```
000.001          PUBLIC  EQU     1                       NOT PUBLIC
                 ***     INITIALIZE - INITIALIZE DISK.
                 *
                 *       J.G. LETWIN, 10/77
                 *
                 *       COPYRIGHT 10/77, HEATH CO.
                 *       COPYRIGHT 79/04, HEATH CO.
                 *
                 *       G. Chandler,     78/10          Maintenence release
                 *                        79/04          Renamed INIT from INIT17 for H89
                 *                                       compatibility reasons.
                 *


                 ***     INITIALIZE INITIALIZES A NEW DISK VOLUME.
                 *
                 *       ACTIONS TAKEN:
                 *
                 *       1) REQUEST MEDIA INSERTION
                 *       2) CHECK FOR PROPER HOLE PATTERN
                 *       3) READ LABEL AND TYPE INFORMATION
                 *       4) ASK FOR # OF TRACKS
                 *       5) ASK FOR DISK SERIAL NUMBER AND LABEL
                 *       6) ZERO AND INITIALIZE TRACKS
                 *       7) ASK FOR LIST OF BAD SECTORS
                 *       8) FORMAT DEVICE
                 *       9) PROMPT RESTORATION OF SYSTEM DEVICE


000.303          MI.JMP  EQU     303Q

000.002          SPG     EQU     2               2 SECTORS PER GROUP
001.220          VOLSIZ  EQU     400             400 SECTORS PER VOLUME

000.000                  XTEXT   MTR
```

```
                            39X **      MTR - PAM/8 EQUIVALENCES.
                            40X *
                            41X *       THIS DECK CONTAINS SYMBOLIC DEFINITIONS USED TO
                            42X *       MAKE USE OF THE PAM/8 CODE AND CONTROL BYTES.


                            44X **      IO PORTS
                            45X
     000.360                46X IP.PAD  EQU     360Q            PAD INPUT PORT
     000.360                47X OP.CTL  EQU     360Q            CONTROL OUTPUT PORT
     000.360                48X OP.DIG  EQU     360Q            DIGIT SELECT OUTPUT PORT
     000.361                49X OP.SEG  EQU     361Q            SEGMENT SELECT OUTPUT PORT


                            51X **      FRONT PANEL CONTROL BITS.
                            52X
     000.020                53X CB.SSI  EQU     00010000B       SINGLE STEP INTERRUPT
     000.040                54X CB.MTL  EQU     00100000B       MONITOR LIGHT
     000.100                55X CB.CLI  EQU     01000000B       CLOCK INTERRUPT ENABLE
     000.200                56X CB.SPK  EQU     10000000B       SPEAKER ENABLE


                            58X **      MONITOR MODE FLAGS.
                            59X
     000.000                60X DM.MR   EQU     0               MEMORY READ
     000.001                61X DM.MW   EQU     1               MEMORY WRITE
     000.002                62X DM.RR   EQU     2               REGISTER READ
     000.003                63X DM.RW   EQU     3               REGISTER WRITE


                            65X **      USER OPTION BITS.
                            66X *
                            67X *       THESE BITS ARE SET IN CELL .MFLAG.
                            68X
     000.200                69X UO.HLT  EQU     10000000B       DISABLE HALT PROCESSING
     000.100                70X UO.NFR  EQU     CB.CLI          NO REFRESH OF FRONT PANEL
     000.002                71X UO.DDU  EQU     00000010B       DISABLE DISPLAY UPDATE
     000.001                72X UO.CLK  EQU     00000001B       ALLOW PRIVATE INTERRUPT PROCESSING


                            74X **      MONITOR IDENTIFICATION FLAGS
                            75X *
                            76X *       THESE BYTES IDENTIFY THE ROM MONITOR.
                            77X *       THEY ARE THE VARIOUS VALUES OF LOCATION .IDENT
                            78X
     000.021                79X M.PAM8  EQU     021Q            'LXI' INSTRUCTION AT 000.000 IN PAM-8
     000.303                80X M.FOX   EQU     303Q            'JMP' INSTRUCTION AT 000.000 IN FOX ROM
```

```
                              82X **      ROUTINE ENTRY POINTS.
                              83X *
                              84X
     000.000                  85X .IDENT  EQU     0000A         IDENTIFICATION LOCATION
     000.053                  86X .DLY    EQU     0053A         DELAY
     001.267                  87X .LOAD   EQU     1267A         TAPE LOAD
     001.374                  88X .DUMP   EQU     1374A         TAPE DUMP
     002.136                  89X .ALARM  EQU     2136A         ALARM ROUTINE
     002.140                  90X .HORN   EQU     2140A         HORN
     002.172                  91X .CTC    EQU     2172A         CHECK TAPE CHECKSUM
     002.205                  92X .TPERR  EQU     2205A         TAPE ERROR ROUTINE
     002.264                  93X .PCHL   EQU     2264A         PCHL INSTRUCTION
     002.265                  94X .SRS    EQU     2265A         SCAN RECORD START
     002.325                  95X .RNP    EQU     2325A         READ NEXT PAIR
     002.331                  96X .RNB    EQU     2331A         READ NEXT BYTE
     002.347                  97X .CRC    EQU     2347A         CRC-16 CALCULATOR
     003.017                  98X .WNP    EQU     3017A         WRITE NEXT PAIR
     003.024                  99X .WNB    EQU     3024A         WRITE NEXT BYTE
     003.122                 100X .DOD    EQU     3122A         DECODE FOR OCTAL DISPLAY
     003.260                 101X .RCK    EQU     3260A         READ CONSOLE KEYSET
     003.356                 102X .DODA   EQU     3356A         SEGMENT CODE TABLE


                             104X **      RAM CELLS USED BY H8MTR.
                             105X *
                             106X
     040.000                 107X .START  EQU     40000A        START DUMP ADDRESS
     040.002                 108X .IOWRK  EQU     40002A        IN OR OUT INSTRUCTION
     040.005                 109X .REGI   EQU     40005A        DISPLAYED REGISTER INDEX
     040.006                 110X .DSPROT EQU     40006A        PERIOD FLAG BYTE
     040.007                 111X .DSPMOD EQU     40007A        DISPLAY MODE
     040.010                 112X .MFLAG  EQU     40010A        USER OPTION BYTE
     040.011                 113X .CTLFLG EQU     40011A        PANEL CONTROL BYTE
     040.013                 114X .ALEDS  EQU     40013A        ABUSS LEDS
     040.021                 115X .DLEDS  EQU     40021A        DBUSS LEDS
     040.024                 116X .ABUSS  EQU     40024A        ABUSS REGISTER
     040.027                 117X .CRCSUM EQU     40027A        CRCSUM WORD
     040.031                 118X .TPERRX EQU     40031A        TAPE ERROR EXIT VECTOR
     040.033                 119X .TICCNT EQU     40033A        CLOCK TICK COUNTER
     040.035                 120X .REGPTR EQU     40035A        REGISTER POINTER
     040.037                 121X .UIVEC  EQU     40037A        USER INTERRUPT VECTORS
     000.000                 122          XTEXT   BOODEF


                             124X **      BOODEF - SPECIAL BOOT-HDOS INTERFACE DEFINTIONS.
                             125X
     047.000                 126X SB.ORG  EQU     47000A        ORG FOR LOAD OF INITIAL HDOS.SAV
     014.000                 127X SB.OVMX EQU     14000A        SIZE OF HOLD AREA FOR SWAPPED USER CODE
                             128X *                             (=MAX SIZE OF HDOSOVL.SYS)
     000.000                 129          XTEXT   ECDEF
```

```
                          131X **      ERROR CODE DEFINITIONS.
                          132X
    000.000               133X         ORG      0
    000.000               134X         DS       1               NO ERROR #0
    000.001               135X EC.EOF  DS       1               END OF FILE
    000.002               136X EC.EOM  DS       1               END OF MEDIA
    000.003               137X EC.ILC  DS       1               ILLEGAL SYSCALL CODE
    000.004               138X EC.CNA  DS       1               CHANNEL NOT AVAILABLE
    000.005               139X EC.DNS  DS       1               DEVICE NOT SUITABLE
    000.006               140X EC.IDN  DS       1               ILLEGAL DEVICE NAME
    000.007               141X EC.IFN  DS       1               ILLEGAL FILE NAME
    000.010               142X EC.NRD  DS       1               NO ROOM FOR DEVICE DRIVER
    000.011               143X EC.FNO  DS       1               CHANNEL NOT OPEN
    000.012               144X EC.ILR  DS       1               ILLEGAL REQUEST
    000.013               145X EC.FUC  DS       1               FILE USAGE CONFLICT
    000.014               146X EC.FNF  DS       1               FILE NAME NOT FOUND
    000.015               147X EC.UND  DS       1               UNKNOWN DEVICE
    000.016               148X EC.ICN  DS       1               ILLEGAL CHANNEL NUMBER
    000.017               149X EC.DIF  DS       1               DIRECTORY FULL
    000.020               150X EC.IFC  DS       1               ILLEGAL FILE CONTENTS
    000.021               151X EC.NEM  DS       1               NOT ENOUGH MEMORY
    000.022               152X EC.RF   DS       1               READ FAILURE
    000.023               153X EC.WF   DS       1               WRITE FAILURE
    000.024               154X EC.WPV  DS       1               WRITE PROTECTION VIOLATION
    000.025               155X EC.WP   DS       1               DISK WRITE PROTECTED
    000.026               156X EC.FAP  DS       1               FILE ALREADY PRESENT
    000.027               157X EC.DDA  DS       1               DEVICE DRIVER ABORT
    000.030               158X EC.FL   DS       1               FILE LOCKED
    000.031               159X EC.FAO  DS       1               FILE ALREADY OPEN
    000.032               160X EC.IS   DS       1               ILLEGAL SWITCH
    000.033               161X EC.UUN  DS       1               UNKNOWN UNIT NUMBER
    000.034               162X EC.FNR  DS       1               FILE NAME REQUIRED
    000.035               163X EC.DIW  DS       1               DEVICE IS NOT WRITABLE (OR WRITE LOCKED)
    000.036               164X EC.UNA  DS       1               UNIT NOT AVAILABLE
    000.037               165X EC.ILV  DS       1               ILLEGAL VALUE
    000.040               166X EC.ILO  DS       1               ILLEGAL OPTION
    000.041               167X EC.VPM  DS       1               VOLUME PRESENTLY MOUNTED ON DEVICE
    000.042               168X EC.NVM  DS       1               NO VOLUME PRESENTLY MOUNTED
    000.043               169X EC.FOD  DS       1               FILE OPEN ON DEVICE
    000.044               170X EC.NPM  DS       1               NO PROVISIONS MADE FOR REMOUNTING MORE DISKS
    000.045               171X EC.DNI  DS       1               DISK NOT INITIALIZED
    000.046               172X EC.DNR  DS       1               DISK IS NOT READABLE
    000.047               173X EC.DSC  DS       1               DISK STRUCTURE IS CORRUPT
    000.050               174X EC.NCV  DS       1               NOT CORRECT VERSION OF HDOS
    000.051               175X EC.NOS  DS       1               NO OPERATING SYSTEM MOUNTED
    000.052               176X EC.IOI  DS       1               ILLEGAL OVERLAY INDEX
    000.053               177X EC.OTL  DS       1               OVERLAY TO LARGE
    000.054               178          XTEXT    H17DEF
```

```
                          180X **      H17 CONTROL INFORMATION.
                          181X
     000.177              182X DF.DC   EQU     07FH            DISK CONTROL PORT
                          183X
     000.001              184X DF.HD   EQU     00000001B       HOLE DETECT
     000.002              185X DF.TO   EQU     00000010B       TRACK 0 DETECT
     000.004              186X DF.WP   EQU     00000100B       WRITE PROTECT
     000.010              187X DF.SD   EQU     00001000B       SYNC DETECT
                          188X
     000.001              189X DF.WG   EQU     00000001B       WRITE GATE ENABLE
     000.002              190X DF.DS0  EQU     00000010B       DRIVE SELECT 0
     000.004              191X DF.DS1  EQU     00000100B       DRIVE SELECT 1
     000.010              192X DF.DS2  EQU     00001000B       DRIVE SELECT 2
     000.020              193X DF.MO   EQU     00010000B       MOTOR ON (BOTH DRIVES)
     000.040              194X DF.DI   EQU     00100000B       DIRECTION (0=OUT)
     000.100              195X DF.ST   EQU     01000000B       STEP COMMAND (ACTIVE HIGH)
     000.200              196X DF.WR   EQU     10000000B       WRITE ENABLE RAM
                          197X
                          198X
                          199X
                          200X **      DISK UART PORTS AND CONTROL FLAGS.
                          201X
     000.174              202X UP.DP   EQU     07CH            DATA PORT
     000.175              203X UP.FC   EQU     07DH            FILL CHARACTER
     000.175              204X UP.ST   EQU     07DH            STATUS FLAGS
     000.176              205X UP.SC   EQU     07EH            SYN CHARACTER (OUTPUT)
     000.176              206X UP.SR   EQU     07EH            SYNC RESET (INPUT)
                          207X
     000.001              208X UF.RDA  EQU     00000001B       RECEIVE DATA AVAILABLE
     000.002              209X UF.ROR  EQU     00000010B       RECEIVER OVERRUN
     000.004              210X UF.RPE  EQU     00000100B       RECEIVER PARITY ERROR
     000.100              211X UF.FCT  EQU     01000000B       FILL CHAR TRANSMITTED
     000.200              212X UF.TBM  EQU     10000000B       TRANSMITTER BUFFER EMPTY
                          213X
                          214X
                          215X
                          216X **      CHARACTER DEFINITIONS.
                          217X
     000.375              218X C.DSYN  EQU     0FDH            PREFIX SYNC CHARACTER
     000.054              219          XTEXT   U8251
```

```
                         222X **      8251 USART BIT DEFINITIONS.
                         223X *
                         224X
                         225X **      PORT ADDRESSES
                         226X
   000.000               227X UDR     EQU     0               DATA REGISTER IS EVEN
   000.001               228X USR     EQU     1               STATUS REGISTER IS NEXT
                         229X
   000.372               230X SC.UART EQU     372Q            CONSOLE USART ADDRESS (IFF 8251)
                         231X
                         232X
                         233X **      MODE INSTRUCTION CONTROL BITS.
                         234X
   000.100               235X UMI.1B  EQU     01000000B       1 STOP BIT
   000.200               236X UMI.HB  EQU     10000000B       1 1/2 STOP BITS
   000.300               237X UMI.2B  EQU     11000000B       2 STOP BITS
   000.040               238X UMI.PE  EQU     00100000B       EVEN PARITY
   000.020               239X UMI.PA  EQU     00010000B       USE PARITY
   000.000               240X UMI.L5  EQU     00000000B       5 BIT CHARACTERS
   000.004               241X UMI.L6  EQU     00000100B       6 BIT CHARACTERS
   000.010               242X UMI.L7  EQU     00001000B       7 BIT CHARACTERS
   000.014               243X UMI.L8  EQU     00001100B       8 BIT CHARACTERS
   000.001               244X UMI.1X  EQU     00000001B       CLOCK X 1
   000.002               245X UMI.16X EQU     00000010B       CLOCK X 16
   000.003               246X UMI.64X EQU     00000011B       CLOCK X 64
                         247X
                         248X **      COMMAND INSTRUCTION BITS.
                         249X
   000.100               250X UCI.IR  EQU     01000000B       INTERNAL RESET
   000.040               251X UCI.RO  EQU     00100000B       READER-ON CONTROL FLAG
   000.020               252X UCI.ER  EQU     00010000B       ERROR RESET
   000.004               253X UCI.RE  EQU     00000100B       RECEIVE ENABLE
   000.002               254X UCI.IE  EQU     00000010B       ENABLE INTERRUPTS FLAG
   000.001               255X UCI.TE  EQU     00000001B       TRANSMIT ENABLE
                         256X
                         257X **      STATUS READ COMMAND BITS.
                         258X
   000.040               259X USR.FE  EQU     00100000B       FRAMING ERROR
   000.020               260X USR.OE  EQU     00010000B       OVERRUN ERROR
   000.010               261X USR.PE  EQU     00001000B       PARITY ERROR
   000.004               262X USR.TXE EQU     00000100B       TRANSMITTER EMPTY
   000.002               263X USR.RXR EQU     00000010B       RECEIVER READY
   000.001               264X USR.TXR EQU     00000001B       TRANSMITTER READY
   000.054               265          XTEXT   U8250



                         267X **      8250 UART CONTROL AND BIT DEFINITIONS.
                         268X
   000.350               269X SC.ACE  EQU     350Q            SYSTEM CONSOLE PORT IF 8250 ACE
   000.156               270X AC.DLY  EQU     110             220 MIL. SEC. DELAY FOR 8250
                         271X
   000.000               272X UR.RBR  EQU     0               RECEIVER BUFFER REGISTER (READ ONLY)
                         273X
   000.000               274X UR.THR  EQU     0               TRANSMITTER HOLDING REGISTER (WRITE ONLY)
```

```
                        275X
      000.000           276X UR.DLL   EQU     0             DIVISOR LATCH (LEAST SIGNIFICANT)
                        277X
      000.001           278X UR.DLM   EQU     1             DIVISOR LATCH (MOST SIGNIFICANT)
                        279X
      000.001           280X UR.IER   EQU     1             INTERRUPT ENABLE REGISTER
      000.001           281X UC.EDA   EQU     00000001B      ENABLE RECEIVED DATA AVAILABLE INTERRUPT
      000.002           282X UC.TRE   EQU     00000010B      ENABLE TRANSMIT HOLD REGISTER EMPTY INTERRUPT
      000.004           283X UC.RSI   EQU     00000100B      ENABLE RECEIVE STATUS INTERRUPT
      000.010           284X UC.MSI   EQU     00001000B      ENABLE MODEM STATUS INTERRUPT
                        285X
      000.002           286X UR.IIR   EQU     2             INTERRUPT IDENTIFICATION REGISTER
      000.001           287X UC.IIP   EQU     00000001B      INVERTED INTERRUPT PENDING (0 MEANS PENDING)
      000.006           288X UC.IID   EQU     00000110B      INTERRUPT ID
                        289X
      000.003           290X UR.LCR   EQU     3             LINE CONTROL REGISTER
      000.000           291X UC.5BW   EQU     00000000B      5 BIT WORDS
      000.001           292X UC.6BW   EQU     00000001B      6 BIT WORDS
      000.002           293X UC.7BW   EQU     00000010B      7 BIT WORDS
      000.003           294X UC.8BW   EQU     00000011B      8 BIT WORDS
      000.004           295X UC.2SB   EQU     00000100B      TWO STOP BITS SELECTED
      000.010           296X UC.PEN   EQU     00001000B      PARITY COMPUTATION ENABLED
      000.020           297X UC.EPS   EQU     00010000B      EVEN PARITY SELECT
      000.040           298X UC.SKP   EQU     00100000B      STICK PARITY
      000.100           299X UC.SB    EQU     01000000B      SET BREAK
      000.200           300X UC.DLA   EQU     10000000B      DIVISOR LATCH ACCESS
                        301X
      000.004           302X UR.MCR   EQU     4             MODEM CONTROL REGISTER
      000.001           303X UC.DTR   EQU     00000001B      DATA TERMINAL READY
      000.002           304X UC.RTS   EQU     00000010B      REQUEST TO SEND
      000.004           305X UC.OU1   EQU     00000100B      OUT 1
      000.010           306X UC.OU2   EQU     00001000B      OUT 2
      000.020           307X UC.LOO   EQU     00010000B      LOOP
                        308X
      000.005           309X UR.LSR   EQU     5             LINE STATUS REGISTER
      000.001           310X UC.DR    EQU     00000001B      DATA READY
      000.002           311X UC.OR    EQU     00000010B      OVERRUN
      000.004           312X UC.PE    EQU     00000100B      PARITY ERROR
      000.010           313X UC.FE    EQU     00001000B      FRAMING ERROR
      000.020           314X UC.BI    EQU     00010000B      BREAK INTERRUPT
      000.040           315X UC.THE   EQU     00100000B      TRANSMITTER HOLDING REGISTER EMPTY
      000.100           316X UC.TSE   EQU     01000000B      TRANSMITTER SHIFT REGISTER EMPTY
                        317X
      000.006           318X UR.MSR   EQU     6             MODEM STATUS REGISTER
      000.001           319X UC.DCS   EQU     00000001B      DELTA CLEAR TO SEND
      000.002           320X UC.DDR   EQU     00000010B      DELTA DATA SET READY
      000.004           321X UC.TER   EQU     00000100B      TRAILING EDGE OF RING
      000.010           322X UC.DRL   EQU     00001000B      DELTA RECEIVE LINE SIGNAL DETECT
      000.020           323X UC.CTS   EQU     00010000B      CLEAR TO SEND
      000.040           324X UC.DSR   EQU     00100000B      DATA SET READY
      000.100           325X UC.RI    EQU     01000000B      RING INDICATOR
      000.200           326X UC.RLS   EQU     10000000B      RECEIVED LINE SIGNAL DETECT
      000.054           327           XTEXT   DDDEF         DEVICE DRIVER CONSTANTS
```

```
                         329X **      DEVICE DRIVER COMMUNICATION FLAGS.
                         330X *
                         331X
   000.000               332X         ORG     0
                         333X
   000.000               334X DC.REA  DS      1               READ
   000.001               335X DC.WRI  DS      1               WRITE
   000.002               336X DC.RER  DS      1               READ REGARDLESS
   000.003               337X DC.OPR  DS      1               OPEN FOR READ
   000.004               338X DC.OPW  DS      1               OPEN FOR WRITE
   000.005               339X DC.OPU  DS      1               OPEN FOR UPDATE
   000.006               340X DC.CLO  DS      1               CLOSE
   000.007               341X DC.ABT  DS      1               ABORT
   000.010               342X DC.MOU  DS      1               MOUNT DEVICE
   000.011               343X DC.LOD  DS      1               LOAD DEVICE DRIVER
   000.012               344X DC.MAX  DS      1               MAXIMUM ENTRY INDEX
   000.013               345          XTEXT   FILDEF


                         347X **      FILDEF - FILE TYPE DEFINITIONS.
