
$CDEHL  030216    885    2084E  2341
$COMP   030060   2027    2121E
$CPF    050355   1297    2143L
$CPF1   050357   2144L   2157
$CPF2   051014   2146    2149   2151   2153   2163L
$CRLF   051017   1585    2176L  2488
$DAD    051025   1565    2204L
$DADA   030072   2268E   3051
$DADA.  030101   1630    2036   2232   2278E
$DDD    051212   1997    2055   2296L
$DDD1   051215   2298L   2315
$DTB    051247   2328L
$DTB1   051253   2332L   2335
$DTB2   051262   2340L   2345
$DTB3   051277   2342    2349L
$DU66   030106   2208    2222   2369E  3057
$FST    051332   1153    2412L
$FST.C  052034   2426    2454   2470L
$FST.L  052032   2421    2469L
$GNL    052035   1081    2483L
$INDL   030234   2525E
$INDLB  052056   2540L
$INDS   052077   2573L
$INDSB  052133   2603L
$MCU    052161   2656L   2678
$MLU    052172   2673L   2804
$MLU1   052175   2676L   2681
$MOVE   030252   1341    1363   1383   1414   2235   2708E  2754
$MOVEL  052212   1318    1335   2738L
$MU10   030324   1628    2306   2768E
$RCHAR  052235   2778L   2779   2808
$RSTALL 031047   2100    2850E
$RTL    052255   2802    2806E
$RTL.   052246   1111    2802L
$RTL1   052256   2808L   2814
$RTL2   052310   2810    2831L
$SAVALL 031054   2096    2864E
$SOB    053050   1294    1475   1548   2951L
$SOB1   053051   2952L   2955   2957
$TJMP   031061   1156    2978E
$TJMP.  031062   2980E
$TYPCC  053066   1574    1591   2994E  3003
$TYPTX  031136   1067    1071   1108   1282   1504   1557   1936   3018E
$TYPTX. 031144   3020E
$UDD    031157   2216    2246   3035E
$UDDN   053102   1597    1603   1609   1620   1635   3050E
$WCHAR  052243   2782L
.       044005   1240S
.ABUSS  040024    269E
.ALARM  002136    242E
.ALEDS  040013    267E
.CHFLG  000060    494L
.CLEAR  000055    491L    875
.CLEARA 000056    492L    914
.CLOSE  000046    484L
.CLRCO  000007    468L    1060   1281
```

.CONSL	000006	467L	2099	2485		
.CRC	002347	250E				
.CRCSUM	040027	270E				
.CTC	002172	244E				
.CTLC	000041	479L	1076			
.CTLFLG	040011	266E				
.DECODE	000053	489L				
.DELET	000050	486L				
.DISMT	000061	495L				
.DLEDS	040021	268E				
.DLY	000053	239E	2933			
.DMNMS	000203	506L				
.DMOUN	000201	504L	1444	1455	1766	
.DOD	003122	253E				
.DODA	003356	255E				
.DSPMOD	040007	264E				
.DSPROT	040006	263E				
.DUMP	001374	241E				
.ERROR	000057	493L	897	1063	1571	1588
.EXIT	000000	461L	1762			
.HORN	002140	243E				
.IDENT	000000	238E				
.IOWRK	040002	261E				
.LINK	000040	478L	1232	1303	1932	
.LOAD	001267	240E				
.LOADD	000062	496L	1526			
.LOADO	000010	469L	1870	1873		
.MANUF.	000001	3E	1082			
.MFLAG	040010	265E				
.MONMS	000202	505L				
.MOUNT	000200	503L	1430			
.NAME	000054	490L				
.OPENC	000045	483L				
.OPENR	000042	480L				
.OPENU	000044	482L				
.OPENW	000043	481L				
.PCHL	002264	246E				
.POSIT	000047	485L				
.PRINT	000003	464L	1622	1637		
.RCK	003260	254E				
.READ	000004	465L				
.REGI	040005	262E				
.REGPTR	040035	273E				
.RENAM	000051	487L				
.RESET	000204	507L	1478			
.RNB	002331	249E				
.RNP	002325	248E				
.SCIN	000001	462L	2778			
.SCOUT	000002	463L	2177	2782	3000	
.SETTP	000052	488L	868			
.SRS	002265	247E				
.START	040000	260E				
.SYSRES	000012	471L				
.TICCNT	040033	272E				
.TPERR	002205	245E				
.TPERRX	040031	271E				