                         348X *
                         349X *       DB      377Q,FT.XXX
                         350X
                         351X
   000.000               352X FT.ABS  EQU     0               ABSOLUTE BINARY
   000.001               353X FT.PIC  EQU     1               POSITION INDEPENDANT CODE
   000.002               354X FT.REL  EQU     2               RELOCATABLE CODE
   000.003               355X FT.BAC  EQU     3               COMPILED BASIC CODE
   000.013               356          XTEXT   ABSDEF


                         358X **      ABS FORMAT EQUIVALENCES.
                         359X
   000.000               360X         ORG     0
                         361X
   000.000               362X ABS.ID  DS      1               377Q = BINARY FILE FLAG
   000.001               363X         DS      1               FILE TYPE (FT.ABS)
   000.002               364X ABS.LDA DS      2               LOAD ADDRESS
   000.004               365X ABS.LEN DS      2               LENGTH OF ENTIRE RECORD
   000.006               366X ABS.ENT DS      2               ENTRY POINT
                         367X
   000.010               368X ABS.COD DS      0               CODE STARTS HERE
   000.010               369          XTEXT   DIRDEF
```

```
                            371X **      DIRECTORY ENTRY FORMAT.
                            372X
      000.000               373X         ORG      0
                            374X
                            375X
      000.377               376X DF.EMP  EQU      377Q            FLAGS ENTRY EMPTY
      000.376               377X DF.CLR  EQU      376Q            FLAGS ENTRY EMPTY, REST OF DIR ALSO CLEAR
                            378X
      000.000               379X DIR.NAM DS       8               NAME
      000.010               380X DIR.EXT DS       3               EXTENSION
      000.013               381X DIR.PRO DS       1               PROJECT
      000.014               382X DIR.VER DS       1               VERSION
      000.015               383X DIRIDL  EQU      *               FILE IDENTIFICATION LENGTH
                            384X
      000.015               385X DIR.CLU DS       1               CLUSTER FACTOR
      000.016               386X DIR.FLG DS       1               FLAGS
      000.017               387X         DS       1               RESERVED
      000.020               388X DIR.FGN DS       1               FIRST GROUP NUMBER
      000.021               389X DIR.LGN DS       1               LAST GROUP NUMBER
      000.022               390X DIR.LSI DS       1               LAST SECTOR INDEX (IN LAST GROUP)
      000.023               391X DIR.CRD DS       2               CREATION DATE
      000.025               392X DIR.ALD DS       2               LAST ALTERATION DATE
                            393X
      000.027               394X DIRELEN EQU      *               DIRECTORY ENTRY LENGTH
      000.027               395          XTEXT    DISDEF


                            397X **      DIRECTORY BLOCK FORMAT.
                            398X
      000.000               399X         ORG      0
                            400X
      000.000               401X DIS.ENT EQU      *               FIRST ENTRY ADDRESS
      000.000               402X         DS       22*DIRELEN      22 DIRECTORY ENTRYS PER BLOCK
      001.372               403X         DS       1               0 BYTE = END OF ENTRYS IN THIS BLOCK
                            404X
      001.373               405X         ORG      512-5           AT END OF BLOCK
      001.373               406X DIS.ENL DS       1               LENGTH OF EACH ENTRY (=DIRELEN)
      001.374               407X DIS.SEC DS       2               BLOCK # OF THIS BLOCK,
      001.376               408X DIS.LNK DS       2               BLOCK # OF NEXT BLOCK, =0 IF THIS IS LAST
      002.000               409          XTEXT    DEVDEF


                            411X **      DEVICE TABLE ENTRYS.
                            412X
      000.000               413X         ORG      0
                            414X
      000.000               415X DEV.NAM DS       2               DEVICE NAME
      000.000               416X DV.EL   EQU      00000000B       END OF DEVICE LIST FLAG
      000.001               417X DV.NU   EQU      00000001B       DEVICE ENTRY NOT IN USE
                            418X
      000.002               419X DEV.RES DS       1               DRIVER RESIDENSE CODE
      000.001               420X DR.IM   EQU      00000001B       DRIVER IN MEMORY
      000.002               421X DR.PR   EQU      00000010B       DRIVER PERMINANTLY RESIDENT
```

```
                             422X
     000.003                 423X DEV.JMP DS      1            JMP TO PROCESSOR
     000.004                 424X DEV.DDA DS      2            DRIVER ADDRESS
     000.006                 425X DEV.FLG DS      1            FLAG BYTE
     000.001                 426X DT.DD   EQU     00000001B    DIRECTORY DEVICE
     000.002                 427X DT.CR   EQU     00000010B    CAPABLE OF READ OPERATION
     000.004                 428X DT.CW   EQU     00000100B    CAPABLE OF WRITE OPERATION
                             429X
     000.007                 430X DEV.SPG DS      1            SECTORS PER GROUP THIS DEVICE
     000.010                 431X DEV.MUM DS      1            MOUNTED UNIT MASK
     000.011                 432X DEV.MNU DS      1            MAXIMUM NUMBER OF UNITS
     000.012                 433X DEV.UNT DS      2            ADDRESS OF UNIT SPECIFIC DATA TABLE
                             434X
     000.014                 435X DEV.DVL DS      2            DRIVER BYTE LENGTH
     000.016                 436X DEV.DVG DS      1            DRIVER ROUTINE GROUP ADDRESS
                             437X
     000.017                 438X DEVELEN EQU     *            DEVICE TABLE ENTRY LENGTH



                             440X **      UNIT SPECIFIC DEVICE DATA TABLE ENTRIES
                             441X
     000.000                 442X         ORG     0
                             443X
     000.000                 444X UNT.FLG DS      1            UNIT SPECIFIC  *DEV.FLG*
     000.001                 445X UNT.GRT DS      2            ADDRESS OF GROUP RESERVATION TABLE (IF DT.DD)
     000.003                 446X UNT.GTS DS      2            GRT SECTOR NUMBER
     000.005                 447X UNT.DIS DS      2            DIRECTORY FIRST SECTOR NUMBER
                             448X
     000.007                 449X UNT.SIZ EQU     *            SIZE OF UNIT SPECIFIC DATA TABLE PER UNIT
     000.007                 450          XTEXT   DDFDEF


                             452X **      DIRECTORY DEVICE FORMAT DEFINITION.
                             453X *
                             454X
                             455X
     000.002                 456X HOS.SPG EQU     2            2 SECTORS PER GROUP REQUIRED FOR NOW
                             457X
     000.000                 458X         ORG     0
     000.000                 459X DDF.BOO DS      9            2K BOOT PROGRAM
     000.011                 460X DDF.BOL EQU     *            LENGTH OF BOOT
     000.011                 461X DDF.LAB DS      1            LABEL SECTOR
     000.012                 462X DDF.RGT DS      2            RESERVED GROUP TABLE
     000.014                 463X DDF.USR DS      0            BEGINNING OF OPEN SPACE
     000.014                 464          XTEXT   LABDEF
```

```
                             466X **      DISK LABEL SECTOR FORMATS.
                             467X
     000.000                 468X         ORG     0
     000.000                 469X LAB.SER DS      1               SERIAL NUMBER OF VOLUME
     000.001                 470X LAB.IND DS      2               INITIALIZATION DATE
     000.003                 471X LAB.DIS DS      2               SECTOR NUMBER OF 1ST DIRECTORY SECTOR
     000.005                 472X LAB.GRT DS      2               INDEX OF GRT SECTOR
     000.007                 473X LAB.SPG DS      1               SECTORS PER GROUP
                             474X
     000.000                 475X LAB.DAT EQU     0               DATA VOLUME ONLY
     000.001                 476X LAB.SYS EQU     1               SYSTEM VOLUME
     000.002                 477X LAB.NOD EQU     2               => LAB.NOD MEANS VOLUME HAS NO DIRECTORY
                             478X
     000.010                 479X LAB.VLT DS      1               VOLUME TYPE
     000.011                 480X LAB.VER DS      1               VERSION OF INIT17 THAT INITED DISK
     000.012                 481X         DS      7               UNUSED
     000.021                 482X LAB.LAB DS      60              LABEL
     000.074                 483X LAB.LBL EQU     *-LAB.LAB       LABEL LENGTH
     000.115                 484          XTEXT   DIFDEF



                             486X **      DIRECTORY FILE FLAGS.
                             487X
     000.200                 488X DIF.SYS EQU     10000000B               SYSTEM FILE
     000.100                 489X DIF.LOC EQU     01000000B               LOCKED FOR CHANGE
     000.040                 490X DIF.WP  EQU     00100000B               WRITE PROTECTED
     000.020                 491X DIF.CNT EQU     00010000B               CONTIGUOUS FILE
                             492X
     000.115                 493          XTEXT   NAMDEF



                             495X **      SYSTEM FILE NAME CONVENTIONS
                             496X *
                             497X *       RGT     .SYS            RESERVED GROUP TABLE (1 SECTOR)
                             498X *       GRT     .SYS            GROUP RESERVATION TABLE (1 SECTOR)
                             499X *       DIRECT  .SYS            DIRECTORY
                             500X *       HOS     .SYS            SYSTEM IMAGE PROGRAM FOR SYSTEM
                             501X
     000.115                 502          XTEXT   OVLDEF



                             504X **      OVERLAY TABLE ENTRYS.
                             505X
     000.000                 506X         ORG     0
                             507X
     000.000                 508X OVL.COD DS      2               FIRST SECTOR OF OVERLAY CODE
     000.002                 509X OVL.SIZ DS      2               OVERLAY SIZE
     000.004                 510X OVL.ENT DS      2               OVERLAY ENTRY POINT
     000.006                 511X OVL.FLB DS      1               OVERLAY FLAG BYTE
     000.007                 512X         DS      1               DUMMY BYTE TO ROUND TABLE SIZE UP TO 8
     000.010              513X OVL.ENS EQU     *                OVERLAY ENTRY SIZE
                          514X
                          515X *       OVERLAY INDICES
                          516X
     000.000              517X         ORG     0
                          518X
     000.000              519X OVL0    DS      1
     000.001              520X OVL1    DS      1
     000.002              521          XTEXT   HOSEQU



                          523X **      HDOS SYSTEM EQUIVALENCES.
                          524X *
                          525X
     024.000              526X S.GRT0  EQU     24000A           SYSTEM AREA FOR  GRT0
     025.000              527X S.GRT1  EQU     25000A           SYSTEM AREA FOR  GRT1
     026.000              528X S.GRT2  EQU     26000A           SYSTEM AREA FOR  GRT2
                          529X
     030.000              530X ROMBOOT EQU     30000A           ROM BOOT ENTRY
                          531X
     040.100              532X         ORG     40100A           FREE SPACE FROM PAM-8
                          533X
     040.100              534X         DS      8                JUMP TO SYSTEM EXIT
     040.110              535X D.CON   DS      16               DISK CONSTANTS
     040.130              536X SYDD    EQU     *                SYSTEM DISK ENTRY POINT
     040.130              537X D.VEC   DS      24*3             SYSTEM ROM ENTRY VECTORS
     040.240              538X D.RAM   DS      31               SYSTEM ROM WORK AREA
     040.277              539X S.VAL   DS      36               SYSTEM VALUES
     040.343              540X S.INT   DS      115              SYSTEM INTERNAL WORK AREAS
     041.126              541X         DS      16
     041.146              542X S.SOVR  DS      2                STACK OVERFLOW WARNING
     041.150              543X         DS      42200A-*         SYSTEM STACK
     001.032              544X STACKL  EQU     *-S.SOVR         STACK SIZE
                          545X
     042.200              546X STACK   EQU     *                LWA+1 SYSTEM STACK
     042.200              547X USERFWA EQU     *                USER FWA
     042.200              548          XTEXT   HOSDEF



                          550X **      HOSDEF - DEFINE HOS PARAMETER.
                          551X *
                          552X
                          553X
     000.026              554X VERS    EQU     1*16+6           VERSION 1.6
                          555X
     000.377              556X SYSCALL EQU     377Q             SYSCALL INSTRUCTION
                          557X
                          558X
     000.000              559X         ORG     0
                          560X
                          561X *       RESIDENT FUNCTIONS
                          562X
     000.000              563X .EXIT   DS      1                EXIT (MUST BE FIRST)
     000.001                564X .SCIN   DS      1              SCIN
     000.002                565X .SCOUT  DS      1              SCOUT
     000.003                566X .PRINT  DS      1              PRINT
     000.004                567X .READ   DS      1              READ
     000.005                568X .WRITE  DS      1              WRITE
     000.006                569X .CONSL  DS      1              SET/CLEAR CONSOLE OPTIONS
     000.007                570X .CLRCO  DS      1              CLEAR CONSOLE BUFFER
     000.010                571X .LOADO  DS      1              LOAD AN OVERLAY
     000.011                572X .VERS   DS      1              RETURN HDOS VERSION NUMBER
     000.012                573X .SYSRES DS      1              PRECEDING FUNCTIONS ARE RESIDENT
                            574X
                            575X
                            576X *       *HDOSOVL0.SYS*  FUNCTIONS
                            577X
     000.040                578X         ORG     40A
                            579X
     000.040                580X .LINK   DS      1              LINK  (MUST BE FIRST)
     000.041                581X .CTLC   DS      1              CTL-C
     000.042                582X .OPENR  DS      1              OPENR
     000.043                583X .OPENW  DS      1              OPENW
     000.044                584X .OPENU  DS      1              OPENU
     000.045                585X .OPENC  DS      1              OPENC
     000.046                586X .CLOSE  DS      1              CLOSE
     000.047                587X .POSIT  DS      1              POSITION
     000.050                588X .DELET  DS      1              DELETE
     000.051                589X .RENAM  DS      1              RENAME
     000.052                590X .SETTP  DS      1              SETTOP
     000.053                591X .DECODE DS      1              NAME DECODE
     000.054                592X .NAME   DS      1              GET FILE NAME FROM CHANNEL
     000.055                593X .CLEAR  DS      1              CLEAR CHAN
     000.056                594X .CLEARA DS      1              CLEAR ALL CHANS
     000.057                595X .ERROR  DS      1              LOOKUP ERROR
     000.060                596X .CHFLG  DS      1              CHANGE FLAGS
     000.061                597X .DISMT  DS      1              FLAG SYSTEM DISK DISMOUNTED
     000.062                598X .LOADD  DS      1              LOAD DEVICE DRIVER
                            599X
                            600X
                            601X *       *HDOSOVL1.SYS*  FUNCTIONS
                            602X
     000.200                603X         ORG     200Q
                            604X
     000.200                605X .MOUNT  DS      1              MOUNT  (MUST BE FIRST)
     000.201                606X .DMOUN  DS      1              DISMOUNT
     000.202                607X .MONMS  DS      1              MOUNT/NO MESSAGE
     000.203                608X .DMNMS  DS      1              DISMOUNT/NO MESSAGE
     000.204                609X .RESET  DS      1              RESET = DISMOUNT/MOUNT OF UNIT
     000.205                610          XTEXT   EDRAM
```

```
                         612X **       EDRAM - DISK RAM WORKAREA DEFINITION.
                         613X *
                         614X *        ZEROED UPON BOOTING UP.
                         615X *
                         616X *        HOSEQU MUST BE CHANGED WHEN THIS DECK IS CHANGED.
                         617X
                         618X
    040.240              619X          ORG     D.RAM
                         620X
    040.240              621X D.TT     DS      1                  TARGET TRACK (CURRENT OPERATION)
    040.241              622X D.TS     DS      1                  TARGET SECTOR (CURRENT OPERATION)
                         623X
    040.242              624X D.DVCTL  DS      1                  DEVICE CONTROL BYTE
                         625X
    040.243              626X D.DLYMO  DS      1                  MOTOR ON DELAY COUNT
    040.244              627X D.DLYHS  DS      1                  HEAD SETTLE DELAY COUNTER
                         628X
    040.245              629X D.TRKPT  DS      2                  ADDRESS IN D.DRVTB FOR TRACK NUMBER
    040.247              630X D.VOLPT  DS      2                  ADDRESS IN D.DRVTB FOR VOLUME NUMBER
                         631X
    040.251              632X D.DRVTB  DS      2*4                TRACK NUMBER AND VOLUME NUMBER FOR 4 DRIVES
                         633X
    040.261              634X D.HECNT  DS      1                  HARD ERROR COUNT
    040.262              635X D.SECNT  DS      2                  SOFT ERROR COUNT
    040.264              636X D.OECNT  DS      1                  OPERATION ERROR COUNT
                         637X
                         638X *        GLOBAL DISK ERROR COUNTERS
                         639X
    040.265              640X D.ERR    DS      0                  BEGINNING OF ERROR BLOCK
    040.265              641X D.E.MDS  DS      1                  MISSING DATA SYNC
    040.266              642X D.E.HSY  DS      1                  MISSING HEADER SYNC
    040.267              643X D.E.CHK  DS      1                  DATA CHECKSUM
    040.270              644X D.E.HCK  DS      1                  HEADER CHECKSUM
    040.271              645X D.E.VOL  DS      1                  WRONG VOLUME NUMBER
    040.272              646X D.E.TRK  DS      1                  BAD TRACK SEEK
    040.273              647X D.ERRL   DS      0                  LIMIT OF ERROR COUNTERS
                         648X
                         649X *        I/O OPERATION COUNTS
                         650X
    040.273              651X D.OPR    DS      2
    040.275              652X D.OPW    DS      2
                         653X
    000.037              654X D.RAML   EQU     *-D.RAM
    040.277              655           XTEXT   EDVEC



                         657X **       JMP VECTORS FOR ROM CODE
                         658X *
                         659X *        SEE DISK ROM FOR ADDRESSES
                         660X *
                         661X *        HOSEQU MUST BE ALTERED WHEN THIS TABLE IS ALTERED.
                         662X
    040.130              663X          ORG     D.VEC
                         664X
```

```
        040.130                 665X D.SYDD  DS      3               JMP  R.SYDD (MUST BE FIRST)
        040.133                 666X D.MOUNT DS      3               JMP  R.MOUNT
        040.136                 667X D.XOK   DS      3               JMP  R.XOK
        040.141                 668X D.ABORT DS      3               JMP  R.ABORT
        040.144                 669X D.XIT   DS      3               JMP  R.XIT
        040.147                 670X D.READ  DS      3               JMP  R.READ
        040.152                 671X D.READR DS      3               JMP  R.READR
        040.155                 672X D.WRITE DS      3               JMP  R.WRITE
        040.160                 673X D.CDE   DS      3               JMP  R.CDE
        040.163                 674X D.DTS   DS      3               JMP  R.DTS
        040.166                 675X D.SDT   DS      3               JMP  R.SDT
        040.171                 676X D.MAI   DS      3               JMP  R.MAI
        040.174                 677X D.MAO   DS      3               JMP  R.MAO
        040.177                 678X D.LPS   DS      3               JMP  R.LPS
        040.202                 679X D.RDB   DS      3               JMP  R.RDB
        040.205                 680X D.SDP   DS      3               JMP  R.SDP
        040.210                 681X D.STS   DS      3               JMP  R.STS
        040.213                 682X D.STZ   DS      3               JMP  R.STZ
        040.216                 683X D.UDLY  DS      3               JMP  R.UDLY
        040.221                 684X D.WSC   DS      3               JMP  R.WSC
        040.224                 685X D.WSP   DS      3               JMP  R.WSP
        040.227                 686X D.WNB   DS      3               JMP  R.WNB
        040.232                 687X D.ERRT  DS      3               JMP  R.ERRT
        040.235                 688X D.DLY   DS      3               JMP  R.DLY
        040.240                 689          XTEXT   ESVAL



                                691X **      S.VAL - SYSTEM VALUE DEFINTIONS.
                                692X *
                                693X *       THESE VALUES ARE SET AND MAINTAINED BY THE SYSTEM.
                                694X *
                                695X *       THE DECK HOSEQU MUST BE MODIFIED WHEN THIS IS MODIFIED.
                                696X
                                697X
        040.277                 698X         ORG     S.VAL
                                699X
        040.277                 700X S.DATE  DS      9               SYSTEM DATE (IN ASCII)
        040.310                 701X S.DATC  DS      2               CODED DATE
        040.312                 702X S.TIME  DS      4               TIME FROM MIDNIGHT (IN TICS)
        040.316                 703X S.HIMEM DS      2               HARDWARE HIGH MEMORY ADRESS+1
                                704X
        040.320                 705X S.SYSM  DS      2               FWA RESIDENT SYSTEM
                                706X
        040.322                 707X S.USRM  DS      2               LWA USER MEMORY
                                708X
        040.324                 709X S.OMAX  DS      2               MAX OVERLAY SIZE FOR SYSTEM
                                710X
                                711X
                                712X **      THE FOLLOWING FIVE CELLS SHOULD BE MODIFIED/READ ONLY VIA THE .CONSL SYSCALL
                                713X
        000.200                 714X CSL.ECH EQU     10000000B       SUPPRESS ECHO
        000.002                 715X CSL.WRP EQU     00000010B       WRAP LINES AT WIDTH
        000.001                 716X CSL.CHR EQU     00000001B       OPERATE IN CHARACTER MODE
                                717X
```

```
    000.000               718X I.CSLMD EQU     0               S.CSLMD IS FIRST BYTE
    040.326               719X S.CSLMD DS      1               CONSOLE MODE
                          720X
    000.200               721X CTP.BKS EQU     10000000B       TERMINAL PROCESSES BACKSPACES
    000.040               722X CTP.MLI EQU     00100000B       MAP LOWER CASE TO UPPER ON INPUT
    000.020               723X CTP.MLO EQU     00010000B       MAP LOWER CASE TO UPPER ON OUTPUT
    000.010               724X CTP.2SB EQU     00001000B       TERMINAL NEEDS TWO STOP BITS
    000.002               725X CTP.BKM EQU     00000010B       MAP BKSP (UPON INPUT) TO RUBOUT
    000.001               726X CTP.TAB EQU     00000001B       TERMINAL SUPPORTS TAB CHARACTERS
                          727X
    000.001               728X I.CONTY EQU     1               S.CONTY IS 2ND BYTE
    000.000               729X         ERRNZ   *-S.CSLMD-I.CONTY
    040.327               730X S.CONTY DS      1               CONSOLE TYPE FLAGS
    000.002               731X I.CUSOR EQU     2               S.CUSOR IS 3RD BYTE
    000.000               732X         ERRNZ   *-S.CSLMD-I.CUSOR
    040.330               733X S.CUSOR DS      1               CURRENT CURSOR POSITION
    000.003               734X I.CONWI EQU     3               S.CONWI IS 4TH BYTE
    000.000               735X         ERRNZ   *-S.CSLMD-I.CONWI
    040.331               736X S.CONWI DS      1               CONSOLE WIDTH
                          737X
    000.001               738X CO.FLG  EQU     00000001B       CTL-O FLAG
    000.200               739X CS.FLG  EQU     10000000B       CTL-S FLAG
                          740X
    000.004               741X I.CONFL EQU     4               S.CONFL IS 5TH BYTE
    000.000               742X         ERRNZ   *-S.CSLMD-I.CONFL
    040.332               743X S.CONFL DS      1               CONSOLE FLAGS
                          744X
    040.333               745X S.CAADR DS      2               ADDRESS FOR ABORT PROCESSING (>256 IF VALID)
    040.335               746X S.CCTAB DS      6               ADDR FOR CTL-A, CTL-B, CTL-C PROCESSING
    040.343               747          XTEXT   ESINT


                          749X **      S.INT - SYSTEM INTERNAL WORKAREA DEFINITIONS.