.UIVEC	040037	274E				
.VERS	000011	470L	863	1489		

```
.WNB     003024    252E
.WNP     003017    251E
.WRITE   000005    466L
ABS.COD  000010    828L     849
ABS.ENT  000006    826L
ABS.ID   000000    822L
ABS.LDA  000002    824L
ABS.LEN  000004    825L
AC.DLY   000156     86E    2932
AIO.CGN  041047    757L
AIO.CHA  041116    772L
AIO.CNT  041111    768L
AIO.CSI  041050    758L
AIO.DDA  041041    753E
AIO.DES  041055    762L
AIO.DEV  041057    763L
AIO.DIR  041062    766L
AIO.DTA  041053    761L    1530
AIO.EOF  041113    770L
AIO.EOM  041112    769L
AIO.FLG  041043    754L
AIO.GRT  041044    755L
AIO.LGN  041051    759L
AIO.LSI  041052    760L
AIO.SPG  041046    756L
AIO.TFF  041114    771L
AIO.UNI  041061    764L
AIO.VEC  041040    752L
BELL     000007     61E    1062   1068   1072   1558   1937
BITS     050111   1957L
BITS1    050116   1962L    1964
BKSP     000010     63E
BOOT.P   000001    732E
BYE      047213   1106     1198   1751L
BYE.     047244   1755     1757   1759   1766L
BYEA     047256   1758     1773L
BYEB     047263   1756     1774L
BYEC     047270   1754     1775L
C.STX    000002     65E
C.SYN    000026     64E
CAD1     050171   2024L    2040
CAD2     050225   2022     2044L  2053   2056   2059   2062   2064
CAD3     050230   2029     2050L
CADA     050273   2023     2071L
CB.CLI   000100    208E     223
CB.MTL   000040    207E
CB.SPK   000200    209E
CB.SSI   000020    206E
CCHIT    044302   1074     1281L
CCT      042330    876      914L
CDB.H84  000001    675E    2380   2882
CDB.H85  000000    674E
CDT      042333    880      929L
CDT1     042336    930L     980
CDT2     042370    942      956L
CDT3     043011    967      977L
CO.FLG   000001    652E    2098
COPY     045153   1195     1397E
```

COT	043022	881	993L				
CR	000015	57E					
CS.FLG	000200	653E					
CSA	047275	1470	1827L				
CSL.CHR	000001	630E					
CSL.ECH	000200	628E					
CSL.WRP	000002	629E					
CTLA	000001	72E					
CTLB	000002	73E					
CTLC	000003	74E	1075				
CTLD	000004	75E	2809				
CTLO	000017	76E					
CTLP	000020	77E					
CTLQ	000021	78E					
CTLS	000023	79E					
CTLZ	000032	80E					
CTP.2SB	000010	638E	2895	2896	2926	2928	
CTP.BKM	000002	639E					
CTP.BKS	000200	635E					
CTP.MLI	000040	636E					
CTP.MLO	000020	637E					
CTP.TAB	000001	640E					
D.CON	040110	522L	541				
D.DLYHS	040244	575L					
D.DLYMO	040243	574L					
D.DRVTB	040251	580L					
D.DVCTL	040242	572L					
D.E.CHK	040267	591L					
D.E.HCK	040270	592L					
D.E.HSY	040266	590L					
D.E.MDS	040265	589L					
D.E.TRK	040272	594L					
D.E.VOL	040271	593L					
D.ERR	040265	588L					
D.ERRL	040273	595L					
D.ERTS	040126	557L	1623	1629	1639		
D.HECNT	040261	582L	1604				
D.LPSA	040116	548L					
D.MAIA	040115	547L					
D.OECNT	040264	584L					
D.OPR	040273	599L	1592				
D.OPW	040275	600L	1598				
D.RAM	040240	525L	567	602			
D.RAML	000037	602E					
D.SDPA	040117	549L					
D.SDPB	040120	550L					
D.SECNT	040262	583L	1610				
D.STSA	040121	551L					
D.STSB	040122	552L					
D.TRKPT	040245	577L					
D.TS	040241	570L					
D.TT	040240	569L					
D.VEC	040130	524L					
D.VOLPT	040247	578L					
D.WHDA	040123	553L					
D.WNHA	040124	554L					
D.WRITA	040112	544L					
D.WRITB	040113	545L					

```
D.WRITC 040114      546L
D.WSCA  040125      555L
D.XITA  040110      543L
DAD1    051143     2214     2227    2229    2252L
DADB    051146     2231     2256L
DATE    046040     1186     1546E
DATE2   046112     1556     1561L
DATE3   046125     1551     1559    1569L
DC.ABT  000007      442L     972
DC.CLO  000006      441L
DC.LOD  000011      444L
DC.MAX  000012      445L
DC.MOU  000010      443L
DC.OPR  000003      438L
DC.OPU  000005      440L
DC.OPW  000004      439L
DC.REA  000000      435L
DC.RER  000002      437L
DC.WRI  000001      436L
DDF.BOL 000011      353E
DDF.BOO 000000      352L
DDF.LAB 000011      354L
DDF.RBT 000012      355L
DDF.USR 000014      356L
DEBUG   000001       2E     1209    1266    1776
DELA    045111     1362     1367L   1368
DELAL   000010     1361     1368E
DELETE  045067     1177     1357E
DEV.DDA 000004      790L     948     955     958     963
DEV.DVG 000016      802L
DEV.DVL 000014      801L
DEV.FLG 000006      791L     963
DEV.JMP 000003      789L
DEV.MNU 000011      798L
DEV.MUN 000010      797L
DEV.NAM 000000      781L
DEV.RES 000002      785L     939     948     955     958    1531
DEV.SPG 000007      796L
DEV.UNT 000012      799L
DEVELEN 000017      804E     978
DF.CLR  000376      365E
DF.EMP  000377      364E
DIF.CNT 000020       51E
DIF.LOC 000100       49E
DIF.SYS 000200       48E
DIF.WP  000040       50E
DIR     045157     1189     1408E
DIR.ALD 000025      380L
DIR.CLU 000015      373L
DIR.CRD 000023      379L
DIR.EXT 000010      368L
DIR.FGN 000020      376L
DIR.FLG 000016      374L
DIR.LGN 000021      377L
DIR.LSI 000022      378L
DIR.NAM 000000      367L
DIR.PRO 000013      369L
DIR.VER 000014      370L
```

```
DIRA     045201      1413      1418L   1419
DIRAL    000006      1412      1419E
DIRELEN  000027       382E      410     766
DIRIDL   000015       371E
DIS.ENL  001373       414L
DIS.ENT  000000       409E
DIS.LNK  001376       416L
DIS.SEC  001374       415L
DM.MR    000000       213E
DM.MW    000001       214E
DM.RR    000002       215E
DM.RW    000003       216E
DMO1     045254      1448      1452L
DMOUNT   045233      1168      1441E
DR.IM    000001       786E      944
DR.PR    000002       787E      941    1534
DT.CR    000002       793E
DT.CW    000004       794E
DT.DD    000001       792E      965
DV.EL    000000       782E
DV.NU    000001       783E
EC.CNA   000004       284L
EC.DDA   000027       303L
EC.DIF   000017       295L
EC.DIW   000035       309L
EC.DNI   000045       317L
EC.DNR   000046       318L
EC.DNS   000005       285L
EC.DSC   000047       319L
EC.EOF   000001       281L
EC.EOM   000002       282L
EC.FAO   000031       305L
EC.FAP   000026       302L
EC.FL    000030       304L
EC.FNF   000014       292L
EC.FNO   000011       289L
EC.FNR   000034       308L
EC.FOD   000043       315L
EC.FUC   000013       291L
EC.ICN   000016       294L
EC.IDN   000006       286L
EC.IFC   000020       296L
EC.IFN   000007       287L
EC.ILC   000003       283L
EC.ILO   000040       312L
EC.ILR   000012       290L
EC.ILV   000037       311L
EC.IOI   000052       322L
EC.IS    000032       306L
EC.NCV   000050       320L      894
EC.NEM   000021       297L
EC.NOS   000051       321L
EC.NPM   000044       316L     1447
EC.NRD   000010       288L
EC.NVM   000042       314L     1768