                          750X *
                          751X *       THESE CELLS ARE REFERENCED BY OVERLAYS AND MAIN CODE, AND
                          752X *       MUST THEREFORE RESIDE IN FIXED LOW MEMORY.
                          753X
                          754X
    040.343               755X         ORG     S.INT
                          756X
                          757X **      CONSOLE STATUS FLAGS
                          758X
    040.343               759X S.CDB   DS      1               CONSOLE DESCRIPTOR BYTE
    000.000               760X CDB.H85 EQU     00000000B
    000.001               761X CDB.H84 EQU     00000001B       =0 IF H8-5, =1 IF H8-4
    040.344               762X S.BAUD  DS      2               [0-14]  H8-4 BAUD RATE, =0 IF H8-5
                          763X *                               [15]    =1 IF BAUD RATE => 2 STOP BITS
                          764X
                          765X **      TABLE ADDRESS WORDS
                          766X
    040.346               767X S.DLINK DS      2               ADDRESS OF DATA IN HDOS CODE
    040.350               768X S.OFWA  DS      2               FWA  OVERLAY  TABLE
    040.352               769X S.CFWA  DS      2               FWA  CHANNEL  TABLE
    040.354               770X S.DFWA  DS      2               FWA  DEVICE   TABLE
```

```
       040.356                      771X S.RFWA  DS      2            FWA  RESIDENT HDOS CODE
                                    772X
                                    773X **      DEVICE DRIVER DELAYED LOAD FLAGS
                                    774X
       040.360                      775X S.DDLDA DS      2            DRIVER LOAD ADDRESS (HIGH BYTE=0 IF NO LOAD PENDING)
       040.362                      776X S.DDLEN DS      2            CODE LENGTH IN BYTES
       040.364                      777X S.DDGRP DS      1            GROUP NUMBER FOR DRIVER
       040.365                      778X         DS      1            HOLD PLACE
                                    779X *S.DDSEC        DS      2            SECTOR NUMBER FOR DRIVER ( * OBSOLETE ! * )
       040.366                      780X S.DDDTA DS      2            DEVICE'S ADDRESS IN DEVLST +DEV.RES
       040.370                      781X S.DDOPC DS      1            OPEN OPCODE PENDEDING
                                    782X
                                    783X **      OVERLAY MANAGEMENT FLAGS
                                    784X
       000.001                      785X OVL.IN  EQU     00000001B    IN MEMORY
       000.002                      786X OVL.RES EQU     00000010B    PERMINANTLY RESIDENT
       000.014                      787X OVL.NUM EQU     00001100B    OVERLAY NUMBER MASK
       000.200                      788X OVL.UCS EQU     10000000B    USER CODE SWAPPED FOR OVERLAY
                                    789X
       040.371                      790X S.OVLFL DS      1            OVERLAY FLAG
       040.372                      791X S.UCSF  DS      2            FWA SWAPPED USER CODE
       040.374                      792X S.UCSL  DS      2            LENGTH SWAPPED USER CODE
       040.376                      793X S.OVLS  DS      2            SIZE OF OVERLAY CODE
       041.000                      794X S.OVLE  DS      2            ENTRY POINT OF OVERLAY CODE
                                    795X
       041.002                      796X S.SSN   DS      2            SWAP AREA SECTOR NUMBER
       041.004                      797X S.OSN   DS      2            OVERLAY SECTOR NUMBER
                                    798X
                                    799X *       SYSCALL PROCESSING WORK AREAS
                                    800X
       041.006                      801X S.CACC  DS      1            (ACC) UPON SYSCALL
       041.007                      802X S.CODE  DS      1            SYSCALL INDEX IN PROGRESS
                                    803X
                                    804X *       JUMPS TO ROUTINES IN RESIDENT HDOS CODE
                                    805X
       041.010                      806X S.JUMPS DS      0            START OF DUMP VECTORS
       041.010                      807X S.SDD   DS      3            JUMP TO STAND-IN DEVICE DRIVER
       041.013                      808X S.FASER DS      3            JUMP TO FATSERR (FATAL SYSTEM ERROR)
       041.016                      809X S.DIREA DS      3            JUMP TO DIREAD (DISK FILE READ)
       041.021                      810X S.FCI   DS      3            JUMP TO FCI (FETCH CHANNEL INFO)
       041.024                      811X S.SCI   DS      3            JUMP TO SCI (STORE CHANNEL INFO)
       041.027                      812X S.GUP   DS      3            JUMP TO GUP (GET UNIT POINTER)
                                    813X
       041.032                      814X S.MOUNT DS      1            <>0 IF THE SYSTEM DISK IS MOUNTED
       041.033                      815X S.DCS   DS      1            DEFAULT CLUSTER SIZE-1
                                    816X
       041.034                      817X S.BOOTF DS      1            BOOT FLAGS
       000.001                      818X BOOT.P  EQU     00000001B    EXECUTE PROLOGUE UPON BOOTUP
                                    819X
                                    820X *       STACK VALUE SAVED FOR OVERLAY SYSCALLS
                                    821X
       041.035                      822X S.OVSTK DS      2            VALUE OF SP UPON SYSCALLS USING OVERLAY
                                    823X
       041.037                      824X         DS      1            RESERVED
```

```
                        826X **      ACTIVE I/O AREA.
                        827X *
                        828X *       THE AIO.XXX AREA CONTAINS INFORMATION ABOUT THE I/O OPERATION
                        829X *       CURRENTLY BEING PERFORMED. THE INFORMATION IS OBTAINED FROM
                        830X *       THE CHANNEL TABLE, AND WILL BE RESTORED THERE WHEN DONE.
                        831X *
                        832X *       NORMALLY, THE AIO.XXX INFORMATION WOULD BE OBTAINED DIRECTLY
                        833X *       FROM VARIOUS SYSTEM TABLES VIA POINTER REGISTERS. SINCE THE
                        834X *       8080 HAS NO GOOD INDEXED ADDRESSING, THE DATA IS MANUALLY
                        835X *       COPIED INTO THE AIO.XXX CELLS BEFORE PROCESSING, AND
                        836X *       BACKDATED AFTER PROCESSING.
                        837X
    041.040             838X AIO.VEC DS      3               JUMP INSTRUCTION
    041.041             839X AIO.DDA EQU     *-2             DEVICE DRIVER ADDRESS
    041.043             840X AIO.FLG DS      1               FLAG BYTE
    041.044             841X AIO.GRT DS      2               ADDRESS OF GROUP RESERV TABLE
    041.046             842X AIO.SPG DS      1               SECTORS PER GROUP
    041.047             843X AIO.CGN DS      1               CURRENT GROUP NUMBER
    041.050             844X AIO.CSI DS      1               CURRENT SECTOR INDEX
    041.051             845X AIO.LGN DS      1               LAST GROUP NUMBER
    041.052             846X AIO.LSI DS      1               LAST SECTOR INDEX
    041.053             847X AIO.DTA DS      2               DEVICE TABLE ADDRESS
    041.055             848X AIO.DES DS      2               DIRECTORY SECTOR
    041.057             849X AIO.DEV DS      2               DEVICE CODE
    041.061             850X AIO.UNI DS      1               UNIT NUMBER (0-9)
                        851X
    041.062             852X AIO.DIR DS      DIRELEN         DIRECTORY ENTRY
                        853X
    041.111             854X AIO.CNT DS      1               SECTOR COUNT
    041.112             855X AIO.EOM DS      1               END OF MEDIA FLAG
    041.113             856X AIO.EOF DS      1               END OF FILE FLAG
    041.114             857X AIO.TFP DS      2               TEMP FILE POINTERS
    041.116             858X AIO.CHA DS      2               ADDRESS OF CHANNEL BLOCK (IOC.DDA)



    041.120             860X S.SCR   DS      2               SYSTEM SCRATCH AREA ADDRESS
    041.122             861          XTEXT   H17ROM



                        863X **      H17 ROM DEFINITIONS
                        864X
    036.235             865X R.WHD   EQU     36235A
    036.271             866X R.WNH   EQU     36271A
    035.303             867X R.DLY   EQU     35303A
    041.122             868          XTEXT   PICDEF
```

```
                          870X **      PIC FORMAT EQUIVALENCES.
                          871X
    000.000               872X         ORG     0
                          873X
    000.000               874X PIC.ID  DS      1               377Q = BINARY FILE FLAG
    000.001               875X         DS      1               FILE TYPE (FT.PIC)
    000.002               876X PIC.LEN DS      2               LENGTH OF ENTIRE RECORD
    000.004               877X PIC.PTR DS      2               INDEX OF START OF PIC TABLE
                          878X
    000.006               879X PIC.COD DS      0               CODE STARTS HERE
    000.006               880          XTEXT   ASCII

                          882X **      ASCII CHARACTER EQUIVALENCES.
                          883X
    000.015               884X CR      EQU     13              CARRIAGE RETURN
    000.012               885X LF      EQU     10              LINE FEED
    000.200               886X NULL    EQU     200Q            PAD CHARACTER
    000.000               887X NUL2    EQU     0
    000.007               888X BELL    EQU     7               BELL CHARACTER
    000.177               889X RUBOUT  EQU     177Q
    000.010               890X BKSP    EQU     10Q             CTL-H
    000.026               891X C.SYN   EQU     26Q             SYNC
    000.002               892X C.STX   EQU     2               STX
    000.047               893X QUOTE   EQU     47Q
    000.011               894X TAB     EQU     11Q
    000.033               895X ESC     EQU     33Q
    000.012               896X NL      EQU     12Q             NEW LINE (HDOS SYSTEMS)
    000.212               897X ENL     EQU     NL+200Q         NL + END-OF-LINE-FLAG
    000.014               898X FF      EQU     14Q             FORM FEED
    000.001               899X CTLA    EQU     01Q             CTL-A
    000.002               900X CTLB    EQU     02Q             CTL-B
    000.003               901X CTLC    EQU     03Q             CTL-C
    000.004               902X CTLD    EQU     04Q             CTL-D
    000.017               903X CTLO    EQU     17Q             CTL-O
    000.020               904X CTLP    EQU     20Q             CTL-P
    000.021               905X CTLQ    EQU     21Q             CTL-Q
    000.023               906X CTLS    EQU     23Q             CTL-S
    000.032               907X CTLZ    EQU     32Q             CTL-Z
```

```
042.170                          ORG     USERFWA-ABS.COD


042.170  377 000                 DB      377Q,FT.ABS
042.172  200 042                 DW      USERFWA         LOAD ADDR
042.174  122 022                 DW      MEML-USERFWA    SIZE
042.176  010 051                 DW      INIT            ENTRY

                                 LON     C


                         **      S0BOOT  - SECTOR 0 BOOT ROUTINE.
                         *
                         *       THIS BOOT STARTS AT SECTOR 0 ON EVERY INITIALIZED
                         *       DISK, AND OCCUPIES THE FIRST 9 SECTORS OF THE DISK.
                         *
                         *       IT IS BROUGHT IN BY THE H17 ROM.



042.200  257             S0BOOT  XRA     A
042.201  323 175                 OUT     UP.FC           SET FILL CHARACTER = 0
042.203  315 000 046             CALL    FCU             FIND CONSOLE USART
042.206  315 155 046             CALL    MSD             MOUNT SYSTEM DISK

042.211  061 200 042     S0BOOTX LXI     SP,STACK
042.214  257                     XRA     A
042.215  062 062 041             STA     AIO.DIR         AM WORKING WITH NO FILES YET
042.220  315 055 045             CALL    $TYPET
042.223  015 012 012             DB      CR,LF,LF,'ACTION? <BOOT>',' '+200Q

                         *       GET REPLY, MAY BE:
                         *
                         *       BOOT
                         *       CHECK
                         *       HELP

042.245  315 257 044             CALL    $ICTT           INPUT TASK TIME
042.250  315 007 050             CALL    $MCU            MAP TO UPPER CASE
042.253  376 015                 CPI     CR
042.255  312 042 043             JE      S0BOOT2         IS BOOT
042.260  376 102                 CPI     'B'
042.262  312 042 043             JE      S0BOOT2         IS BOOT
042.265  376 103                 CPI     'C'
042.267  312 105 043             JE      S0BOOT4         IS CHECK
042.272  376 111                 CPI     'I'
042.274  312 124 043             JE      S0BOOT5         IS IGNORE

                         *       ASSUME HELP

042.277  315 055 045             CALL    $TYPET
042.302  110 105 114             DB      'HELP',0,0
042.310  114 105 107             DB      'LEGAL COMMANDS:',0
042.330  102 117 117             DB      'BOOT   - BOOT HDOS',0
042.353  103 110 105             DB      'CHECK  - SECTOR CHECKSUMS',0
043.005  110 105 114             DB      'HELP   - PRINT THIS LIST',0
```

```
000.001                 966               IF      PUBLIC
                        967               DB      'IGNORE - IGNORE PROLOGUE FILE'
                        968               ENDIF
043.036  200            969               DB      200Q
043.037  303 211 042    970               JMP     SOBOOTX          TRY AGAIN
                        971
                        972  *            IS BOOT
                        973
043.042  315 055 045    974  SOBOOT2 CALL    $TYPET
043.045  102 117 117    975               DB      'BOOT',200Q
043.052  072 034 041    976               LDA     S.BOOTF
043.055  366 001        977               ORI     BOOT.P           FLAY PROLOGUE EXECUTION UPON BOOTUP
043.057  062 034 041    978               STA     S.BOOTF
043.062  315 155 046    979  SOBOOT3 CALL    MSD              MOUNT THIS DISK
043.065  315 151 043    980               CALL    LEP              LOAD AND EXECUTE PROGRAM
043.070  110 104 117    981               DB      'HDOS',0,0,0,0
043.100  123 131 123    982               DB      'SYS',0,0
                        983
                        984  *            IS CHECK
                        985
043.105  315 055 045    986  SOBOOT4 CALL    $TYPET
043.110  103 110 105    987               DB      'CHECK',200Q
043.116  315 020 050    988               CALL    CDC              COMPUTE DISK CCHECKS
043.121  303 211 042    989               JMP     SOBOOTX          TRY AGAIN
                        990
                        991  *            IS IGNORE
                        992
043.124  315 055 045    993  SOBOOT5 CALL    $TYPET
043.127  111 107 116    994               DB      'IGNORE',200Q
043.136  072 034 041    995               LDA     S.BOOTF
043.141  346 376        996               ANI     377Q-BOOT.P
043.143  062 034 041    997               STA     S.BOOTF          TURN OFF PROLOGUE UPON BOOT FLAG
043.146  303 062 043    998               JMP     SOBOOT3
```

```
                      1001  **      LEP IS CALLED TO LOAD AND EXECUTE A DISK FILE.
                      1002  *
                      1003  *       THE DISKS DIRECTORY IS SEARCHED FOR THE APPROPRIATE FILE NAME.
                      1004  *       IF FOUND, IT IS LOADED INTO MEMORY AT SB.ORG AND EXECUTED.
                      1005  *
                      1006  *       IF NOT FOUND, TYPE ERROR MESSAGE:
                      1007  *
                      1008  *       NEEDED FILE * FNAME * IS MISSING
                      1009  *
                      1010  *       AND RETURN TO SOBOOT.
                      1011  *
                      1012  *       IF ERROR IN READING THE FILE, TYPE
                      1013  *
                      1014  *       'DISK READ ERROR IN FILE * FNAME *'
                      1015  *
                      1016  *       AND RE-BOOT.
                      1017  *
                      1018  *       ENTRY   ((SP)) = FILE NAME
                      1019  *       EXIT    TO SB.ORG IF LOAD SUCCESSFUL,
                      1020  *               TO ROMBOOT IF READ ERROR,
                      1021  *               TO SOBOOTX IF FILE MISSING
                      1022  *       USES    ALL
                      1023
                      1024
043.151  321          1025  LEP     POP     D                (DE) = NAME ADDRESS
043.152  001 015 000  1026          LXI     B,DIRIDL
043.155  041 062 041  1027          LXI     H,AIO.DIR
043.160  315 252 030  1028          CALL    $MOVE            MOVE IN NAME
043.163  072 020 051  1029          LDA     BLABEL+LAB.VLT           (A) = VOLUME TYPE
043.166  247          1030          ANA     A
000.000               1031          ERRNZ   LAB.DAT
043.167  312 032 044  1032          JZ      LEP5             IS DATA DISK
043.172  075          1033          DCR     A
000.000               1034          ERRNZ   LAB.SYS-1
043.173  302 131 044  1035          JNZ     LEP6             IS GARBAGE DISK
043.176  001 015 000  1036          LXI     B,DIRIDL
043.201  052 013 051  1037          LHLD    BLABEL+LAB.DIS
043.204  315 212 045  1038          CALL    LDE..            LOAD ENTRY
043.207  322 262 043  1039          JNC     LEP1             FOUND
                      1040
                      1041  *       COULDNT FIND IT
                      1042
043.212  315 055 045  1043          CALL    $TYPET
043.215  007 077 060  1044          DB      BELL,'?00 REQUIRED FILE',' '+200Q
043.240  315 341 044  1045          CALL    TFN              TYPE FILE NAME
043.243  315 055 045  1046          CALL    $TYPET
043.246  040 115 111  1047          DB      ' MISSING',BELL+200Q
043.257  303 211 042  1048          JMP     SOBOOTX
                      1049
                      1050  *       GOT DIRECTORY ENTRY. TRY TO READ IT
                      1051
043.262  021 016 000  1052  LEP1    LXI     D,DIR.FLG
043.265  031          1053          DAD     D                (HL) = ADDRESS OF FLG
043.266  176          1054          MOV     A,M
043.267  346 020      1055          ANI     DIF.CNT
043.271  312 343 043  1056          JZ      LEP4             NOT CONTIGUOUG
```

```
000.000                        ERRNZ  DIR.FGN-DIR.FLG-2
043.274  043                   INX    H
043.275  043                   INX    H                (HL) = #DIR.FGN
043.276  136                   MOV    E,M
043.277  026 000               MVI    D,0              (DE) = GROUP NUMBER
043.301  072 017 051           LDA    BLABEL+LAB.SPG   (A) = SECTORS PER GROUP
043.304  315 007 031           CALL   $MU86            (HL) = SECTOR NUMBER FOR FILE

                         *     (HL) = SECTOR NUMBER FOR FILE

043.307  001 000 001     LEP3  LXI    B,256
043.312  021 000 047           LXI    D,SB.ORG
043.315  345                   PUSH   H
043.316  315 276 045           CALL   READD            READ DISK
043.321  052 002 047           LHLD   SB.ORG+PIC.LEN
043.324  053                   DCX    H                (HL) = SECTOR COUNT
043.325  104                   MOV    B,H
043.326  016 000               MVI    C,0
043.330  341                   POP    H                (HL) = SECTOR NUMBER OF FWA
043.331  043                   INX    H                ALREADY READ 1
043.332  021 000 050           LXI    D,SB.ORG+256
043.335  315 276 045           CALL   READD            READ THE REMAINDER
043.340  303 006 047           JMP    SB.ORG+PIC.COD   ALL OK, EXECUTE IT

                         *     FILE NOT CONTIGUOUS

043.343  315 055 045     LEP4  CALL   $TYPET
043.346  000 007 077           DB     0,BELL,'?00 THIS DISK HAS NOT BEEN PROPERLY SYSGENED.',BELL,200Q
044.027  303 131 044           JMP    LEP6

                         *     IS DATA DISK. NOT YET SYSGENED

044.032  315 055 045     LEP5  CALL   $TYPET
044.035  000 007 077           DB     0,BELL,'?00 THIS DISK MUST BE SYSGENED BEFORE IT CAN BE USED.',BELL,200Q
044.126  303 211 042           JMP    SOBOOTX

                         *     GARBAGE DISK.

044.131  315 055 045     LEP6  CALL   $TYPET
044.134  000 007 077           DB     0,BELL,'?00 THIS DISK MUST BE INITIALIZED AND THEN SYSGENED'
044.221  000 040 040           DB     0,'  BEFORE IT CAN BE USED.',BELL,200Q
044.254  303 211 042           JMP    SOBOOTX
```

```
     044.257              1101        XTEXT   ICTT


                          1103X **        $ICTT - INPUT FROM CONSOLE TASK TIME.
                          1104X *
                          1105X *         $ICTT IS A TASK-TIME CONSOLE INPUT ROUTINE, WHICH
                          1106X *         PERFORMS SIMPLE SINGLE CHARACTER INPUTS.
                          1107X *
                          1108X *         IT IS CALLED DURING BOOT OPERATIONS, AND BY SPECIAL ROUTINES
                          1109X *         WHICH MAY BE RUNNING IN ENVIRONMENTS WHERE KEYBOARD INTERRUPTS
                          1110X *         ARE UNDESIRABLE.
                          1111X *
                          1112X *         Modified to handle H8-4 ports by G. Chandler, 1-SEP-78
                          1113X *         This routine assumes that the ports have been previously initialized,
                          1114X *         and that S.CDB has been previously initialized.
                          1115X *
                          1116X *         ENTRY   NONE
                          1117X *         EXIT    (A) = CHARACTER
                          1118X *         USES    A,F
                          1119X
                          1120X
     044.257  315 271 044 1121X $ICTT     CALL    $ICTT.
     044.262  332 257 044 1122X           JC      $ICTT
     044.265  315 317 044 1123X           CALL    $ICTT..
     044.270  311         1124X           RET
                          1125X
     044.271  072 343 040 1126X $ICTT.    LDA     S.CDB
     044.274  376 001     1127X           CPI     CDB.H84
     044.276  312 310 044 1128X           JZ      ICTT2           IF  H8-4 PORT
                          1129X
                          1130X *         HAVE 8251 FOR CONSOLE
                          1131X
     044.301  333 373     1132X ICTT1     IN      SC.UART+USR
     044.303  346 002     1133X           ANI     USR.RXR
     044.305  300         1134X           RNZ                     READY
                          1135X
     044.306  067         1136X           STC                     FLAG NOT READY
     044.307  311         1137X           RET
                          1138X
                          1139X *         HAVE 8250 PORT FOR CONSOLE
                          1140X
     044.310  333 355     1141X ICTT2     IN      SC.ACE+UR.LSR
     044.312  346 001     1142X           ANI     UC.DR
     044.314  300         1143X           RNZ                     READY
                          1144X
     044.315  067         1145X           STC                     FLAG NOT READY
     044.316  311         1146X           RET
                          1147X
     044.317  072 343 040 1148X $ICTT..   LDA     S.CDB
     044.322  376 001     1149X           CPI     CDB.H84
     044.324  312 334 044 1150X           JZ      ICTT3
                          1151X
                          1152X *         HAVE 8251 FOR CONSOLE
                          1153X
```

```
044.327  333 372       1154X          IN      SC.UART+UDR
044.331  346 177       1155X          ANI     177Q
044.333  311           1156X          RET
                       1157X
                       1158X *        HAVE 8250 FOR CONSOLE
                       1159X
044.334  333 350       1160X ICTT3    IN      SC.ACE+UR.RBR
044.336  346 177       1161X          ANI     177Q
044.340  311           1162X          RET



                       1164  **       TFN - TYPE FILE NAME.
                       1165  *
                       1166  *        TFN TYPES THE FILE WHOSE NAME APPEARS IN AIO.XXX
                       1167  *
                       1168  *        ENTRY   NONE
                       1169  *        EXIT    NONE
                       1170  *        USES    A,F,B,H,L
                       1171
                       1172
044.341  041 062 041   1173  TFN      LXI     H,AIO.DIR+DIR.NAM
044.344  006 010       1174           MVI     B,8
044.346  315 360 044   1175           CALL    TFN1            TYPE NAME
044.351  076 056       1176           MVI     A,'.'
044.353  315 112 045   1177           CALL    $TYPEC.
044.356  006 003       1178           MVI     B,3
                       1179
044.360  176           1180  TFN1     MOV     A,M
044.361  247           1181           ANA     A
044.362  304 112 045   1182           CNZ     $TYPEC.
044.365  043           1183           INX     H
044.366  005           1184           DCR     B
044.367  302 360 044   1185           JNZ     TFN1
044.372  311           1186           RET
044.373                1187           XTEXT   BCRC



                       1189X **       $BCRC - GENERATE CRC16 ON A BLOCK OF DATA.
                       1190X *
                       1191X *        *** WARNING ***
                       1192X *
                       1193X *        THIS CRC-16 IS NOT COMPATIBLE WITH THE ONE
                       1194X *        PRODUCED BY PAM-8, AND THE DECK CRC.COM!
                       1195X *
                       1196X *        ENTRY   (BC) = BYTE COUNT
                       1197X *                (HL) = ADDRESS
                       1198X *                (DE) = CRC ACCUMULATOR
                       1199X *        EXIT    (HL) = (HL)+(BC)
                       1200X *                (DE) = NEW CRC
                       1201X *        USES    ALL
                       1202X
                       1203X
```

```
044.373  170          1204X $BCRC   MOV     A,B
044.374  261          1205X         ORA     C
044.375  310          1206X         RZ                      NO MORE
044.376  176          1207X         MOV     A,M             (A) = NEW BYTE
044.377  345          1208X         PUSH    H
045.000  305          1209X         PUSH    B               SAVE REGISTERS
045.001  253          1210X         XRA     E
045.002  107          1211X         MOV     B,A
045.003  017          1212X         RRC
045.004  017          1213X         RRC
045.005  017          1214X         RRC
045.006  017          1215X         RRC
045.007  117          1216X         MOV     C,A
045.010  250          1217X         XRA     B
045.011  346 360      1218X         ANI     0F0H
045.013  252          1219X         XRA     D
045.014  157          1220X         MOV     L,A
045.015  171          1221X         MOV     A,C
045.016  007          1222X         RLC
045.017  346 037      1223X         ANI     1FH
045.021  255          1224X         XRA     L
045.022  157          1225X         MOV     L,A
045.023  170          1226X         MOV     A,B
045.024  007          1227X         RLC
045.025  346 001      1228X         ANI     1
045.027  252          1229X         XRA     D
045.030  255          1230X         XRA     L
045.031  127          1231X         MOV     D,A
045.032  171          1232X         MOV     A,C
045.033  346 360      1233X         ANI     0F0H
045.035  250          1234X         XRA     B
045.036  137          1235X         MOV     E,A
045.037  171          1236X         MOV     A,C
045.040  250          1237X         XRA     B
045.041  007          1238X         RLC
045.042  346 340      1239X         ANI     0E0H
045.044  253          1240X         XRA     E
045.045  137          1241X         MOV     E,A
045.046  301          1242X         POP     B
045.047  341          1243X         POP     H
045.050  043          1244X         INX     H
045.051  013          1245X         DCX     B
045.052  303 373 044  1246X         JMP     $BCRC
045.055               1247          XTEXT   MU86


                      1249X **      $MU86 - MULTIPLY 8X16 UNSIGNED.
                      1250X *
                      1251X *       $MU86 MULTIPLIES A 16 BIT VALUE BY A 8
                      1252X *       BIT VALUE.
                      1253X *
                      1254X *       ENTRY   (A) = MULTIPLIER
                      1255X *               (DE) = MULTIPLICAND
                      1256X *       EXIT    (HL) = RESULT
```

```
                           1257X *                'Z' SET IF NOT OVERFLOW
                           1258X *        USES    A,F,H,L
                           1259X
                           1260X
  031.007                  1261X $MU86    EQU     31007A            IN H17 ROM
  045.055                  1262           XTEXT   TYPET



                           1264X **       $TYPET - TYPE TEXT.
                           1265X *
                           1266X *        $TYPET IS CALLED TO TYPE A BLOCK OF TEXT ON THE SYSTEM CONSOLE
                           1267X *        AT TASK TIME RATHER THAN AT INTERRUPT TIME.
                           1268X *
                           1269X *        IMBEDDED ZERO BYTES INDICATE A CARRIAGE RETURN LINE FEED.
                           1270X *        A BYTE WITH THE 200Q BIT SET IS THE LAST BYTE OF THE MESSAGE.
                           1271X *
                           1272X *        This routine modified to accomodate H8-4 ports by G.Chandler, 1-SEP-78.
                           1273X *        This routine assumes that the ports have been previously initialized,
                           1274X *        and that S.CDB has been previously initialized.
                           1275X *
                           1276X *        ENTRY   (RET) = TEXT
                           1277X *        EXIT    TO (RET+LENGTH)
                           1278X *        USES    A,F
                           1279X
                           1280X
  045.055  343             1281X $TYPET   XTHL                      (HL) = TEXT ADDRESS
  045.056  315 063 045     1282X          CALL    $TYPET.           TYPE IT
  045.061  343             1283X          XTHL
  045.062  311             1284X          RET
                           1285X
  045.063  176             1286X $TYPET.  MOV     A,M
  045.064  346 177         1287X          ANI     177Q
  045.066  304 112 045     1288X          CNZ     $TYPEC.           IF NOT CRLF
  045.071  247             1289X          ANA     A
  045.072  314 103 045     1290X          CZ      $TYPET1           IS CRLF
  045.075  276             1291X          CMP     M
  045.076  043             1292X          INX     H
  045.077  300             1293X          RNE                       WAS 200 BIT SET
  045.100  303 063 045     1294X          JMP     $TYPET.
                           1295X
                           1296X *        TYPE CRLF
                           1297X
  045.103  315 055 045     1298X $TYPET1  CALL    $TYPET
  045.106  015 212         1299X          DB      CR,LF+200Q
  045.110  257             1300X          XRA     A                 RESTORE (A)
  045.111  311             1301X          RET
```

```
                          1303X **       $TYPEC. - TYPE SINGLE CHARACTER.
                          1304X *
                          1305X *        IF CR, PADD WITH 4 ZERO BYTES
                          1306X *
                          1307X *        ENTRY   (A) = CHARACTER
                          1308X *        EXIT    (A) = CHARACTER
                          1309X *        USES    A,F
                          1310X
                          1311X
045.112  365              1312X $TYPEC.  PUSH    PSW              SAVE CHAR
045.113  072 343 040      1313X          LDA     S.CDB
045.116  376 001          1314X          CPI     CDB.H84
045.120  312 140 045      1315X          JZ      TYPEC2           IF H8-4 PORT
                          1316X
                          1317X *        HAVE 8251 PORT FOR CONSOLE
                          1318X
045.123  333 373          1319X TYPEC1   IN      SC.UART+USR
045.125  346 001          1320X          ANI     USR.TXR
045.127  312 123 045      1321X          JZ      TYPEC1           NOT READY
045.132  361              1322X          POP     PSW
045.133  323 372          1323X          OUT     SC.UART+UDR
045.135  303 152 045      1324X          JMP     TYPEC3
                          1325X
                          1326X *        HAVE 8250 PORT FOR CONSOLE
                          1327X
045.140  333 355          1328X TYPEC2   IN      SC.ACE+UR.LSR
045.142  346 040          1329X          ANI     UC.THE
045.144  312 140 045      1330X          JZ      TYPEC2           NOT READY
045.147  361              1331X          POP     PSW
045.150  323 350          1332X          OUT     SC.ACE+UR.THR
                          1333X
045.152  376 015          1334X TYPEC3   CPI     CR
045.154  300              1335X          RNE                      NOT CR
                          1336X
                          1337X *        IS CR, PADD 4 TIMES
                          1338X
045.155  076 004          1339X          MVI     A,4
045.157  365              1340X TYPEC4   PUSH    PSW
045.160  257              1341X          XRA     A
045.161  315 112 045      1342X          CALL    $TYPEC.
045.164  361              1343X          POP     PSW
045.165  075              1344X          DCR     A
045.166  302 157 045      1345X          JNZ     TYPEC4
045.171  076 015          1346X          MVI     A,CR
045.173  311              1347X          RET
045.174                   1348           XTEXT   MOVE
```

```
                         1350X **        $MOVE - MOVE DATA
                         1351X *
                         1352X *         $MOVE MOVES A BLOCK OF BYTES TO A NEW MEMORY ADDRESS.
                         1353X *         IF THE MOVE IS TO A LOWER ADDRESS, THE BYTES ARE MOVED FROM
                         1354X *         FIRST TO LAST.
                         1355X *
                         1356X *         IF THE MOVE IS TO A HIGHER ADDRESS, THE BYTES ARE MOVED FROM
                         1357X *         LAST TO FIRST.
                         1358X *
                         1359X *         THIS IS DONE SO THAT AN OVERLAPED MOVE WILL NOT 'RIPPLE'.
                         1360X *
                         1361X *         ENTRY   (BC) = COUNT
                         1362X *                 (DE) = FROM
                         1363X *                 (HL) = TO
                         1364X *         EXIT    MOVED
                         1365X *                 (DE) = ADDRESS OF NEXT FROM BYTE
                         1366X *                 (HL) = ADDRESS OF NEXT *TO* BYTE
                         1367X *                 'C' CLEAR
                         1368X *         USES    ALL
                         1369X
                         1370X
   030.252               1371X $MOVE     EQU     30252A          IN H17 ROM
   045.174               1372            XTEXT   COMP


                         1374X **        $COMP - COMPARE TWO CHARACTER STRINGS.
                         1375X *
                         1376X *         $COMP COMPARES TWO BYTE STRINGS.
                         1377X *
                         1378X *         ENTRY   (C) = COMPARE COUNT
                         1379X *                 (DE) = FWA OF STRING #1
                         1380X *                 (HL) = FWA OF STRING #2
                         1381X *         EXIT    'Z' CLEAR, IS MIS-MATCH
                         1382X *                  (C) = LENGTH REMAINING
                         1383X *                  (DE) = ADDRESS OF MISMATCH IN STRING#1
                         1384X *                  (HL) = ADDRESS OF MISMATCH IN STRING #2
                         1385X *                 'C' SET, HAVE MATCH
                         1386X *                  (C) = 0
                         1387X *                  (DE) = (DE) + (OC)
                         1388X *                  (HL) = (HL) + (OC)
                         1389X *         USES    A,F,C,D,E,H,L
                         1390X
                         1391X
   030.060               1392X $COMP     EQU     30060A          IN H17 ROM
   045.174               1393            XTEXT   DADA2
```

```
                     **      $DADA. - ADD (0,A) TO (H,L)
                     *
                     *       ENTRY   NONE
                     *       EXIT    (HL) = (HL) + (0A)
                     *       USES    A,F,H,L


030.101              $DADA.  EQU     30101A          IN H17 ROM

```

```
                     **      CSC - COMPUTE SECTOR CRC.
                     *
                     *       CSC IS CALLED TO COMPUTE THE CRC OVER A SECTOR'S WORTH OF
                     *       DATA.
                     *
                     *       ENTRY   (HL) = CURRENT CRC VALUE
                     *               BUFF = BUFFER WITH SECTORS OF DATA
                     *               (A) = INDEX (0 TO 9) OF SECTOR IN BUFF
                     *       EXIT    (HL) = UPDATED CRC VALUE
                     *       USES    ALL


045.174 353          CSC     XCHG            (DE) = CRC VALUE
045.175 041 010 052          LXI     H,BUFF
045.200 204                  ADD     H
045.201 147                  MOV     H,A             (HL) = ADDRESS OF SECTOR
045.202 001 000 001          LXI     B,256           (BC) = COUNT
045.205 315 373 044          CALL    $BCRC           BLOCK CRC IT
045.210 353                  XCHG                    (HL) = RESULTANT CRC
045.211 311                  RET
```

```
                     **      LDE - LOCATE DIRECTORY ENTRY.
                     *
                     *       LDE LOCATES A DIRECTORY ENTRY CORRESPONDING TO THE AIO.DIR ENTRY.
                     *
                     *       ENTRY   (BC) = NUMBER OF CHARACTERS TO MATCH ON
                     *       EXIT    'C' CLEAR IF FOUND
                     *                AIO.DES SETUP
                     *                (HL) = ADDRESS OF DIRECTORY ENTRY IN BUFF
                     *               'C' SET IF NOT FOUND
                     *                (A) = CODE
                     *       USES    ALL



                     **      ENTRY FOR (HL) = SECTOR NUMBER TO START WITH

045.212 305          LDE..   PUSH    B               SAVE COUNT
045.213 001 000 002          LXI     B,512
045.216 021 010 052          LXI     D,BUFF
```

```
045.221  042 055 041               SHLD    AIO.DES          ASSUME WILL FIND IN THIS BLOCK
045.224  315 276 045               CALL    READD            READ FRM DEVICE
045.227  301                       POP     B                RESTORE (BC)

                          *        SCAN SECTOR FOR INFO

045.230  041 010 052               LXI     H,DIS.ENT+BUFF

                          *        COMPARE

045.233  021 062 041      LDE3     LXI     D,AIO.DIR
045.236  305                       PUSH    B                SAVE COPY OF (BC)
045.237  345                       PUSH    H                SAVE ADDRESS
045.240  315 060 030               CALL    $COMP            COMPARE
045.243  341                       POP     H
045.244  301                       POP     B                (BC) = COMPARE COUNT
045.245  310                       RE                       GOT MATCH
045.246  021 027 000               LXI     D,DIRELEN        MISSED, SCAN TO NEXT ENTRY
045.251  031                       DAD     D
045.252  176                       MOV     A,M
045.253  247                       ANA     A
045.254  302 233 045               JNZ     LDE3             MORE IN SECTOR

                          *        DIDNT FIND IT IN THIS SECTOR, TRY NEXT

045.257  052 006 054               LHLD    DIS.LNK+BUFF
045.262  042 055 041               SHLD    AIO.DES          SET POSSIBLE SECTOR INDEX
045.265  174                       MOV     A,H
045.266  265                       ORA     L
045.267  302 212 045               JNZ     LDE.,            HAVE MORE SECTORS
045.272  076 014                   MVI     A,EC.FNF         FILE NOT FOUND
045.274  067                       STC
045.275  311                       RET



                          **       READD - READ DISK.
                          *
                          *        READD CALLS THE SYSTEM DEVICE DRIVER FOR A
                          *        READ OPERATION.
                          *
                          *        IF AN ERROR OCCURS, A MESSAGE IS PRINTED, AND THE
                          *        BOOT OPERATION RESTARTS.
                          *
                          *        ENTRY   REGISTERS SET FOR READ
                          *        EXIT    FROM SYDD
                          *        USES    ALL


045.276  076 000          READD    MVI     A,DC.REA
045.300  315 130 040               CALL    SYDD             ISSUE READ
045.303  320                       RNC                      ALL OK

                          *        READ ERROR

```

```
045.304  315 055 045  1498  READERR CALL    $TYPET
045.307  000 000 007  1499          DB      0,0,BELL,'?00 DISK READ ERROR DURING BOOT.',0
045.353  007 040 102  1500          DB      BELL,' BOOT RESTARTED.',200Q
045.375  303 000 030  1501          JMP     ROMBOOT



                      1503  **      FCU - FIND CONSOLE USART.
                      1504  *
                      1505  *       FCU FINDS AND CONFIGURES THE CONSOLE USART.
                      1506  *
                      1507  *       ENTRY   NONE
                      1508  *       EXIT    NONE
                      1509  *       USES    A,F,(BC),(HL)
                      1510
                      1511
046.000  257          1512  FCU     XRA     A
046.001  323 351      1513          OUT     SC.ACE+UR.IER           OFF INTERRUPTS
046.003  323 373      1514          OUT     SC.UART+USR             OFF INTERRUPTS
                      1515
                      1516  *       SEE IF WE HAVE AN 8250
                      1517
046.005  076 003      1518          MVI     A,UC.8BW
046.007  323 353      1519          OUT     SC.ACE+UR.LCR
046.011  333 353      1520          IN      SC.ACE+UR.LCR
046.013  376 003      1521          CPI     UC.8BW          SEE IF UNCHANGED
046.015  076 000      1522          MVI     A,CDB.H85
046.017  041 000 000  1523          LXI     H,0
046.022  302 050 046  1524          JNE     FCU1            IS 8251
046.025  315 324 046  1525          CALL    ABR             AUTO SET BAUD RATE
046.030  174          1526          MOV     A,H
046.031  346 200      1527          ANI     10000000B
046.033  312 046 046  1528          JZ      FCU0
046.036  072 327 040  1529          LDA     S.CONTY
046.041  366 010      1530          ORI     CTP.2SB
046.043  062 327 040  1531          STA     S.CONTY         SET TWO STOP BITS
046.046  076 001      1532  FCU0    MVI     A,CDB.H84
                      1533
                      1534  *       HAVE TYPE AND BAUDRATE.
                      1535  *       (A) = S.CDB VALUE
                      1536  *       (HL) = BAUD RATE (0 IF 8251)
                      1537
046.050  042 344 040  1538  FCU1    SHLD    S.BAUD
046.053  062 343 040  1539          STA     S.CDB
046.056  315 252 047  1540          CALL    SCU
046.061  311          1541          RET
                      1542
```

```
                         1544  **       TTDD - TYPE DECIMAL DIGITS.
                         1545  *
                         1546  *        TTDD TYPES A 16 BIT VALUE AS 1 TO 5 DECIMAL DIGITS.
                         1547  *
                         1548  *        ENTRY   (D,E) = VALUE
                         1549  *                (A) = DIGIT COUNT
                         1550  *        EXIT    VALUE TYPED.
                         1551  *        USES    A,B,C,F
                         1552
                         1553
046.062  076 005         1554  TTDD.    MVI     A,5
046.064  345             1555  TTDD     PUSH    H
046.065  365             1556  TTDD1    PUSH    PSW
046.066  041 131 046     1557           LXI     H,TTDDA-2
046.071  007             1558           RLC                     (A) = DIGIT NUMBER*2
046.072  315 101 030     1559           CALL    $DADA.
046.075  176             1560           MOV     A,M
046.076  043             1561           INX     H
046.077  146             1562           MOV     H,M
046.100  157             1563           MOV     L,A             (HL) = MULTIPLE OF 10
046.101  353             1564           XCHG                    (DE) = DEVISOR, (HL) = VALUE
046.102  076 377         1565           MVI     A,377Q
046.104  031             1566  TTDD2    DAD     D
046.105  074             1567           INR     A
046.106  332 104 046     1568           JC      TTDD2           IF MORE TO GO
046.111  306 060         1569           ADI     '0'
046.113  315 112 045     1570           CALL    $TYPEC.         TYPE DIGIT
046.116  175             1571           MOV     A,L
046.117  223             1572           SUB     E
046.120  137             1573           MOV     E,A             REMOVE EXTRA SUBTRACTION
046.121  174             1574           MOV     A,H
046.122  232             1575           SBB     D
046.123  127             1576           MOV     D,A
046.124  361             1577           POP     PSW
046.125  075             1578           DCR     A
046.126  302 065 046     1579           JNZ     TTDD1           IF MORE DIGITS
046.131  341             1580           POP     H
046.132  311             1581           RET                     EXIT
                         1582
046.133                  1583  TTDDA    EQU     *
046.133  377 377         1584           DW      -1
046.135  366 377         1585           DW      -10
046.137  234 377         1586           DW      -100
046.141  030 374         1587           DW      -1000
046.143  360 330         1588           DW      -10000



                         1590  **       TTDDCR - TYPE DECIMAL DIGITS, THEN CRLF.
                         1591  *
                         1592  *        ENTRY   SAME AS TTDD
                         1593  *        EXIT    SAME AS TD
                         1594  *        USES    SAME AS TTDD
                         1595
                         1596
```

```
046.145  315 064 046  TTDDCR  CALL    TTDD
046.150  315 055 045          CALL    $TYPET
046.153  200                  DB      200Q
046.154  311                  RET


                      **      MSD - MOUNT SYSTEM DISK.
                      *
                      *       MSD MOUNTS THE SYSTEM DISK.