EC.OTL   000053       323L
EC.RF    000022       298L
EC.UNA   000036       310L
```

```
EC.UND  000015   293L
EC.UUN  000033   307L
EC.VPM  000041   313L
EC.WF   000023   299L
EC.WP   000025   301L
EC.WPV  000024   300L
ECI     051310  1051    2379L
ECI1    051325  2381    2391L
ENL     000212    70E   1509  1558  1732  1734  1937
ERROR   043113  1059L   1304  1431  1450  1456  1479  1527  1771  1871  1874
ESC     000033    68E
FB.CHA  000000    35L
FB.FLG  000001    36L
FB.FWA  000002    37L
FB.LIM  000006    39L
FB.LWA  000010    40L
FB.NAM  000012    41L     42
FB.NAML 000021    42E   2143
FB.PTR  000004    38L
FBENL   000033    43E
FEC     047313  1338    1360  1380  1411  1847L 1899
FEC1    047315  1849L   1853
FF      000014    71E
FST1    051352  2428L   2460
FST2    051365  2435    2441L 2448
FST3    052003  2443    2450L 2453
FST4    052025  2445    2462L
FT.ABS  000000   424E     850
FT.BAC  000003   427E
FT.PIC  000001   425E
FT.REL  000002   426E
HELP    044355  1171    1317E
HELPA   044375  1319    1324L 1325
HELPAL  000025  1319    1325E
HOS.SFG 000002   349E
I.BYE   000014  1197E    1262
I.CONFL 000004   655E     656  2097
I.CONTY 000001   642E     643
I.CONWI 000003   648E     649
I.COP   000013  1194E    1261
I.CSLMD 000000   632E
I.CUSOR 000002   645E     646  2483
I.DAT   000010  1185E    1254
I.DEL   000005  1176E    1251
I.DIR   000011  1188E    1255  1256  1257  1258
I.DMO   000002  1167E    1247
I.HEL   000003  1170E    1248
I.LIS   000004  1173E    1249  1250
I.LOA   000017  1206E    1265
I.MOU   000007  1182E    1253
I.REN   000006  1179E    1252
I.RES   000015  1200E    1263
I.RUN   000000  1161E    1246
I.STA   000012  1191E    1259  1260
I.SYS   000001  1164E
I.VER   000016  1203E    1264
ILDEHL  052051  2504L    2576
ILLCMD  043166  1071L    1233
```

```
ILLSYN  043127  1067L  1146  1477
IP.PAD  000360   199E
ISDEHL  052154  2583   2639L
LAB.DAT 000006   394E
LAB.DIS 000003   390L
LAB.GRT 000005   391L
LAB.IND 000001   389L
LAB.LAB 000021   401L   402
LAB.LBL 000074   402E
LAB.NOD 000002   396E
LAB.SER 000000   388L
LAB.SPG 000007   392L
LAB.SYS 000001   395E
LAB.VER 000011   399L
LAB.VLT 000010   398L
LABEL   054347  3122L
LF      000012    58E
LINE    054157  1110   1118  1230  1296  1302  1319  1320  1336  1342  3120L
LIST    045022  1174   1322  1334E
LISTA   045056  1336   1345L 1346
LISTAL  000004  1336   1346E
LISTB   045062  1340   1347L 1348
LISTBL  000005  1339   1348E
LOADD   046006  1207   1522E
LOADOV  047324  1453   1473  1753  1869L
M.CDCA  000017   845L
M.CDLY  000016   844L
M.CFWA  000012   842L   1036
M.CIN   000006   840L   1030  1045
M.CINT  000005   839L   1027  1030  1036
M.CLWA  000014   843L
M.COUT  000010   841L   1045
M.CPRE  000003   837L   1021  1024
M.CRUB  000004   838L   1024  1027
M.CSLC  000002   836L   1018  1021
M.FOX   000303   233E
M.FAM8  000021   232E
M.SALO  000001   835L   1101  1829
M.SYSM  000000   834L
MEML    053166   852    3113E
MOUNT   045207  1183   1428E
MOUNTA  045220  1434L
NL      000012    69E    70   896  1068  1072  1727  1730  1732  1734  1740  1937  2176
                2813
NUL2    000000    60E
NULL    000200    59E
OP.CTL  000360   200E
OP.DIG  000360   201E
OP.SEG  000361   202E
OVL.COD 000000   330L