                      *
                      *       1) ABORT DRIVER
                      *       2) READ BLABEL RECORD
                      *       3) SET VOLUME NUMBER FOR DIRIVER
                      *       4) INITIALIZE DEVICE TABLE
                      *       5) BUILD GRT


046.155  076 007      MSD     MVI     A,DC.ABT
046.157  315 130 040          CALL    SYDD                    ABORT DRIVER
046.162  001 000 001          LXI     B,256
046.165  021 010 051          LXI     D,BLABEL
046.170  041 011 000          LXI     H,DDF.LAB
046.173  076 002              MVI     A,DC.RER                READ REGARDLESS
046.175  315 130 040          CALL    SYDD
046.200  334 304 045          CC      READERR                 BAD ERROR

                      *       CALL DEVICE MOUNT ROUTINE

046.203  072 010 051          LDA     BLABEL+LAB.SER
046.206  157                  MOV     L,A
046.207  046 000              MVI     H,0                     (HL) = SERIAL NUMBER
046.211  076 010              MVI     A,DC.MOU
046.213  315 130 040          CALL    SYDD                    MOUNT UNIT
046.216  334 304 045          CC      READERR                 BAD ERROR
046.221  311                  RET

046.222  000 000      CRCSUM  DW      0                       CRCSUM WORKAREA


046.224                       DS      64                      PATCH AREA

                      **      ALL CODE FOLLOWING MAY BE OVERLAID BY THE HDOS.SYS
                      *       PROGRAM DURING BOOT.

377.324                       ERRPL   *-SB.ORG                MUST BE BEFORE ORG ADDRESS

046.324                       XTEXT   ABR                     INCLUDE HERE TO BE OVERLAID
```

```
                             1644X **       ABR - AUTO BAUD RATE SELECTION.
                             1645X *
                             1646X *        ABR READS CHARACTERS FROM THE SYSTEM CONSOLE ACE UNTIL
                             1647X *        THE CURRENT BAUD RATE IS DETERMINED.
                             1648X *
                             1649X *        ENTRY   NONE
                             1650X *        EXIT    (HL) = BAUD RATE DIVISOR
                             1651X *                ACE SETUP WITH BAUD RATE, NO INTERRUPTS
                             1652X *        USES    ALL
                             1653X
                             1654X
       046.324               1655X ABR      EQU     *
                             1656X
                             1657X *        INITIALIZE LED DISPLAY FOR PROMPT
                             1658X
       046.324  072 010 040  1659X ABRO.1   LDA     .MFLAG                                        /79.01.GC/
       046.327  365          1660X          PUSH    PSW                                           /79.01.GC/
       046.330  366 002      1661X          ORI     UO.DDU                                        /79.01.GC/
       046.332  062 010 040  1662X          STA     .MFLAG                                        /79.01.GC/
       046.335  001 011 000  1663X          LXI     B,9
       046.340  021 155 047  1664X          LXI     D,ABR.A
       046.343  041 013 040  1665X          LXI     H,.ALEDS
       046.346  315 252 030  1666X          CALL    $MOVE
       046.351  021 013 040  1667X          LXI     D,.ALEDS
       046.354  076 144      1668X          MVI     A,100
       046.356  315 140 002  1669X          CALL    .HORN
                             1670X
       046.361  041 166 047  1671X          LXI     H,TABLE
                             1672X
       046.364  257          1673X ABRO.3   XRA     A                                             /79.01.GC/
       046.365  323 351      1674X          OUT     SC.ACE+UR.IER                                 /79.01.GC/
       046.367  076 020      1675X          MVI     A,UC.LOO                                      /79.01.GC/
       046.371  323 354      1676X          OUT     SC.ACE+UR.MCR   SET LOOP BACK                 /79.01.GC/
       046.373  076 200      1677X          MVI     A,UC.DLA
       046.375  323 353      1678X          OUT     SC.ACE+UR.LCR   LINE CONTROL ACCESS
       046.377  176          1679X          MOV     A,M
       047.000  043          1680X          INX     H
       047.001  323 350      1681X          OUT     SC.ACE+UR.DLL   DIVISOR LEAST SIGNIFICANT
       047.003  176          1682X          MOV     A,M
       047.004  346 177      1683X          ANI     177Q            CLEAR STOP BITS FLAG
       047.006  323 351      1684X          OUT     SC.ACE+UR.DLM   DIVISOR MOST SIGNIFICANT
       047.010  276          1685X          CMP     M               SEE IF 2 STOP BITS
       047.011  043          1686X          INX     H
       047.012  076 003      1687X          MVI     A,UC.8BW        ASSUME 8 BIT WORDS, 1 STOP
       047.014  312 021 047  1688X          JE      ABRO.5
       047.017  076 007      1689X          MVI     A,UC.8BW+UC.2SB SET 2 STOP BITS
       047.021  323 353      1690X ABRO.5   OUT     SC.ACE+UR.LCR   LINE CONTROL ACCESS
       047.023  076 156      1691X          MVI     A,AC.DLY                                      /79.01.GC/
       047.025  315 053 000  1692X          CALL    .DLY            WAIT FOR 8250 TO SETTLE       /79.01.GC/
       047.030  333 354      1693X          IN      SC.ACE+UR.MCR                                 /79.01.GC/
       047.032  346 357      1694X          ANI     377Q-UC.LOO                                   /79.01.GC/
       047.034  323 354      1695X          OUT     SC.ACE+UR.MCR   TURN OFF LOOP                 /79.01.GC/
                             1696X
                             1697X *        WAIT FOR CHARACTER TO BE HIT
                             1698X
       047.036  333 350      1699X ABRO     IN      SC.ACE+UR.RBR   GOBBLE OVERRUN
```

```
047.040  333 355      1700X ABR1     IN       SC.ACE+UR.LSR
000.000               1701X          ERRNZ    UC.OR-2
047.042  037          1702X          RAR
047.043  037          1703X          RAR
047.044  332 036 047  1704X          JC       ABR0              OVERRUN
047.047  027          1705X          RAL
047.050  027          1706X          RAL
047.051  346 015      1707X          ANI      UC.DR+UC.PE+UC.FE
047.053  312 040 047  1708X          JZ       ABR1              NOTHING YET
047.056  365          1709X          PUSH     PSW
047.057  032          1710X          LDAX     D                 ECHO ' ' AS '.' ON LEDS
047.060  346 177      1711X          ANI      01111111B         TURN ON '.'
047.062  022          1712X          STAX     D
047.063  023          1713X          INX      D
047.064  361          1714X          POP      PSW
047.065  346 010      1715X          ANI      UC.FE
047.067  302 107 047  1716X          JNZ      ABR3              USER IS SLOWER THAN THIS
047.072  333 350      1717X          IN       SC.ACE+UR.RBR     GET DATA
047.074  346 177      1718X          ANI      177Q              TRIM
047.076  376 040      1719X          CPI      ' '
047.100  312 124 047  1720X          JE       ABR5
                      1721X
                      1722X *        USER IS FASTER THAN WE ARE. FOLLOW FASTER LINKAGE
                      1723X
047.103  156          1724X ABR2     MOV      L,M               FOLLOW LINK
047.104  303 364 046  1725X          JMP      ABR0.3            TRY AGAIN
                      1726X
                      1727X *        USER IS SLOWER THAN WE ARE. READ NEXT CHARACTER
                      1728X
047.107  076 067      1729X ABR3     MVI      A,110/2
047.111  315 053 000  1730X          CALL     .DLY              WAIT FOR THINGS TO SETTLE OUT
047.114  333 350      1731X          IN       SC.ACE+UR.RBR
047.116  333 355      1732X          IN       SC.ACE+UR.LSR
047.120  043          1733X          INX      H
047.121  303 103 047  1734X          JMP      ABR2
                      1735X
                      1736X *        FOUND THE BAUD RATE. RETURN WITH ANSWERS
                      1737X
047.124  021 013 040  1738X ABR5     LXI      D,.ALEDS                                  BLANK DISPLAY
047.127  006 011      1739X          MVI      B,9
047.131  076 377      1740X          MVI      A,377A
047.133  022          1741X ABR5.1   STAX     D
047.134  023          1742X          INX      D
047.135  005          1743X          DCR      B
047.136  302 133 047  1744X          JNZ      ABR5.1
047.141  053          1745X          DCX      H
047.142  126          1746X          MOV      D,M
047.143  053          1747X          DCX      H
047.144  136          1748X          MOV      E,M
047.145  353          1749X          XCHG                       (HL) = BAUD RATE
047.146  333 350      1750X          IN       SC.ACE+UR.RBR     GOBBLE THE GARBAGE                /79.01.GC/
047.150  361          1751X          POP      PSW                                                 /79.01.GC/
047.151  062 010 040  1752X          STA      .MFLAG                                              /79.01.GC/
047.154  311          1753X          RET
                      1754X
047.155  244 230 220  1755X ABR.A    DB       244Q,230Q,220Q,215Q,214Q,377Q,377Q,377Q,377Q    'SPACE' FOR LEDS
```

```
                          1756X


                          1758X **       BAUD RATE SELECTION TREE.
                          1759X *
        047.166           1760X TABLE    DS      0              START OF BAUD TABLE
                          1761X
        047.166  060 000  1762X          DW      000060A        2400 BAUD
        047.170  172      1763X          DB      #T9600         USER IS FASTER
        047.171  176      1764X          DB      #T600          USER IS SLOWER
                          1765X
                          1766X *        2ND TRY GROUPS
                          1767X
        047.172  014 000  1768X T9600    DW      000014A        9600 BAUD
        047.174  202      1769X          DB      #T19200        USER IS FASTER
        047.175  206      1770X          DB      #T4800         USER IS SLOWER
                          1771X
        047.176  300 000  1772X T600     DW      000300A        600 BAUD
        047.200  212      1773X          DB      #T1200         USER IS FASTER
        047.201  216      1774X          DB      #T300          USER IS SLOWER
                          1775X
                          1776X *        3RD TRY GROUPS
                          1777X
        047.202  006 000  1778X T19200   DW      000006A        19200 BAUD
        047.204  166      1779X          DB      #TABLE         USER IS FASTER, MUST BE SCREWED UP
        047.205  166      1780X          DB      #TABLE         USER IS SLOWER, MUST BE SCREWED UP
                          1781X
        047.206  030 000  1782X T4800    DW      000030A        4800 BAUD
        047.210  222      1783X          DB      #T7200         USER IS FASTER
        047.211  226      1784X          DB      #T3600         USER IS SLOWER
                          1785X
        047.212  140 000  1786X T1200    DW      000140A        1200 BAUD
        047.214  232      1787X          DB      #T1800         USER IS FASTER
        047.215  166      1788X          DB      #TABLE         USER IS SLOWER, MUST BE SCREWED UP
                          1789X
        047.216  200 001  1790X T300     DW      001200A        300 BAUD
        047.220  166      1791X          DB      #TABLE         USER IS FASTER, MUST BE SCREWED UP
        047.221  236      1792X          DB      #T110          USER IS SLOWER
                          1793X
                          1794X *        4TH TRY GROUPS
                          1795X
        047.222  020 000  1796X T7200    DW      000020A        7200 BAUD
        047.224  166      1797X          DB      #TABLE         USER IS FASTER, MUST BE SCREWED UP
        047.225  166      1798X          DB      #TABLE         USER IS SLOWER, MUST BE SCREWED UP
                          1799X
        047.226  040 000  1800X T3600    DW      000040A        3600 BAUD
        047.230  166      1801X          DB      #TABLE         USER IS FASTER, MUST BE SCREWED UP
        047.231  166      1802X          DB      #TABLE         USER IS SLOWER, MUST BE SCREWED UP
                          1803X
        047.232  100 000  1804X T1800    DW      000100A        1800 BAUD
        047.234  166      1805X          DB      #TABLE         USER IS FASTER, MUST BE SCREWED UP
        047.235  166      1806X          DB      #TABLE         USER IS SLOWER, MUST BE SCREWED UP
                          1807X
        047.236  027 204  1808X T110     DW      204027A        110 BAUD
```

```
   047.240  242          1809X          DB      #T150          USER IS FASTER
   047.241  246          1810X          DB      #T75           USER IS SLOWER
                         1811X
                         1812X *        5TH TRY GROUPS
                         1813X
   047.242  000 003      1814X T150     DW      003000A        150 BAUD
   047.244  166          1815X          DB      #TABLE         USER IS FASTER, MUST BE SCREWED UP
   047.245  166          1816X          DB      #TABLE         USER IS SLOWER, MUST BE SCREWED UP
                         1817X
   047.246  000 006      1818X T75      DW      006000A        75 BAUD
   047.250  166          1819X          DB      #TABLE         USER IS FASTER, MUST BE SCREWED UP
   047.251  166          1820X          DB      #TABLE         USER IS SLOWER, MUST BE SCREWED UP
                         1821X
   000.047               1822X .        SET     */256
   000.000               1823X          ERRNZ   TABLE/256-.    MUST BE IN SAME PAGE
   047.252               1824           XTEXT   SCU


                         1826X **       SCU - SETUP CONSOLE USART.
                         1827X *
                         1828X *        SCU CONFIGURES THE CONSOLE USART.
                         1829X *
                         1830X *        IF  8250
                         1831X *          THEN  PORT = 372-3Q
                         1832X *          ELSE  PORT = 340-7Q
                         1833X *
                         1834X *
                         1835X *        ENTRY   NONE
                         1836X *        EXIT    NONE
                         1837X *        USES    A,F,(BC),(HL)
                         1838X
                         1839X
   047.252  072 343 040  1840X SCU      LDA     S.CDB
   047.255  376 001      1841X          CPI     CDB.H84
   047.257  312 322 047  1842X          JZ      SCU1                     IF  8250
                         1843X
                         1844X *        PRESET 8251
                         1845X
   047.262  076 201      1846X          MVI     A,201Q
   047.264  323 373      1847X          OUT     SC.UART+USR              GET USART IN KNOWN STATE
   047.266  323 373      1848X          OUT     SC.UART+USR
   047.270  323 373      1849X          OUT     SC.UART+USR
   047.272  323 373      1850X          OUT     SC.UART+USR
   047.274  076 100      1851X          MVI     A,UCI.IR                 RESET
   047.276  323 373      1852X          OUT     SC.UART+USR
   047.300  072 327 040  1853X          LDA     S.CONTY
   047.303  346 010      1854X          ANI     CTP.2SB
   000.000               1855X          ERRNZ   CTP.2SB*16+UMI.1B-UMI.2B
   047.305  007          1856X          RLC
   047.306  007          1857X          RLC
   047.307  007          1858X          RLC
   047.310  007          1859X          RLC
   047.311  366 116      1860X          ORI     UMI.1B+UMI.L8+UMI.16X
   047.313  323 373      1861X          OUT     SC.UART+USR
```

```
047.315  076 025              MVI    A,UCI.ER+UCI.RE+UCI.TE
047.317  323 373              OUT    SC.UART+USR
047.321  311                  RET

                *             IS 8250

047.322  333 355        SCU1  IN     SC.ACE+UR.LSR                                  /80.01.GC/
047.324  346 100              ANI    UC.TSE          CHECK FOR SHIFT EMPTY          /80.01.GC/
047.326  312 322 047          JZ     SCU1                                           /80.01.GC/

047.331  257                  XRA    A                                              /79.01.GC/
047.332  323 351              OUT    SC.ACE+UR.IER   TURN OFF ANY INTERRUPTS        /79.01.GC/
047.334  076 020              MVI    A,UC.LOO                                       /79.01.GC/
047.336  323 354              OUT    SC.ACE+UR.MCR                                  /79.01.GC/
047.340  052 344 040          LHLD   S.BAUD
047.343  076 200              MVI    A,UC.DLA
047.345  323 353              OUT    SC.ACE+UR.LCR   ACCESS DIVISOR LATCHES
047.347  175                  MOV    A,L
047.350  323 350              OUT    SC.ACE+UR.DLL   SET LEAST SIGNIFICANT
047.352  174                  MOV    A,H
047.353  346 177              ANI    177Q            TRIM STOP BITS
047.355  323 351              OUT    SC.ACE+UR.DLM   SET MOST SIGNIFICANT
047.357  072 327 040          LDA    S.CONTY
047.362  346 010              ANI    CTP.2SB
047.364  017                  RRC
000.000                       ERRNZ  CTP.2SB/2-UC.2SB
000.000                       ERRNZ  UC.2SB-4        (A) = UC.2SB IF 2 STOP BITS
047.365  366 003              ORI    UC.8BW          8 BIT WORDS
047.367  323 353              OUT    SC.ACE+UR.LCR
047.371  076 156              MVI    A,AC.DLY                                       /79.01.GC/
047.373  315 053 000          CALL   .DLY                                           /79.01.GC/
047.376  333 350              IN     SC.ACE+UR.RBR   GOBBLE ANY TRASH               /79.01.GC/
050.000  333 354              IN     SC.ACE+UR.MCR                                  /79.01.GC/
050.002  346 357              ANI    377Q-UC.LOO                                    /79.01.GC/
050.004  323 354              OUT    SC.ACE+UR.MCR                                  /79.01.GC/
050.006  311                  RET
050.007                       XTEXT  MCU             INCLUDED HERE TO BE USED AT BOOT-UP



                **            MCU - MAP LOWER CASE TO UPPER CASE.
                *
                *             MCU MAPS A LOWER CASE ALPHABETIC TO UPPER
                *             CASE.
                *
                *             ENTRY  (A) = CHARACTER
                *             EXIT   (A) = CHARACTER RESULT
                *             USES   A,F


050.007  376 141        $MCU  CPI    'a'
050.011  330                  RC                     NOT LOWER CASE
050.012  376 173              CPI    'z'+1
050.014  320                  RNC                    NOT LOWER CASE
050.015  326 040              SUI    'a'-'A'
```

```
     050.017  311          1915X          RET
```

```
                         **      THIS ROUTINE IS OVERLAID BY THE HDOS.SYS PRORGRAM DURING BOOT.

                         **      CDC - COMPUTE DISK CHECKSUMS.
                         *
                         *       CDC READS EACH DISK SECTOR (EXCEPT FOR TRACK 0) TO SEE IF
                         *       THERE ARE ANY PROBLEMS.
                         *
                         *       THE CHECKSUM OF EACH SECTOR IS PRINTED, TOGETHER
                         *       WITH THE TOTAL CHECKSUM FOR EACH TRACK, AND THE VOLUME CHECKSUM.
                         *


050.020  041 000 000     CDC     LXI     H,0
050.023  042 000 051             SHLD    CDCA            VOLUME
050.026  042 002 051             SHLD    CDCB            TRACK
050.031  044                     INR     H
050.032  042 006 051             SHLD    CDCE            SET TRACK AND SECTOR NUMBERS
050.035  041 012 000             LXI     H,10
050.040  042 004 051             SHLD    CDCD            SET FIRST SECTOR NUMBER

                         *       READ TRACK

050.043  052 004 051     CDC1    LHLD    CDCD
050.046  021 010 052             LXI     D,BUFF
050.051  001 000 012             LXI     B,10*256
050.054  076 000                 MVI     A,DC.REA
050.056  315 130 040             CALL    SYDD            READ DISK
050.061  322 110 050             JNC     CDC2            NO ERROR

                         *       READ ERROR

050.064  315 055 045             CALL    $TYPET
050.067  007 077 060             DB      BELL,'?00 * ERROR * ',BELL,200Q

                         *       CRC SECTOR

050.110  315 055 045     CDC2    CALL    $TYPET
050.113  123 105 103             DB      'SECTOR',' '+200Q
050.122  052 004 051             LHLD    CDCD
050.125  353                     XCHG
050.126  076 003                 MVI     A,3
050.130  315 064 046             CALL    TTDD
050.133  315 055 045             CALL    $TYPET
050.136  040 075 240             DB      ' =',' '+200Q
050.141  052 002 051             LHLD    CDCB
050.144  072 006 051             LDA     CDCE
050.147  315 174 045             CALL    CSC             COMPUTE TRACK CRC
050.152  042 002 051             SHLD    CDCB
050.155  052 000 051             LHLD    CDCA
050.160  072 006 051             LDA     CDCE
050.163  315 174 045             CALL    CSC             COMPUTE VOLUME CRC
050.166  042 000 051             SHLD    CDCA
050.171  041 000 000             LXI     H,0
050.174  072 006 051             LDA     CDCE
050.177  315 174 045             CALL    CSC             COMPUTE SECTOR CRC
050.202  353                     XCHG
```

```
050.203  076 005                 MVI     A,5
050.205  315 145 046             CALL    TTDDCR          TYPE SECTOR CHECKSUM
050.210  052 004 051             LHLD    CDCD            (HL) = SECTOR COUNT
050.213  043                     INX     H
050.214  042 004 051             SHLD    CDCD
050.217  041 006 051             LXI     H,CDCE
050.222  315 271 044             CALL    $ICTT.          CHECK FOR CHARACTER
050.225  332 240 050             JC      CDC3            NO CHARACTER WAS HIT
050.230  315 317 044             CALL    $ICTT..         GET CHARACTER
050.233  376 003                 CPI     CTLC
050.235  312 367 050             JE      CDC4            CTL-C HIT
050.240  064             CDC3    INR     M               COUNT SECTOR
050.241  076 012                 MVI     A,10
050.243  226                     SUB     M
050.244  302 110 050             JNE     CDC2            MORE ON TRACK

                         *       HAVE COMPLETED TRACK

050.247  167                     MOV     M,A
050.250  043                     INX     H
050.251  064                     INR     M               COUNT TRACK
050.252  315 055 045             CALL    $TYPET
050.255  124 122 101             DB      'TRACK TOTAL ',200Q+' '
050.272  052 002 051             LHLD    CDCB
050.275  353                     XCHG
050.276  041 000 000             LXI     H,0
050.301  042 002 051             SHLD    CDCB            RESET COUNT
050.304  076 005                 MVI     A,5
050.306  315 145 046             CALL    TTDDCR

050.311  052 004 051             LHLD    CDCD
050.314  001 160 376             LXI     B,-400
050.317  011                     DAD     B
050.320  322 043 050             JNC     CDC1            MORE TO GO

                         *       ALL DONE

050.323  315 055 045             CALL    $TYPET
050.326  000 040 126             DB      0,' VOLUME TOTAL CRC =',200Q+' '
050.353  052 000 051             LHLD    CDCA
050.356  353                     XCHG
050.357  076 005                 MVI     A,5
050.361  315 145 046             CALL    TTDDCR
050.364  303 211 042             JMP     SOBOOTX

                         *       CTL-C STRUCK

050.367  315 055 045     CDC4    CALL    $TYPET
050.372  136 103 200             DB      '^C',200Q
050.375  303 211 042             JMP     SOBOOTX

051.000  000 000         CDCA    DW      0               VOLUME CRC
051.002  000 000         CDCB    DW      0               TRACK CRC
051.004  000 000         CDCD    DW      0               SECTOR NUMBER
051.006  000             CDCE    DB      0               SECTOR NUMBER
051.007  000                     DB      0               TRACK NUMBER
```

```
    007.000             2032  SOBOOTL EQU     *-SOBOOT+255/256*256     SOBOOT LENGTH IN SECTORS *256
                        2033
    051.010             2034  BLABEL  EQU     *               LABEL BUFFER STARTS HERE
    052.010             2035  BUFF    EQU     BLABEL+256      10 SECTOR BUFFER
    064.010             2036  BOOEND  EQU     10*256+BUFF     BUFFER ENDS HERE
```

```
                          2039  ***      INIT - MAIN INITIALIZE LOOP.