OVL.ENS 000010   335E   998
OVL.ENT 000004   332L
OVL.FLB 000006   333L   995
OVL.IN  000001   699E
OVL.NUM 000014   701E
OVL.RES 000002   700E
OVL.SIZ 000002   331L
OVL.UCS 000200   702E
```

```
OVL0     000000    341L    1869
OVL1     000001    342L    1872
PCHL     043021     973     982L
PCL      047343    1229    1300   1895L  1930
PCL1     047363    1906L   1912
PCL2     047374    1902    1914L
PCLA     050000    1897    1918E
PIP      050002    1343    1365   1385   1399   1416   1930L
PIPA     050075    1931    1940L
PRSCL    043041     873    1013E
QUOTE    000047      66E
RENA     045143    1382    1387L  1388
RENAL    000010    1381    1388E
RENAME   045121    1180    1377E
RESET    045271    1201    1469E
RMEML    055347     867    3124E
ROMBOOT  030000     517E    899
RUBOUT   000177      62E
RUN      044314    1162    1293L
RUNA     044347    1301    1306L
S.BAUD   040344     676L    2917
S.BOOTF  041034     731L
S.CAADR  040333     659L
S.CACC   041006     715L
S.CCTAB  040335     660L
S.CDB    040343     673L    2379   2881
S.CFWA   040352     683L
S.CODE   041007     716L
S.CONFL  040332     657L
S.CONTY  040327     644L    2894   2925
S.CONWI  040331     650L
S.CSLMD  040326     633L     643    646    649    656   1079
S.CUSOR  040330     647L
S.DATC   040310     615L    1562
S.DATE   040277     614L    1564   1572   1589
S.DCS    041033     729L
S.DDDTA  040366     694L
S.DDGRP  040364     691L
S.DDLDA  040360     689L
S.DDLEN  040362     690L
S.DDOPC  040370     695L
S.DFWA   040354     684L     929
S.DIREA  041016     723L
S.DLINK  040346     681L    1015   1102   1830
S.FASER  041013     722L     898
S.FCI    041021     724L
S.GRT0   024000     513E
S.GRT1   025000     514E
S.GRT2   026000     515E
S.GUP    041027     726L
S.HIMEM  040316     617L
S.INT    040343     527L     669
S.JUMPS  041010     720L
S.MOUNT  041032     728L     877   1098
S.OFWA   040350     682L     993
S.OMAX   040324     623L
S.OSN    041004     711L
S.OVLE   041000     708L
```

```
S.OVLFL 040371      704L      888
S.OVLS  040376      707L
S.OVSTK 041035      736L
S.RFWA  040356      685L      884     1529
S.SCI   041024      725L
S.SCR   041120      774L
S.SDD   041010      721L      949      951
S.SOVR  041146      529L      531
S.SSN   041002      710L
S.SYSN  040320      619L      882      889     1528
S.TIME  040312      616L
S.UCSF  040372      705L
S.UCSL  040374      706L
S.USRN  040322      621L
S.VAL   040277      526L      612
SC.ACE  000350       85E     2392     2909     2914     2916     2919     2921     2924     2931     2934     2935     2937
SC.UART 000372      154E     2386     2888     2889     2890     2891     2893     2902     2904
SCU     052313     1050     2881L
SCU1    052363     2883     2909L    2911
STACK   042200      533E      870     1077     1898
STACKL  001032      531E
STATO   046332     1625     1643E
STATA   046335     1621     1727L
STATB   046350     1595     1728L
STATC   046365     1601     1729L
STATD   047025     1608     1731L
STATE   047046     1619     1732L
STATF   047077     1636     1733L
STATG   047143     1633     1734L
STATH   047150     1736L
STATI   047157     1737L
STATJ   047177     1739L     1741
STATK   047201     1740L
STATL   000005     1741E