                          2040  *
                          2041
                          2042           LOF      C                   RESTORE LISTING CONTROL
                          2043
051.010                   2044  INIT     EQU      *
051.010  377 011          2045           DB       SYSCALL,.VERS
051.012  332 125 051      2046           JC       INIT1               NO .VERS SYSTEM CALL
051.015  376 026          2047           CPI      VERS
051.017  302 125 051      2048           JNZ      INIT1               VERSIONS DO NOT MATCH
051.022  076 377          2049           MVI      A,377Q
051.024  377 046          2050           DB       SYSCALL,.CLOSE      CLOSE THE CHANNEL WE CAME IN ON
051.026  257              2051           XRA      A
051.027  062 326 040      2052           STA      S.CSLMD             SETUP CONSOLE MODE
051.032  315 240 062      2053           CALL     $DOS                DISMOUNT OPERATING SYSTEM
051.035  332 136 051      2054           JC       ERROR               ERROR
051.040  315 172 051      2055           CALL     PRS                 PRESET SYSTEM, INITIAL MESSAGES
051.043  041 000 072      2056           LXI      H,RMEML
051.046  377 052          2057           DB       SYSCALL,.SETTP      SET TOP
051.050  332 136 051      2058           JC       ERROR
051.053  303 064 051      2059           JMP      INIT0               DONT ASK  IF MORE FOR THE FIRST TIME
                          2060
                          2061  *        RESTART HERE TO INIT NEW DISK
                          2062
051.056  315 317 052      2063  RESTART  CALL     AMW                 ASK IF MORE WORK WANTED
051.061  302 133 051      2064           JNZ      EXIT                ALL DONE
051.064  061 200 042      2065  INIT0    LXI      SP,STACK
051.067  257              2066           XRA      A
051.070  062 061 041      2067           STA      AIO.UNI             SELECT UNIT 0
051.073  315 104 053      2068           CALL     RMI                 REQUEST MEDIA INSERTION
051.076  076 007          2069           MVI      A,DC.ABT
051.100  315 343 063      2070           CALL     SYDD.               RESET DISK
051.103  315 230 054      2071           CALL     AAL                 ASK ABOUT LABEL
051.106  315 224 055      2072           CALL     GVI                 GET VOLUME ID
051.111  315 101 056      2073           CALL     IDS                 INIT DISK SURFACE
051.114  315 346 056      2074           CALL     GBL                 GET BAD SECTOR LIST
051.117  315 013 060      2075           CALL     FOV                 FORMAT VOLUME
051.122  303 056 051      2076           JMP      RESTART
                          2077
051.125  076 050          2078  INIT1    MVI      A,EC.NCV            NOT CORRECT VERSION
051.127  067              2079           STC
051.130  303 136 051      2080           JMP      ERROR
                          2081
                          2082  *        EXIT
                          2083
051.133  257              2084  EXIT     XRA      A
051.134  377 000          2085  EXIT.    DB       SYSCALL,.EXIT       LET  *HDOS*  HANDLE THE EXIT PARAMETERS
```

```
                              2087  **       ERROR - ERROR ENCOUNTERED.
                              2088
       051.136  365           2089  ERROR    PUSH     PSW
       051.137  315 075 062   2090           CALL     $CCO             CLEAR CTL-O
       051.142  315 136 031   2091           CALL     $TYPTX
       051.145  012 007 105   2092           DB       NL,BELL,'ERROR - ',' '+200Q
       051.160  361           2093           POP      PSW
       051.161  046 012       2094           MVI      H,NL
       051.163  377 057       2095           DB       SYSCALL,.ERROR
       051.165  076 001       2096           MVI      A,1              FLAG ERROR
       051.167  303 134 051   2097           JMP      EXIT.
```

```
                     **      PRS - PRESET PROGRAM.
                     *
                     *       PRS PERFORMS ANY INITIALIZATION TASKS, AND  INFORMS THE
                     *       USER OF THE FACTS OF LIFE CONCERNING INITIALIZATION/.
                     *
                     *       ENTRY   NONE
                     *       EXIT    TO SYSTEM IF USER CHICKENS OUT
                     *               TO CALLER IF OK
                     *       USES    ALL


051.172 315 075 062  PRS     CALL    $CCO              CLEAR CTL-O
051.175 315 136 031          CALL    $TYPTX
051.200 012 011 011          DB      NL,TAB,TAB,TAB,'    ','INIT'
051.214 012 011 011          DB      NL,TAB,TAB,TAB,'Version:  ',VERS/16+'0','.',VERS&00001111B+'0'
051.235 012 011 011          DB      NL,TAB,TAB,'      ','Issue:  #50.05.00'
051.267 012                  DB      NL
051.270 012 011 124          DB      NL,TAB,'This  routine is used to initialize HDOS floppy'
051.351 040 144 151          DB      ' disks.'
051.360 012 111 164          DB      NL,'It is  a  stand-alone utility,  and  will destroy any'
052.046 040 146 151          DB      ' files on'
052.057 012 164 150          DB      NL,'the disks it initializes.  Do not attempt to use this'
052.145 040 040 160          DB      '  program'
052.156 012 165 156          DB      NL,'until you have studied the appropriate manual.'
052.235 212                  DB      ENL
052.236 315 075 062          CALL    $CCO              CLEAR CTL-O
052.241 315 136 031          CALL    $TYPTX
052.244 012 120 162          DB      NL,'Proceed (YES/NO) <NO> ','?'+200Q
052.274 315 205 063          CALL    $ITL.
052.277 176                  MOV     A,M
052.300 247                  ANA     A
052.301 312 133 051          JZ      EXIT              WANTS TO EXIT
052.304 376 116              CPI     'N'
052.306 312 133 051          JE      EXIT              EXIT
052.311 376 131              CPI     'Y'
052.313 302 172 051          JNE     PRS
052.316 311                  RET
```

```
                          2141  **      AMW - ASK FOR MORE WORK.
                          2142  *
                          2143  *       AMW SEES IF THE USER WANTS TO CONTINUE THE PROCESS.
                          2144  *
                          2145  *       ENTRY   NONE
                          2146  *       EXIT    'Z' CLEAR IF  NO MORE WORK
                          2147  *               'Z' SET IF MORE WORK
                          2148  *       USES    ALL
                          2149
                          2150
052.317  315 136 031      2151  AMW     CALL    $TYPTX
052.322  012 040 104      2152          DB      NL,' Disk Initialization complete.',ENL
                          2153
052.362  315 075 062      2154  AMW1    CALL    $CCO                CLEAR CTL-O
052.365  315 136 031      2155          CALL    $TYPTX
052.370  012 104 157      2156          DB      NL,'Do you have any more disks to initialize (YES/NO) <NO> ','?'+200Q
053.061  315 205 063      2157          CALL    $ITL.
053.064  176              2158          MOV     A,M                 (A) = REPLY
053.065  376 131          2159          CPI     'Y'
053.067  310              2160          RE                          IS YES
053.070  247              2161          ANA     A
053.071  312 101 053      2162          JZ      AMW2                IS NO
053.074  376 116          2163          CPI     'N'
053.076  302 362 052      2164          JNE     AMW1                ASK AGAIN
053.101  366 001          2165  AMW2    ORI     1                   ANSWER IS NO
053.103  311              2166          RET
```

```
                      **      RMI - REQUEST MEDIA INSERTION.
                      *
                      *       RMI PROMPTS THE USR TO LOAD THE DESIRED MEDIA.
                      *
                      *       WHEN IN PLACE, RMI CHECKS THE HOLE TIMINGS
                      *
                      *       ENTRY   NONE
                      *       EXIT    DONE
                      *       USES    ALL


053.104  315 075 062  RMI     CALL    $CCO
053.107  315 136 031          CALL    $TYPTX
053.112  012 040 111          DB      NL,' Insert the volume you wish to initialize into SY0:;'
053.177  012 040 162          DB      NL,' remember, any data on this volume will be destroyed.'
053.265  012                  DB      NL
053.266  012 110 151          DB      NL,'Hit RETURN when ready.'
053.315  012 122 145          DB      NL,'Ready?',' '+200Q

053.325  315 213 063          CALL    $ITL

                      *       GOT HIS OK. CHECK PATTERN.

053.330  076 022              MVI     A,DF.DS0+DF.MO
053.332  323 177              OUT     DP.DC               ON DRIVE
053.334  315 124 054          CALL    W1S                 WAIT ONE SECOND
053.337  315 271 036          CALL    R.WNH               WAIT FOR NO HOLE
053.342  315 235 036          CALL    R.WHD               WAIT FOR HOLE DETECT
053.345  315 271 036          CALL    R.WNH               WAIT FOR NO HOLE DETECT

                      *       CHECK INSERTED MEDIA

053.350  315 136 054          CALL    CIM                 CHECK INSERTED MEDIA
053.353  320                  RNC                         MEDIA IS GOOD
053.354  315 136 054          CALL    CIM                 GIVE MEDIA A SECOND CHECK
053.357  320                  RNC                         WAS GOOD THE SECOND TIME

                      *       ERROR IN MEDIA FORMAT.

053.360  315 136 031          CALL    $TYPTX
053.363  012 007 040          DB      NL,BELL,' Wrong type of media, media inserted improperly, or'
054.050  012 040 155          DB      NL,' media damaged. Check it and try again.',ENL
054.121  303 104 053          JMP     RMI



                      **      W1S - WAIT ONE SECOND.
                      *
                      *       W1S IS CALLED TO DELAY ONE SECOND.


054.124  076 372      W1S     MVI     A,250
054.126  315 303 035          CALL    R.DLY
054.131  076 372              MVI     A,250
054.133  303 303 035          JMP     R.DLY               DELAY AND EXIT
```

```
                         **        CIM    -  CHECK INSERTED MEDIA
                         *
                         *         CIM CHECKS THE INSERTED MEDIA
                         *
                         *
                         *         ENTRY   NONE
                         *
                         *         EXIT    (PSW)   = 'C' CLEAR IF GOOD MEDIA
                         *                         = 'C' SET   IF BAD  MEDIA
                         *

000.002                  CIM.DLY EQU       2                        NUMBER OF SECONDS ALLOWED FOR CHECK
000.334                  CIM.CNT EQU       300/60*11*CIM.DLY*2      300 RPM, 60 SEC./MIN., 11 HOLES/TRACK
000.320                  CIM.MIN EQU       1000*11*CIM.DLY/21/10*2 MIN TOLERANCE COUNT = 21 MIL. SEC. GAP
000.346                  CIM.MAX EQU       1000*11*CIM.DLY/19/10*2 MAX TOLERANCE COUNT = 19 MIL. SEC. GAP

054.136  001 000 000     CIM      LXI      B,0                 ZERO TRANSITION COUNT
054.141  052 033 040              LHLD     .TICCNT             GET CURRENT TIC
054.144  021 350 003              LXI      D,CIM.DLY*1000/2          SET DELAY
054.147  031                      DAD      D
054.150  315 224 030              CALL     $CHL                (HL) = -(HL)
054.153  124                      MOV      D,H
054.154  135                      MOV      E,L                 (DE) = -(TARGET STOP TIME)

                         *        INITIALIZE HOLE DETECT FLAG

054.155  333 177                  IN       DP.DC               DISK CONTROL PORT
054.157  346 001                  ANI      DF.HD               HOLE DETECT
054.161  062 227 054              STA      CIMA                SET INITIAL VALUE

                         *        COUNT THE NUMBER OF TRANSITIONS IN CIM.DLY SECONDS

054.164  041 227 054     CIM1     LXI      H,CIMA              'M' POINTS TO THE FLAG BYTE
054.167  333 177                  IN       DP.DC               DISK CONTROL PORT
054.171  346 001                  ANI      DF.HD               HOLE DETECT
054.173  276                      CMP      M
054.174  312 201 054              JZ       CIM2                NO TRANSITION DETECTED
054.177  003                      INX      B                   COUNT TRANSITION
054.200  167                      MOV      M,A
054.201  052 033 040     CIM2     LHLD     .TICCNT
054.204  031                      DAD      D
054.205  174                      MOV      A,H
054.206  247                      ANA      A
054.207  372 164 054              JM       CIM1                IT IS NOT TIME TO STOP YET

                         *        COMPARE THE COUNT TO EXPECTED VALUES

054.212  170                      MOV      A,B
054.213  376 000                  CPI      CIM.CNT/256
000.000                  ..       SET      CIM.CNT/256
000.000                           ERRNZ    CIM.MIN/256-..
000.000                           ERRNZ    CIM.MAX/256-..
054.215  067                      STC                          ASSUME BAD VALUE
054.216  300                      RNZ                          NOT A GOOD VALUE
054.217  171                      MOV      A,C
054.220  376 320                  CPI      #CIM.MIN
```

```
054.222  330          2279          RC                  LESS THAN THE MINIMUM ACCEPTABLE COUNT VALUE
054.223  376 347      2280          CPI     #CIM.MAX+1
054.225  077          2281          CMC
054.226  311          2282          RET
                      2283
054.227  000          2284  CIMA    DB      0
```

```
                              2287  **      AAL - ASK ABOUT LABEL.
                              2288  *
                              2289  *       AAL ATTEMPTS TO READ THE VOLUME LABEL RECORD. IF SUCCESSFUL, THE
                              2290  *       VOLUME # AND LABEL ARE PRINTED. THE USER GETS A CHANCE TO
                              2291  *       CHICKEN OUT.
                              2292  *
                              2293  *       ENTRY   NONE
                              2294  *       EXIT    NONE
                              2295  *       USES    ALL
                              2296
                              2297
054.230  257                  2298  AAL     XRA     A
054.231  062 252 040          2299          STA     D.DRVTB+1                    CLEAR VOLUME NUMBER
054.234  315 136 031          2300          CALL    $TYPTX
054.237  012 124 150          2301          DB      NL,'The volume now in the drive ...',ENL
054.300  021 000 071          2302          LXI     D,RGTAB         USE RGTAB FOR SCRATCH BUFFER
054.303  001 000 001          2303          LXI     B,1000A
054.306  041 011 000          2304          LXI     H,DDF.LAB
054.311  076 002              2305          MVI     A,DC.RER        READ REGARDLESS
054.313  315 130 040          2306          CALL    SYDD            READ LABEL SECTOR
054.316  322 002 055          2307          JNC     AAL1            OK
                              2308
                              2309  *       COULDNT READ IT
                              2310
054.321  315 136 031          2311          CALL    $TYPTX
054.324  101 160 160          2312          DB      'Apparently has not been initialized before',ENL
054.377  303 066 055          2313          JMP     AAL2
                              2314
                              2315  *       COULD READ IT. TYPE DISK NUMBER AND LABEL
                              2316
055.002  315 136 031          2317  AAL1    CALL    $TYPTX
055.005  151 163 040          2318          DB      'is volume #',' '+200Q
055.021  072 000 071          2319          LDA     RGTAB+LAB.SER
055.024  137                  2320          MOV     E,A
055.025  026 000              2321          MVI     D,0
055.027  076 003              2322          MVI     A,3
055.031  315 223 063          2323          CALL    $TDD            TYPE DECIMAL DIGITS
055.034  315 136 031          2324          CALL    $TYPTX
055.037  012 040 114          2325          DB      NL,' Label = ','"'+200Q
055.052  076 074              2326          MVI     A,LAB.LBL
055.054  041 021 071          2327          LXI     H,RGTAB+LAB.LAB
055.057  315 112 063          2328          CALL    $TYPL.          TYPE LABEL
055.062  315 136 031          2329          CALL    $TYPTX
055.065  242                  2330          DB      '"'+200Q
                              2331
                              2332  *       GIVE CHANCE TO CHICKEN OUT
                              2333
055.066  315 075 062          2334  AAL2    CALL    $CCO            CLEAR CTL-O
055.071  315 136 031          2335          CALL    $TYPTX
055.074  012 124 171          2336          DB      NL,'Type NO to cancel, type YES to erase and initialize the disk. (YES/NO) ','?'+200
Q
055.205  315 205 063          2337          CALL    $ITL.
055.210  176                  2338          MOV     A,M             (A) = REPLY
055.211  376 116              2339          CPI     'N'
055.213  312 056 051          2340          JE      RESTART
055.216  376 131              2341          CPI     'Y'
055.220  302 066 055          2342          JNE     AAL2
```

```
   055.223  311           2343        RET                             GOT THE OK
```

```
                            **      GVI - GET VOLUME ID.
                            *
                            *       GVI GETS A NEW VOLUME SERIAL NUMBER AND AN OPTIONAL VOLUME LABEL.
                            *
                            *       ENTRY   NONE
                            *       EXIT    LABEL+LAB.SER = VOLUME SERIAL #
                            *               LABEL+LAB.LAB = 60 CHARACTER VOLUME LABEL
                            *       USES    ALL


055.224                     GVI     EQU     *
055.224  315 075 062                CALL    $CCO                CLEAR CTL-O
055.227  315 136 031                CALL    $TYPTX
055.232  012 105 156                DB      NL,'Enter a unique volume serial number from 1 to 255:',' '+200Q
055.316  315 134 063                CALL    $IDN                INPUT DECIMAL NUMBER
055.321  332 224 055                JC      GVI                 BAD RANGE
055.324  174                        MOV     A,H
055.325  267                        ORA     A
055.326  302 224 055                JNZ     GVI                 TOO LARGE
055.331  265                        ORA     L
055.332  312 224 055                JZ      GVI                 ZERO ILLEGAL
055.335  062 074 064                STA     LABEL+LAB.SER

                            *       GET LABEL

055.340  315 075 062        GVI1    CALL    $CCO                CLEAR CTL-O
055.343  315 136 031                CALL    $TYPTX
055.346  105 156 164                DB      'Enter a volume label of 60 characters or less',ENL
056.024  315 213 063                CALL    $ITL
056.027  353                        XCHG                        (DE) = LINE ADDRESS
056.030  041 115 064                LXI     H,LABEL+LAB.LAB (HL) = TO ADDRESS

                            *       COPY FROM LINE TO LABEL+LAB.LAB, BLANK FILL OUT TO 60 CHARACTERS

056.033  006 075                    MVI     B,61
056.035  032                GVI2    LDAX    D
056.036  167                        MOV     M,A                 COPY
056.037  023                        INX     D
056.040  247                        ANA     A
056.041  312 071 056                JZ      GVI3                END OF LINE
056.044  043                        INX     H                   INCREMENT TO POINTER
056.045  005                        DCR     B
056.046  302 035 056                JNZ     GVI2                MORE TO TO
056.051  315 136 031                CALL    $TYPTX
056.054  124 157 157                DB      'Too Long.',ENL
056.066  303 340 055                JMP     GVI1

056.071  066 040            GVI3    MVI     M,' '               BLANK REST OF LABEL
056.073  043                        INX     H
056.074  005                        DCR     B
056.075  302 071 056                JNZ     GVI3
056.100  311                        RET
```

```
                         **      IDS - INITIALIZE DISK SURFACE.
                         *
                         *       IDS IS CALLED TO INITIALIZE THE DISK SURFACE WITH RECORD LABELS,
                         *       AND A TEST PATTERN.