STATUS  046145     1192     1584E
SYDD    040130      523E
SYSC0   043262     1100     1108L
SYSC1   043312     1120L     1137
SYSC2   043344     1122     1126     1134L
SYSC3   043352     1124     1128     1130     1141L
SYSC5   044045     1154     1228L    1471
SYSCA   044005     1159L    1161     1164     1167     1170     1173     1176     1179     1182     1185     1188     1191
                   1194     1197     1200     1203     1206     1240
SYSCALL 000377      454E      863      868      875      897      914     1060     1063     1076     1232     1281     1303
                   1430     1444     1455     1478     1489     1526     1571     1588     1622     1637     1762     1766     1870
                   1873     1932     2099     2177     2485     2778     2782     3000
SYSCB   044064     1231     1235L
SYSCC   044072     1152     1242L    1243     1273
SYSCCL  000160     1243     1273E
SYSCMD  043213      879      886      890     1069     1074L    1112     1165     1284     1432     1446     1457     1480
                   1510     1536     1575     1723     1938
SYSX1   042312      864      866      894L
SYSXIT  042200      860E     1065
SYSXITA 042327      862      871      901L
TAB     000011       67E     1505     1587     2956
UC.2SB  000004      111E     2928     2929
UC.5BW  000000      107E
UC.6BW  000001      108E
```

```
UC.7BW  000002    109E
UC.8BW  000003    110E    2930
UC.BI   000020    130E
UC.CTS  000020    139E
UC.DCS  000001    135E
UC.DDR  000002    136E
UC.DLA  000200    116E    2918
UC.DR   000001    126E
UC.DRL  000010    138E
UC.DSR  000040    140E
UC.DTR  000001    119E
UC.EDA  000001     97E    2391
UC.EPS  000020    113E
UC.FE   000010    129E
UC.IID  000006    104E
UC.IIP  000001    103E
UC.LOO  000020    123E    2915    2936
UC.MSI  000010    100E
UC.OR   000002    127E
UC.OU1  000004    121E
UC.OU2  000010    122E
UC.PE   000004    128E
UC.PEN  000010    112E
UC.RI   000100    141E
UC.RLS  000200    142E
UC.RSI  000004     99E
UC.RTS  000002    120E
UC.SB   000100    115E
UC.SKP  000040    114E
UC.TER  000004    137E
UC.THE  000040    131E
UC.TRE  000002     98E
UC.TSE  000100    132E    2910
UCI.ER  000020    176E    2385    2903
UCI.IE  000002    178E    2385
UCI.IR  000100    174E    2892
UCI.RE  000004    177E    2385    2903
UCI.RO  000040    175E
UCI.TE  000001    179E    2385    2903
UDDN1   053106   3054L    3070
UDDN1.5 053140   3074L    3081
UDDN2   053142    3067    3079L
UDDN3   053143   3080L    3084
UDR     000000    151E
UMI.16X 000002    169E    2901
UMI.1B  000100    159E    2896    2901
UMI.1X  000001    168E
UMI.2B  000300    161E    2896
UMI.64X 000003    170E
UMI.HB  000200    160E
UMI.L5  000000    164E
UMI.L6  000004    165E
UMI.L7  000010    166E
UMI.L8  000014    167E    2901
UMI.PA  000020    163E
UMI.PE  000040    162E
UNT.DIS 000005    813L
UNT.FLG 000000    810L
```

```
UNT.GRT 000001    811L
UNT.GTS 000003    812L
UNT.SIZ 000007    815E
UO.CLK  000001    225E
UO.DDU  000002    224E
UO.HLT  000200    222E
UO.NFR  000100    223E
UR.DLL  000000     92E   2921
UR.DLM  000001     94E   2924
UR.IER  000001     96E   2392  2914
UR.IIR  000002    102E
UR.LCR  000003    106E   2919  2931
UR.LSR  000005    125E   2909