                         *
                         *       ENTRY   MAXTRK = MAX TRACK NUMBER
                         *       EXIT    NONE
                         *       USES    ALL


056.101  076 007         IDS     MVI     A,DC.ABT
056.103  315 343 063             CALL    SYDD.                   RESET DEVICE
056.106  257                     XRA     A
056.107  062 172 056             STA     IDSA                    SET VOLUME NUMBER
056.112  323 175                 OUT     UP.FC                   SET FILL CHARACTER
056.114  107                     MOV     B,A                     (B) = TRACK NUMBER
056.115  117                     MOV     C,A                     (C) = SECTOR NUMBER

                         *       ERASE THIS TRACK

056.116  373             IDS1    EI
056.117  076 022                 MVI     A,DF.DS0+DF.MO
056.121  323 177                 OUT     DP.DC                   START THINGS
056.123  062 242 040             STA     D.DVCTL                 SET DEVICE CONTROL
056.126  076 043                 MVI     A,70/2
056.130  315 303 035             CALL    R.DLY                   WAIT HEAD SETTLE
056.133  076 023                 MVI     A,DF.DS0+DF.MO+DF.WG
056.135  323 177                 OUT     DP.DC                   ON WRITE GATE
056.137  076 175                 MVI     A,250/2
056.141  315 303 035             CALL    R.DLY                   LET TRACK ERASE
056.144  315 034 064             CALL    WIH                     WAIT INDEX HOLE
056.147  363                     DI                              DISABLE INTERRUPTS

                         *       WRITE 10 SECTOR HEADERS ON THIS TRACK

056.150  076 377         IDS2    MVI     A,377Q
056.152  062 243 040             STA     D.DLYMO                 LEAVE MOTOR ON
056.155  315 235 036             CALL    R.WHD                   WAIT HOLE DETECT
056.160  076 001                 MVI     A,1
056.162  141                     MOV     H,C                     SAVE (C) IN H
056.163  016 012                 MVI     C,10
056.165  315 224 040             CALL    D.WSP                   WRITE SYNC PATTERN
056.170  114                     MOV     C,H                     RESTORE (C)
056.171  076 000                 MVI     A,0
056.172                  IDSA    EQU     *-1                     TRACK SERIAL
056.173  315 227 040             CALL    D.WNB                   WRITE BYTE
056.176  170                     MOV     A,B                     (A) = TRACK
056.177  315 227 040             CALL    D.WNB
056.202  171                     MOV     A,C                     (A) = SECTOR
056.203  315 227 040             CALL    D.WNB
056.206  172                     MOV     A,D
056.207  315 227 040             CALL    D.WNB                   WRITE CHECK BYTE
056.212  257                     XRA     A
056.213  315 227 040             CALL    D.WNB                   WRITE BYTE
056.216  076 001                 MVI     A,1
056.220  141                     MOV     H,C                     SAVE (C) IN H
```

```
056.221  016 012      2456          MVI     C,10
056.223  315 224 040  2457          CALL    D.WSP             WRITE SYNC PATTERN
056.226  114          2458          MOV     C,H               RESTORE (C)
056.227  046 200      2459          MVI     H,128
056.231  076 107      2460  IDS3    MVI     A,'G'
056.233  315 227 040  2461          CALL    D.WNB             WRITE TEST PATTERN
056.236  076 114      2462          MVI     A,'L'
056.240  315 227 040  2463          CALL    D.WNB             WRITE
056.243  045          2464          DCR     H
056.244  302 231 056  2465          JNZ     IDS3              WRITE 256
056.247  172          2466          MOV     A,D
056.250  315 227 040  2467          CALL    D.WNB
056.253  315 227 040  2468          CALL    D.WNB
056.256  315 227 040  2469          CALL    D.WNB             WRITE CHECKSUM, LET TUNNEL GAP PASS
056.261  072 242 040  2470          LDA     D.DVCTL
056.264  323 177      2471          OUT     DP.DC             OFF WRITE GATE
056.266  014          2472          INR     C                 COUNT SECTOR
056.267  076 012      2473          MVI     A,10
056.271  221          2474          SUB     C
056.272  302 150 056  2475          JNE     IDS2              NOT NEW TRACK
056.275  373          2476          EI                        RESTORE INTERRUPTS
056.276  004          2477          INR     B                 COUNT NEW TRACK
056.277  117          2478          MOV     C,A               ZERO SECTOR
056.300  072 074 064  2479          LDA     LABEL+LAB.SER
056.303  062 172 056  2480          STA     IDSA              USE VOLUME NUMBER FOR OTHER TRACKS
056.306  072 321 064  2481          LDA     MAXTRK
056.311  270          2482          CMP     B
056.312  312 323 056  2483          JE      IDS4              ALL DONE
056.315  315 171 040  2484          CALL    D.MAI             MOVE ARM IN
056.320  303 116 056  2485          JMP     IDS1              WRITE NEXT TRACK
                      2486
                      2487  *       ALL DONE, RESET DISK ARM.
                      2488
056.323  076 007      2489  IDS4    MVI     A,DC.ABT
056.325  315 343 063  2490          CALL    SYDD.             ABORT DISK
                      2491
                      2492  *       WRITE DUMY BOOTSTRAP
                      2493
056.330  001 000 007  2494          LXI     B,SOBOOTL
056.333  021 200 042  2495          LXI     D,SOBOOT
056.336  041 000 000  2496          LXI     H,0
056.341  076 001      2497          MVI     A,DC.WRI
056.343  303 343 063  2498          JMP     SYDD.             WRITE BOOT
```

```
                         **      GBL - GET BAD SECTOR LIST.
                         *
                         *       GBL GETS A LIST OF BAD SECTORS, AND FLAGS THE BYTES IN THE RGTAB.
                         *
                         *       ENTRY   NONE
                         *       EXIT    RGTAB SETUP
                         *       USES    ALL


056.346                  GBL     EQU     *
056.346  041 000 071             LXI     H,RGTAB
056.351  076 310                 MVI     A,400/SPG          (A) = # OF EXISTING GROUPS
056.353  066 001         GBL1    MVI     M,1                FLAG UNUSED
056.355  043                     INX     H
056.356  075                     DCR     A
056.357  302 353 056             JNZ     GBL1               ZERO TABLE
000.310                  .       SET     400/SPG
056.362  076 070                 MVI     A,256-.            (A) = REMAINDER
000.070                          ERRZR   256-.              REQUIRE SOME
056.364  066 377         GBL1.5  MVI     M,377Q
056.366  043                     INX     H
056.367  075                     DCR     A
056.370  302 364 056             JNZ     GBL1.5             MORE TO GO

056.373  247             GBL2    ANA     A                  CLEAR CARRY
056.374  322 117 057             JNC     GBL4               DONT ALLOW THIS QUESTION
056.377  315 075 062             CALL    $CCO               CLEAR CTL-O
057.002  315 136 031             CALL    $TYPTX
057.005  012 122 105             DB      NL,'RESTRICT TO 35 TRACKS (YES/NO) <NO> ?',' '+200Q
057.054  315 213 063             CALL    $ITL
057.057  176                     MOV     A,M
057.060  247                     ANA     A
057.061  312 117 057             JZ      GBL4               IS DEFAULT <NO>
057.064  376 116                 CPI     'N'
057.066  312 117 057             JE      GBL4               IS NO
057.071  376 015                 CPI     CR
057.073  312 117 057             JE      GBL4               IS NO
057.076  376 131                 CPI     'Y'
057.100  302 373 056             JNE     GBL2               TRY AGAIN

                         *       RESTRICT TO 35. FLAG LAST 5 BAD

057.103  041 257 071             LXI     H,35*10/2+RGTAB
057.106  076 031                 MVI     A,5*10/2
057.110  066 377         GBL3    MVI     M,-1
057.112  043                     INX     H
057.113  075                     DCR     A
057.114  302 110 057             JNZ     GBL3               FLAG BAD

                         *       GET BAD SECTOR LIST

057.117  315 075 062     GBL4    CALL    $CCO               CLEAR CTL-O
057.122  315 136 031             CALL    $TYPTX
057.125  012 105 156             DB      NL,'Enter the numbers of the bad sectors one at a time. Hit RETURN'
057.224  012 141 146             DB      NL,'after each entry, and when finished.',ENL
057.272  315 075 062     GBL5    CALL    $CCO               CLEAR CTL-O
```

```
057.275  315 136 031  2557         CALL    $TYPTX
057.300  123 145 143  2558         DB      'Sector?',' '+200Q
057.310  315 134 063  2559         CALL    $IDN                INPUT DECIMAL NUBER
057.313  353          2560         XCHG
057.314  041 160 376  2561         LXI     H,-400
057.317  031          2562         DAD     D
057.320  322 361 057  2563         JNC     GBL7                OK NUMBER
057.323  315 136 031  2564  GBL6   CALL    $TYPTX
057.326  040 040 111  2565         DB      '  Illegal Sector Number',ENL
057.356  303 272 057  2566         JMP     GBL5
                      2567
057.361  172          2568  GBL7   MOV     A,D
057.362  263          2569         ORA     E
057.363  310          2570         RZ                          CARRIAGE RETURN
057.364  041 365 377  2571         LXI     H,-11
057.367  031          2572         DAD     D
057.370  322 323 057  2573         JNC     GBL6                BAD NUMBER
                      2574
                      2575  *      FLAG SECTOR BAD
                      2576
057.373  172          2577         MOV     A,D
057.374  247          2578         ANA     A
057.375  037          2579         RAR
057.376  147          2580         MOV     H,A
057.377  173          2581         MOV     A,E
060.000  037          2582         RAR
060.001  157          2583         MOV     L,A                 DIVIDE BY 2
060.002  021 000 071  2584         LXI     D,RGTAB
060.005  031          2585         DAD     D
060.006  066 377      2586         MVI     M,-1                FLAG BAD
060.010  303 272 057  2587         JMP     GBL5                GET MORE
```

```
                          2590  **       FOV - FORMAT VOLUME.
                          2591  *
                          2592  *        FOV FORMATS THE NEW VOLUME.
                          2593  *
                          2594  *        IT ALLOCATES SPACE FOR THE DIRECTORY, BUILDS THE
                          2595  *        DIRECTORY STRUCTURE, AND BUILDS THE
                          2596  *        RGT AND THE GRT.
                          2597
                          2598
  060.013                 2599  FOV      EQU     *
                          2600
                          2601  *        FLAG (IN THE RGT) THE LOW SECTORS RESERVED
                          2602  *        FOR THE SYSTEM.
                          2603
  060.013  041 002 071    2604           LXI     H,RGTAB+2         SKIP GROUPS 0 AND 1, WHICH ARE SPECIAL
  377.374                 2605           ERRPL   SPG*3-DDF.RGT     MUST BE AT LEAST 3 SPECIAL GROUPS
  060.016  076 003        2606           MVI     A,DDF.RGT+SPG-1/SPG-2
  060.020  066 377        2607  FOV1     MVI     M,377Q            FLAG RESERVED
  060.022  043            2608           INX     H
  060.023  075            2609           DCR     A
  060.024  302 020 060    2610           JNZ     FOV1
  060.027  041 000 000    2611           LXI     H,0
  060.032  042 000 071    2612           SHLD    RGTAB             FIRST TWO BLOCKS ARE SPECIAL
                          2613
  060.035  315 147 060    2614           CALL    ADB               ASSIGN DIRECTORY BLOCKS
                          2615
                          2616  *        WRITE THE RGT
                          2617
  060.040  072 074 064    2618           LDA     LABEL+LAB.SER
  060.043  062 252 040    2619           STA     D.DRVTB+1
  060.046  001 000 001    2620           LXI     B,256
  060.051  021 000 071    2621           LXI     D,RGTAB
  060.054  041 012 000    2622           LXI     H,DDF.RGT
  060.057  076 001        2623           MVI     A,DC.WRI
  060.061  315 343 063    2624           CALL    SYDD.             WRITE IT
                          2625
                          2626
                          2627  *        WRITE THE LABEL SECTOR
                          2628
  060.064  052 310 040    2629           LHLD    S.DATC
  060.067  042 075 064    2630           SHLD    LABEL+LAB.IND
  060.072  257            2631           XRA     A
  060.073  062 252 040    2632           STA     D.DRVTB+1                    CLEAR VOLUME NUMBER
  060.076  001 000 001    2633           LXI     B,256
  060.101  021 074 064    2634           LXI     D,LABEL
  060.104  041 011 000    2635           LXI     H,DDF.LAB
  060.107  076 001        2636           MVI     A,DC.WRI
  060.111  315 343 063    2637           CALL    SYDD.             WRITE LABEL
                          2638
                          2639  *        WRITE THE GRT
                          2640
  060.114  315 324 060    2641           CALL    BGT               BUILD THE GRT FIRST
  060.117  072 074 064    2642           LDA     LABEL+LAB.SER
  060.122  062 252 040    2643           STA     D.DRVTB+1
  060.125  001 000 001    2644           LXI     B,256
  060.130  021 000 070    2645           LXI     D,GRTAB
```

```
060.133  052 101 064  2646          LHLD    LABEL+LAB.GRT    (HL) = GRT SECTOR INDEX
060.136  076 001      2647          MVI     A,DC.WRI
060.140  315 343 063  2648          CALL    SYDD.            WRITE IT
                      2649
                      2650  *       INITIALIZE THE DIRECTORY
                      2651
060.143  315 053 061  2652          CALL    IDD              INITIALIZE DEVICE DIRECTORY
060.146  311          2653          RET
```

```
                            **      ADB - ASSIGN DIRECTORY BLOCKS.
                            *
                            *       ADB LOCATES 20 CONTIGUOUS SECTORS TO HOLD THE DIRECTORY
                            *       AND THE GRT TABLE.
                            *
                            *       ENTRY   NONE
                            *       EXIT    GRTBLK = BLOCK INDEX OF GRT
                            *               DIRBLK = BLOCK INDEX OF DIRECTORY FIRST BLOCK
                            *               LABEL+LAB.GRT = GRT SECTOR INDEX
                            *               LABEL+LAB.DIS = DIRECTORY START SECTOR INDEX (NOT 1ST DIR SECTOR!)
                            *       USES    ALL


  060.147                   ADB     EQU     *

                            *       FIND 10 CONTIGUOUS FREE BLOCKS

  000.000                           ERRNZ   SPG-2                   CODE ASSUMES 2

  060.147  041 156 071              LXI     H,RGTAB+110                     START LOOKING 1/3 THROUGH
  060.152  006 012          ADB1    MVI     B,10                    NEED 10 BLOCKS
  060.154  176              ADB2    MOV     A,M                     (A) = RESERVATION BYTE
  060.155  043                      INX     H
  060.156  075                      DCR     A
  060.157  302 171 060              JNZ     ADB3                    RESERVED
  060.162  005                      DCR     B                       COUNT GOT ONE
  060.163  302 154 060              JNZ     ADB2                    NEED MORE
  060.166  303 267 060              JMP     ADB4                    GOT OUR 10 BLOCKS

                            *       RAN INTO A BAD BLOCK. START LOOKING OVER AGAIN

  060.171  076 271          ADB3    MVI     A,VOLSIZ/SPG-15+#RGTAB
  060.173  225                      SUB     L
  060.174  362 152 060              JP      ADB1                    STILL ENOUGH FOR A CHANCE
  060.177  306 017                  ADI     15
  060.201  372 152 060              JM      ADB1                    STILL ENOUGH FOR A CHANCE
  060.204  315 136 031              CALL    $TYPTX                  CANT GET 10 GOOD ONES IN A ROW!
  060.207  007 040 126              DB      BELL,' Volume too decrepid for use. Try another.',BELL,ENL
  060.264  303 056 051              JMP     RESTART

                            *       GOT THE TRACKS. (HL) = INDEX OF FIRST GRT+11

  060.267  053              ADB4    DCX     H                       (HL) = INDEX IF LAST GRT
  060.270  175                      MOV     A,L
  060.271  062 071 064              STA     GRTBLK                  USE THIS BLOCK FOR GRT
  060.274  345                      PUSH    H                       SAVE INDEX
  060.275  046 000                  MVI     H,0
  000.000                           ERRNZ   SPG-2
  060.277  051                      DAD     H                       (HL) = SECTOR INDEX
  060.300  042 101 064              SHLD    LABEL+LAB.GRT           SET GRT ADDRESS
  060.303  341                      POP     H
  060.304  001 367 377              LXI     B,-9
  060.307  011                      DAD     B                       (HL) = BLOCK # FOR BEGINNING OF DIRECTORY

                            *       SETUP POINTERS FOR LABEL AND INIT PROGRAM.

```

```
     060.310  175          2713           MOV     A,L                (A) = DIRECTORY BLOCK INDEX
     060.311  062 072 064  2714           STA     DIRBLK
     000.000               2715           ERRNZ   DIRSTRT-1          ASSUME DIRECTORY STARTS ON 2ND BLOCK
     060.314  054          2716           INR     L
     060.315  046 000      2717           MVI     H,0
     060.317  051          2718           DAD     H                  (HL) = SECTOR INDEX OF START OF DIRECTORY
     060.320  042 077 064  2719           SHLD    LABEL+LAB.DIS
     060.323  311          2720           RET


                           2722  **       BGT - BUILD GRT TABLE.
                           2723  *
                           2724  *        BGT BUILDS THE GRT FROM THE RGT, ENTERING BLOCK CHAINS
                           2725  *        FOR THE DIRECTORY, THE RGT AND THE GRT FILES.
                           2726  *
                           2727  *        FIRST, THE BLOCKS USED IN THE RGT, GRT, AND DIRECTORY
                           2728  *        FILES ARE LINKED UP.
                           2729  *
                           2730  *        THEN, ALL FREE BLOCKS LEFT (INDEX = 1) ARE LINKED INTO
                           2731  *        THE FREE CHAIN.
                           2732  *
                           2733  *        ENTRY   NONE
                           2734  *        EXIT    NONE
                           2735  *        USES    ALL
                           2736
                           2737
     060.324  001 000 001  2738  BGT      LXI     B,256
     060.327  021 000 071  2739           LXI     D,RGTAB
     060.332  041 000 070  2740           LXI     H,GRTAB
     060.335  315 252 030  2741           CALL    $MOVE              COPY RGTAB INTO GRTAB
     060.340  257          2742           XRA     A
     060.341  062 005 070  2743           STA     DDF.RGT/SPG+GRTAB          RESERVE THE RGT BLOCK (FOR THE RGT FILE)
                           2744
                           2745  *        LINK THE DIRECTORY BLOCKS TOGETHER, IN THE ORDER
                           2746  *
                           2747  *          23  67                    DIRECTORY
                           2748  *        01  45  89
                           2749  *          23' 67'
                           2750  *        01' 45'
                           2751  *
                           2752  *                89'                GRT
                           2753
     000.001               2754  DIRSTRT  EQU     1                  START DIRECTORY WITH 2ND BLOCK (BLOCKS 23)
                           2755
     000.000               2756           ERRNZ   SPG-2              ASSUME SPG
     060.344  041 000 070  2757           LXI     H,GRTAB
     060.347  072 072 064  2758           LDA     DIRBLK             (A) = DIRECTORY INDEX OF 01
     060.352  306 001      2759           ADI     DIRSTRT            (A) = INDEX OF 23
     060.354  157          2760           MOV     L,A
     060.355  306 002      2761           ADI     2
     060.357  167          2762           MOV     M,A                23 -> 67
     060.360  157          2763           MOV     L,A
     060.361  326 003      2764           SUI     3
     060.363  167          2765           MOV     M,A                67 -> 01
```

```
060.364  157                         MOV     L,A
060.365  306 002                     ADI     2
060.367  167                         MOV     M,A               01 -> 45
060.370  157                         MOV     L,A
060.371  306 002                     ADI     2
060.373  167                         MOV     M,A               45 -> 89
060.374  157                         MOV     L,A
060.375  306 002                     ADI     2
060.377  167                         MOV     M,A               89 -> 23'
061.000  157                         MOV     L,A
061.001  306 002                     ADI     2
061.003  167                         MOV     M,A               23' -> 67'
061.004  157                         MOV     L,A
061.005  326 003                     SUI     3
061.007  167                         MOV     M,A               67' -> 01'
061.010  157                         MOV     L,A
061.011  306 002                     ADI     2
061.013  167                         MOV     M,A               01' -> 45'
061.014  157                         MOV     L,A
061.015  066 000                     MVI     M,0               45' -> 00
061.017  062 073 064                 STA     DIRLBLK           SET DIRECTORY LAST BLOCK
061.022  072 071 064                 LDA     GRTBLK
061.025  157                         MOV     L,A
061.026  066 000                     MVI     M,0               GRT BLOCK CHAINS TO NONE

                              *      CHAIN ALL UNUSED (001) BLOCKS TOGETHER.
                              *
                              *      PUT START POINTER IN BLOCK 00, LAST BLOCK POINTS TO 00

061.030  016 000                     MVI     C,0               (C) = NEXT FREE BLOCK
061.032  041 377 070                 LXI     H,GRTAB+255       WORK FROM THE BACK
061.035  076 001              BGT2   MVI     A,1
061.037  276                         CMP     M
061.040  302 045 061                 JNE     BGT3              NOT FREE
061.043  161                         MOV     M,C               LINK ON FRONT OF CHAIN
061.044  115                         MOV     C,L               IS NOW FRONT OF CHAIN
061.045  055                  BGT3   DCR     L
061.046  302 035 061                 JNZ     BGT2              MORE TO GO
061.051  161                         MOV     M,C               SET POINTER IN GROUP 00
061.052  311                         RET



                              **     IDD - INITIALIZE DEVICE DIRECTORY.
                              *
                              *      IDD INITIALIZES THE DEVICE DIRECTORY BY WRITTING THE DIRECTORY
                              *      BLOCK 1 CLEAR, BLOCKS 3 - N EMPTY, AND BLOCK 2 CONTAINING
                              *      THE FILES
                              *
                              *      RGT     .SYS
                              *      GRT     .SYS
                              *      DIRECT  .SYS
                              *
                              *      THE DIRECTORYS BLOCKS ARE INTERNALLY LINKED TO FOLLOW THEIR POSITION
                              *      IN THE GRT.
```

```
                         *
                         *         ENTRY   NONE
                         *         EXIT    NONE
                         *         USES    ALL


061.053  076 002           IDD     MVI     A,2
061.055  062 042 062               STA     IDDD                SET COUNT FOR SECOND DIRECTORY BLOCK
061.060  006 377                   MVI     B,DF.EMP            SET ENTRYS EMPTY
061.062  315 043 062               CALL    PDS                 PRESET DIRECTORY SECTOR

                         *         WRITE DIRECTORY BLOCKS, IN ORDER

061.065  076 027                   MVI     A,DIRELEN
061.067  062 015 067               STA     IDDA+DIS.ENL        SET DIRECTORY ENTRY LENGTH
061.072  052 077 064               LHLD    LABEL+LAB.DIS       (HL) = INDEX OF FIRST BLOCK
061.075  042 016 067       IDD2    SHLD    DIS.SEC+IDDA        SET IN DIRECTORY ENTRY
000.000                            ERRNZ   SPG-2               ASSUME =2
061.100  174                       MOV     A,H
061.101  037                       RAR
061.102  175                       MOV     A,L
061.103  037                       RAR                         (A) = GROUP INDEX OF THIS DIRECTORY BLOCK
061.104  157                       MOV     L,A
061.105  046 070                   MVI     H,GRTAB/256
061.107  156                       MOV     L,M                 (L) = INDEX OF FOLLOWING ENTRY
061.110  046 000                   MVI     H,0
061.112  051                       DAD     H                   (HL) = SECTOR INDEX OF FOLLOWING ENTRY
061.113  042 020 067               SHLD    DIS.LNK+IDDA        SET IN BLOCK
061.116  041 042 062               LXI     H,IDDD
061.121  065                       DCR     M                   COUNT SECTOR
061.122  302 257 061               JNZ     IDD4                IS NOT SECTOR 2

                         *         IS SECTOR 2. ADD SPECIAL FILES TO THE END OF IT.