UR.MCR  000004    118E   2916  2935  2937
UR.MSR  000006    134E
UR.RBR  000000     88E   2934
UR.THR  000000     90E
USERFWA 042200    534E    849   851   852   853
USR     000001    152E   2386  2888  2889  2890  2891  2893  2902  2904
USR.FE  000040    183E
USR.OE  000020    184E
USR.PE  000010    185E
USR.RXR 000002    187E
USR.TXE 000004    186E
USR.TXR 000001    188E
VERB    053367   1117   1119  1151  3118L
VERS    000026    452E    865
VERS1   045331   1490   1492L
VERSA   045377   1499   1506L
VERSB   046001   1503   1508L
VERSN   045322   1204   1488E
XCHGBC  053155   2573   2577  2585  2587  3100L

20844 BYTES FREE
```

149Stop
150File Already Exists
151Illegal Format for File Name
152Too Many or Too Few Arguments Specified
153File is not Open
154Warning--Line Length Too Long, excess ignored.
155Channel is Currently in Use
000HEATH HDOS Issue #50.x5.00 07-Dec-79

```
128CTL-C Struck
129CTL-B Struck
130Data Exhausted
131Attempted Divide by Zero
132Illegal Number Value
133Illegal Usage
134Data Lock Engaged
135Cant Find Variable Mentioned in NEXT Statement
136Floating Point Overflow (Number too Large)
137No Corresponding GOSUB for this RETURN statement
138String Length Exceeds 256 Characters
139Illegal or Unknown Statement Number
140Syntax Error
141Type Conflict (Illegal Mix of Strins and Number Values)
142Out of RAM Space
143Subscript Out of Range
144Too Many or Too Few Subscripts Supplied
145This Array Was Not Dimensioned
146An Illegal Character Was Encountered
147Undefined Function
148End
001?02 End of File
002?02 No Free Space on Media
003?02 Illegal 'SYSCALL' Function Code
004?02 Channel is Already in Use
005?02 Device is Not Capable of This Operation
006?02 Illegal Format for Device Name
007?02 Illegal Format for File Name
008?02 Not Enough Memory for the Device Driver
009?02 Channel is not Open
010?02 Illegal Function Request
011?02 File Usage Conflicts
012?02 File Cannot Be Located
013?02 Unknown Device Name
014?02 Illegal Channel Number
015?02 The Volume Directory is Full
016?02 The File's Contents are not Correct for This Operation
017?02 Not Enough RAM for this Program
018?02 Read Failure on the Device
019?02 Write Failure on the Device
020?02 Attempted Write Protection Violation
021?02 Disk is Write Protected
022?02 The File is Already Present
023?02 Aborted by Device Driver
024?02 File is Locked Against Flag Change
025?02 A File is Already Open
026?02 Illegal or Unknown Switch Specified
027?02 Unknown Unit for this Device
028?02 Non-null File Name is Required
029?02 Device is incapable of write operations (or is write locked)
030?02 Unit Not Available
031?02 Illegal Value
032?02 Illegal Option
033?02 Volume Presently Mounted on the Device
034?02 No Volume Presently Mounted on the Device
035?02 File Open on the Device
036?02 No Provisions Made for Remounting More Disks
037?02 This Disk must be Initialized before it can be Mounted
038?02 Unable to Read this Disk, It Probably has not been Properly Initialized
039?02 Disk Structure is Corrupt.  Contact Technical Correspondence for Help.
040?02 Not the Correct Version of HDOS for this Program.
041?02 No Operating System Mounted, Required for this Operation.
042?02 Illegal Overlay Index.
043?02 Overlay Too Large.
```