061.125  052 310 040               LHLD    S.DATC
061.130  042 331 061               SHLD    IDDC2               SET CREATION AND ALTERATION DATES FOR ALL
061.133  042 333 061               SHLD    IDDC2+2
061.136  042 360 061               SHLD    IDDC4
061.141  042 362 061               SHLD    IDDC4+2
061.144  042 007 062               SHLD    IDDC6
061.147  042 011 062               SHLD    IDDC6+2

                         *         SET DIRECTORY POINTERS TO GRT BLOCK AND DIRECTORY

061.152  072 071 064               LDA     GRTBLK
061.155  062 355 061               STA     IDDC3               SET GRT BLOCK
061.160  062 356 061               STA     IDDC3+1             ONLY ONE BLOCK IN FILE
061.163  076 001                   MVI     A,1
061.165  062 357 061               STA     IDDC3+2             ONLY ONE SECTOR IN FILE
061.170  052 077 064               LHLD    LABEL+LAB.DIS
061.173  174                       MOV     A,H
061.174  037                       RAR
061.175  175                       MOV     A,L
061.176  037                       RAR                         (A) = GROUP NUMBER OF DIRECTORY START
061.177  062 004 062               STA     IDDC5
061.202  072 073 064               LDA     DIRLBLK
```

```
061.205  062 005 062  2875          STA     IDDC5+1          SET GROUP NUMBER OF LAST BLOCK
061.210  076 002      2876          MVI     A,2
061.212  062 006 062  2877          STA     IDDC5+2          SET LAST SECTOR INDEX
                      2878
061.215  001 134 000  2879          LXI     B,IDDCL
061.220  021 306 061  2880          LXI     D,IDDC
061.223  041 260 066  2881          LXI     H,IDDB+IDDBL-IDDCL
061.226  315 252 030  2882          CALL    $MOVE            MOVE ENTRYS INTO BLOCK
061.231  001 000 002  2883          LXI     B,512
061.234  021 022 065  2884          LXI     D,IDDA
061.237  052 016 067  2885          LHLD    DIS.SEC+IDDA
061.242  076 001      2886          MVI     A,DC.WRI
061.244  315 343 063  2887          CALL    SYDD.            WRITE BLOCK AND EXIT
                      2888
                      2889  *       HAVE WRITTEN 1ST AND 2ND DIRECTORY BLOCKS. FLAG
                      2890  *       ALL REMAINING BLOCKS EMPTY
                      2891
061.247  006 376      2892          MVI     B,DF.CLR         FLAG REST OF DIRECTORY EMPTY
061.251  315 043 062  2893          CALL    PDS              PRESET DIRECTORY SECTOR
061.254  303 275 061  2894          JMP     IDD5             SKIP RE-WRITTING 2ND SECTOR
                      2895
                      2896  *       WRITE BLOCK TO DISK
                      2897
061.257  001 000 002  2898  IDD4    LXI     B,512
061.262  021 022 065  2899          LXI     D,IDDA
061.265  052 016 067  2900          LHLD    DIS.SEC+IDDA
061.270  076 001      2901          MVI     A,DC.WRI
061.272  315 343 063  2902          CALL    SYDD.            WRITE BLOCK
061.275  052 020 067  2903  IDD5    LHLD    DIS.LNK+IDDA
061.300  174          2904          MOV     A,H
061.301  265          2905          ORA     L
061.302  302 075 061  2906          JNZ     IDD2             NOT LAST ONE, DO SOME MORE
061.305  311          2907          RET
                      2908
                      2909
                      2910  **      DIRECTORY ENTRYS FOR SPECIAL FILES
                      2911
061.306               2912  IDDC    DS      0
000.000               2913          ERRNZ   *-IDDC-DIR.NAM
061.306  122 107 124  2914          DB      'RGT',0,0,0,0,0  RGT.SYS
000.000               2915          ERRNZ   *-IDDC-DIR.EXT
061.316  123 131 123  2916          DB      'SYS'
000.000               2917          ERRNZ   *-IDDC-DIR.PRO
061.321  000 000      2918          DB      0,0              VERSION AND PPN
000.000               2919          ERRNZ   *-IDDC-DIR.CLU
061.323  000          2920          DB      0                CLUSTER
000.000               2921          ERRNZ   *-IDDC-DIR.FLG
061.324  360          2922          DB      DIF.SYS+DIF.LOC+DIF.CNT+DIF.WP  SET UNCHANGABLY WRITE-PROTECTED
061.325  000          2923          DB      0                UNUSED
000.000               2924          ERRNZ   *-IDDC-DIR.FGN
061.326  005          2925          DB      DDF.RGT/SPG      FIRST GROUT
000.000               2926          ERRNZ   *-IDDC-DIR.LGN
061.327  005          2927          DB      DDF.RGT/SPG      LAST GROUP
000.000               2928          ERRNZ   *-IDDC-DIR.LSI
061.330  001          2929          DB      1                SECTOR INDEX OF EOF
000.000               2930          ERRNZ   *-IDDC-DIR.CRD
```

```
     061.331                     IDDC2   DS      4                   CREATION AND ALTERATION DATE

     061.335  107 122 124                DB      'GRT',0,0,0,0,0             GRT.SYS
     061.345  123 131 123                DB      'SYS'
     061.350  000 000                    DB      0,0                 PPN, VERSION
     061.352  000                        DB      0                   CLUSTER
     061.353  360                        DB      DIF.SYS+DIF.LOC+DIF.CNT+DIF.WP  UNCHANGABLY WRITE-PROTECT
     061.354  000                        DB      0                   UNUSED
     061.355                     IDDC3   DS      3                   FIRST GROUP, LAST GROUP, LAST SECTOR
     061.360                     IDDC4   DS      4                   CREATION AND ALTERATION DATE

     061.364  104 111 122                DB      'DIRECT',0,0                DIRECT.SYS
     061.374  123 131 123                DB      'SYS'
     061.377  000 000                    DB      0,0                 PPN, VERSION
     062.001  000                        DB      0                   CLUSTER
     062.002  340                        DB      DIF.SYS+DIF.LOC+DIF.WP  UNCHANGABLY WRITE PROTECT
     062.003  000                        DB      0                   UNUSED
     062.004                     IDDC5   DS      3                   FIRST GROUP, LAST GROUP, LAST SECTOR
     062.007                     IDDC6   DS      4                   CREATION AND ALTERATION DATE
     062.013  376                        DB      DF.CLR              LAST SPOT IN 2ND BLOCK IS EMPTY
     062.014  107 101 103                DB      'GAC / HEATH CO.'
     062.033                             DS      DIRELEN-1-15        REST OF ENTRY MEANINGLESS

     000.134                     IDDCL   EQU     *-IDDC
     000.000                             ERRNZ   4*DIRELEN-IDDCL             SHOULD BE FOUR ENTRYS

     062.042  000                IDDD    DB      0                   DIRECTORY BLOCK COUNTER



                                 **      PDS - PRESET DIRECTORY SECTOR.
                                 *
                                 *       PDS BUILDS A DIRECTORY BLOCK INTO *IDDB*  FULL OF DIRECTORY
                                 *               ENTRYS IN THE FORMAT:
                                 *
                                 *       FLAG,0,0, . . . ,0
                                 *
                                 *       WHERE FLAG = SOME  SUPPLIED VALUE.
                                 *
                                 *       ENTRY   (B) = FLAG BYTE
                                 *       EXIT    NONE
                                 *       USES    ALL


     062.043  041 022 065        PDS     LXI     H,IDDB
     062.046  021 372 001                LXI     D,IDDBL
     062.051  160                PDS1    MOV     M,B                 SET FIRST BYTE
     062.052  043                        INX     H
     062.053  033                        DCX     D

                                 *       NOW ZERO REST OF ENTRY

     062.054  016 026                    MVI     C,DIRELEN-1
     062.056  066 000            PDS2    MVI     M,0                 ZERO IT
     062.060  043                        INX     H
```

```
062.061  033           2984          DCX     D               COUNT BYTE FROM DIRECTORY BLOCK
062.062  015           2985          DCR     C               COUNT BYTE FROM DIRECTORY ENTRY
062.063  302 056 062   2986          JNZ     FDS2
062.066  172           2987          MOV     A,D
062.067  263           2988          ORA     E
062.070  302 051 062   2989          JNZ     FDS1
062.073  167           2990          MOV     M,A             ZERO BYTE FOLLOWS ENTRYS
062.074  311           2991          RET                     EXIT
```

```
    062.075                2994          XTEXT   SAVALL


                           2996X **      $RSTALL - RESTORE ALL REGISTERS.
                           2997X *
                           2998X *       $RSTALL RESTORES ALL THE REGISTERS OFF THE STACK, AND
                           2999X *       RETURNS TO THE PREVIOUS CALLER.
                           3000X *
                           3001X *       ENTRY   (SP) = PSW
                           3002X *               (SP+2) = BC
                           3003X *               (SP+4) = DE
                           3004X *               (SP+6) = HL
                           3005X *               (SP+8) = RET
                           3006X *       EXIT    TO *RET*, REGISTERS RESTORED
                           3007X *       USES    ALL
                           3008X
                           3009X
    031.047                3010X $RSTALL EQU     31047A          IN H17 ROM


                           3012X **      $SAVALL - SAVE ALL REGISTERS ON STACK.
                           3013X *
                           3014X *       $SAVALL SAVES ALL THE REGISTERS ON THE STACK.
                           3015X *
                           3016X *       ENTRY   NONE
                           3017X *       EXIT    (SP) = PSW
                           3018X *               (SP+2) = BC
                           3019X *               (SP+4) = DE
                           3020X *               (SP+6) = HL
                           3021X *       USES    H,L
                           3022X
                           3023X
    031.054                3024X $SAVALL EQU     31054A          IN H17 ROM
    062.075                3025          XTEXT   CCO


                           3027X **      $CCO - CLEAR CONTROL-O
                           3028X *
                           3029X *       $CCO IS CALLED TO CLEAR THE EFFECT OF THE CTL-O CHARACTER.
                           3030X *
                           3031X *       ENTRY   NONE
                           3032X *       EXIT    NONE
                           3033X *       USES    NONE
                           3034X
                           3035X
    062.075  315 054 031   3036X $CCO    CALL    $SAVALL         SAVE REGISTERS
    062.100  076 004       3037X         MVI     A,I.CONFL
    062.102  001 001 000   3038X         LXI     B,CO.FLG        CLEAR CO.FLG
    062.105  377 006       3039X         DB      SYSCALL,.CONSL
    062.107  303 047 031   3040X         JMP     $RSTALL         RESTORE REGISTERS AND RETURN
    062.112                3041          XTEXT   CDEHL
```

```
                         3043X **        $CDEHL - COMPARE (DE) TO (HL)
                         3044X *
                         3045X *         $CDEHL COMPARES (DE) TO (HL) FOR EQUALITY.
                         3046X *
                         3047X *         ENTRY   NONE
                         3048X *         EXIT    'Z' SET IF (DE) = (HL)
                         3049X *         USES    A,F
                         3050X
                         3051X
      030.216            3052X $CDEHL    EQU     30216A          IN H17 ROM
      062.112            3053            XTEXT   CHL



                         3055X **        $CHL - COMPLEMENT (HL).
                         3056X *
                         3057X *         (HL) = -(HL)            TWO'S COMPLEMENT
                         3058X *
                         3059X *         ENTRY   NONE
                         3060X *         EXIT    NONE
                         3061X *         USES    A,F,H,L
                         3062X
                         3063X
      030.224            3064X $CHL      EQU     30224A          IN H17 ROM
      062.112            3065            XTEXT   DTB



                         3067X **        $DTB - DELETE TRAILING BLANKS.
                         3068X *
                         3069X *         $DTB DELETES THE TRAILING BLANKS FROM A CODED LINE.
                         3070X *
                         3071X *         ENTRY   (HL) = LINE FWA
                         3072X *         EXIT    (A) = LENGTH OF RESULT (ENCLUDING 00 TERMINATOR BYTE)
                         3073X *         USES    A,F
                         3074X
                         3075X
      062.112  325       3076X $DTB      PUSH    D               SAVE (DE)
      062.113  124       3077X           MOV     D,H
      062.114  135       3078X           MOV     E,L             (DE) = FWA
      062.115  033       3079X           DCX     D               (DE) = FWA-1
      062.116  176       3080X $DTB1     MOV     A,M
      062.117  043       3081X           INX     H
      062.120  247       3082X           ANA     A               FIND END OF LINE
      062.121  302 116 062 3083X         JNZ     $DTB1
      062.124  053       3084X           DCX     H               (HL) = ADDRESS OF TERMINATING ZERO BYTE
                         3085X
                         3086X *         GOT END OF LINE, DELETE TRAILING BLANKS
                         3087X
      062.125  053       3088X $DTB2     DCX     H               BACKUP ONE CHARACTER
      062.126  315 216 030 3089X         CALL    $CDEHL
      062.131  312 142 062 3090X         JE      $DTB3           GONE PAST FRONT OF LINE, MUST BE ALL BLANKS
      062.134  176       3091X           MOV     A,M
      062.135  376 040   3092X           CPI     ' '
```

```
     062.137  312 125 062  3093X             JE       $DTB2            GOT BLANK
                           3094X
                           3095X *           HAVE TRIMED LINE. COMPUTE LENGTH
                           3096X
     062.142  043          3097X $DTB3       INX      H
     062.143  066 000      3098X             MVI      M,0              TERMINATE LINE
     062.145  175          3099X             MOV      A,L
     062.146  223          3100X             SUB      E                (A) = LENGTH +1 (FOR 00 BYTE)
     062.147  353          3101X             XCHG
     062.150  043          3102X             INX      H                (HL) = LINE FWA
     062.151  321          3103X             POP      D                RESTORE (DE)
     062.152  311          3104X             RET
     062.153               3105              XTEXT    MLU



                           3107X **          MLU - MAP LOWER CASE LINE TO UPPER CASE.
                           3108X *
                           3109X *           MLU MAPS THE LOWER CASE ALPHABETICS IN A LINE TO UPPER CASE.
                           3110X *
                           3111X *           ENTRY    (HL)  = LINE FWA
                           3112X *           EXIT     NONE
                           3113X *           USES     NONE
                           3114X
                           3115X
     062.153  365          3116X $MLU        PUSH     PSW              SAVE (PSW)
     062.154  345          3117X             PUSH     H                SAVE FWA
     062.155  053          3118X             DCX      H                ANTICIPATE INX H
     062.156  043          3119X $MLU1       INX      H
     062.157  176          3120X             MOV      A,M              (A)= CHARACTER
     062.160  315 007 050  3121X             CALL     $MCU             MAP CHAR TO UPPER
     062.163  167          3122X             MOV      M,A
     062.164  247          3123X             ANA      A
     062.165  302 156 062  3124X             JNZ      $MLU1            MORE TO GO
     062.170  341          3125X             POP      H                RESTORE (HL)
     062.171  361          3126X             POP      PSW              RESTORE (PSW)
     062.172  311          3127X             RET
     062.173               3128              XTEXT    RTL



                           3130X **          $RTL - READ TEXT LINE.
                           3131X *
                           3132X *           $RTL READS A LINE FROM THE TERMINAL.
                           3133X *
                           3134X *           CHARACTER ARE ACCEPTED FROM THE TERMINAL, RUBOUT AND BACKSPACE
                           3135X *           CHARACTERS ARE PROCESSED. WHEN A CARRIAGE RETURN IS ENTERED,
                           3136X *           $RTL RETURNS.
                           3137X *
                           3138X *           ENTRY    (HL) = BUFFER FWA
                           3139X *           EXIT     'C' CLEAR IF OK
                           3140X *                     DATA IN BUFFER
                           3141X *                     (A) = TEXT LENGTH
                           3142X *                   'C' SET IF CTL-D STRUCK
                           3143X *       USES     A,F
                           3144X
                           3145X
     062.173  315 202 062  3146X $RTL.   CALL     $RTL              $RTL IN UPPER CASE
     062.176  330          3147X         RC                         CTL-D
     062.177  303 153 062  3148X         JMP      $MLU              MAP LINE TO UPPER CASE
                           3149X
     062.202               3150X $RTL    EQU      *
     062.202  345          3151X         PUSH     H                 SAVE FWA
     062.203  315 054 063  3152X $RTL1   CALL     $RCHAR
     062.206  376 004      3153X         CPI      CTLD
     062.210  312 235 062  3154X         JE       $RTL2             CTL-D STRUCK
     062.213  167          3155X         MOV      M,A
     062.214  043          3156X         INX      H
     062.215  376 012      3157X         CPI      NL
     062.217  302 203 062  3158X         JNE      $RTL1
     062.222  053          3159X         DCX      H
     062.223  066 000      3160X         MVI      M,0
     062.225  043          3161X         INX      H
                           3162X
                           3163X *       ALL DONE, COMPUTE LENGTH
                           3164X
     062.226  353          3165X         XCHG                       (DE) = LWA+1
     062.227  343          3166X         XTHL                       (HL) = FWA
     062.230  173          3167X         MOV      A,E
     062.231  225          3168X         SUB      L                 (A) = LENGTH
     062.232  247          3169X         ANA      A                 CLEAR CARRY
     062.233  321          3170X         POP      D                 RESTORE (DE)
     062.234  311          3171X         RET
                           3172X
                           3173X *       CTL-D STRUCK
                           3174X
     062.235  341          3175X $RTL2   POP      H                 (HL) = FWA
     062.236  067          3176X         STC
     062.237  311          3177X         RET
     062.240               3178          XTEXT    UDD



                           3180X **      $UDD - UNPACK DECIMAL DIGITS.
                           3181X *
                           3182X *       UDD CONVERTS A 16 BIT VALUE INTO A SPECIFIED NUMBER OF
                           3183X *       DECIMAL DIGITS. THE RESULT IS ZERO FILLED.
                           3184X *
                           3185X *       ENTRY    (B,C) = ADDRESS VALUE
                           3186X *                (A) = DIGIT COUNT
                           3187X *                (H,L) = MEMORY ADDRESS
                           3188X *       EXIT     (HL) = (HL) + (A)
                           3189X *       USES     ALL
                           3190X
                           3191X
     031.157               3192X $UDD    EQU      31157A            IN H17 ROM
     062.240               3193          XTEXT    HLIHL
```

```
                        3195X **        $HLIHL - LOAD HL INDIRECT THROUGH HL.
                        3196X *
                        3197X *         (HL) = ((HL))
                        3198X *
                        3199X *         ENTRY   NONE
                        3200X *         EXIT    NONE
                        3201X *         USES    A,H,L
                        3202X
    030.211             3203X $HLIHL  EQU     30211A          IN H17 ROM
    062.240             3204          XTEXT   DOS



                        3206X **        $DOS - DISMOUNT OPERATING SYSTEM.
                        3207X *
                        3208X *         $DOS DISMOUNTS SY2:, SY1: (IF MOUNTED), AND SY0:.       /79.11.GC/
                        3209X *
                        3210X *         THE USER IS MESSAGED ABOUT THE DISKS, AND THE OPERATING
                        3211X *         SYSTEM IS NOTIFIED.
                        3212X *
                        3213X *
                        3214X *         ENTRY   NONE
                        3215X *
                        3216X *         EXIT    (PSW)   = 'C' CLEAR IF NO ERROR
                        3217X *                           'C' SET   IF    ERROR
                        3218X *                            (A)  = ERROR CODE
                        3219X *
                        3220X *         USES    ALL
                        3221X *
                        3222X
    062.240  315 136 031 3223X $DOS     CALL    $TYPTX
    062.243  012 007 104 3224X          DB      NL,BELL,'Dismounting All Disks:',NL,ENL
                        3225X
    062.275  076 000     3226X          MVI     A,OVLO
    062.277  377 010     3227X          DB      SYSCALL,.LOADO
    062.301  330         3228X          RC
    062.302  076 001     3229X          MVI     A,OVL1
    062.304  377 010     3230X          DB      SYSCALL,.LOADO
    062.306  330         3231X          RC
                        3232X
    062.307  041 047 063 3233X          LXI     H,DOSC
    062.312  315 025 063 3234X          CALL    DOS.
    062.315  330         3235X          RC
    062.316  041 042 063 3236X          LXI     H,DOSB
    062.321  315 025 063 3237X          CALL    DOS.
    062.324  330         3238X          RC                      FATAL ERROR
    062.325  041 035 063 3239X          LXI     H,DOSA
    062.330  315 025 063 3240X          CALL    DOS.
    062.333  330         3241X          RC
                        3242X
    062.334  315 136 031 3243X          CALL    $TYPTX
    062.337  012 122 145 3244X          DB      NL,'Remove the Disk(s). Hit RETURN when ready:',' '+200Q
    063.013  315 054 063 3245X DOS1     CALL    $RCHAR          READ CHARACTER
    063.016  376 012     3246X          CPI     NL
    063.020  302 013 063 3247X          JNE     DOS1
    063.023  247            3248X          ANA     A                CLEAR CARRY
    063.024  311            3249X          RET
                            3250X
                            3251X *        DISMOUNT A DEVICE WITHOUT REGARD TO WHETHER MOUNTED OR NOT
                            3252X
    063.025  377 201        3253X DOS.     DB      SYSCALL,.DMOUN
    063.027  320            3254X          RNC
    063.030  376 042        3255X          CPI     EC.NVM           NO VOLUME MOUNTED ERROR NOT CONSIDERED FATAL
    063.032  310            3256X          RZ                       NOT FATAL, CARRY NOW CLEAR
    063.033  067            3257X          STC                      FLAG FATAL ERROR
    063.034  311            3258X          RET
                            3259X
    063.035  123 131 060    3260X DOSA     DB      'SY0:',0
    063.042  123 131 061    3261X DOSB     DB      'SY1:',0
    063.047  123 131 062    3262X DOSC     DB      'SY2:',0
    063.054                 3263           XTEXT   WER



                            3265X **       $WER - WRITE ENABLE RAM.
                            3266X *
                            3267X *        $WER IS CALLED TO ENABLE WRITTING TO THE H17 CONTROLLER'S
                            3268X *        RAM AREA.
                            3269X *
                            3270X *        ENTRY   NONE
                            3271X *        EXIT    NONE
                            3272X *        USES    NONE
                            3273X
                            3274X
    031.241                 3275X $WER     EQU     31241A           IN H17 ROM



                            3277X **       $WDR - WRITE DISABLE RAM.
                            3278X *
                            3279X *        $WDR IS CALLED TO DISABLE WRITTING TO THE H17 CONTROLLER'S
                            3280X *        RAM AREA.
                            3281X *
                            3282X *        ENTRY   NONE
                            3283X *        EXIT    NONE
                            3284X *        USES    NONE
                            3285X
                            3286X
    031.222                 3287X $WDR     EQU     31222A           IN H17 ROM
    063.054                 3288           XTEXT   RCHAR
```

```
                                 3290X **       $RCHAR - READ SINGLE CHARACTER FROM CONSOLE.
                                 3291X *
                                 3292X *        ENTRY    NONE
                                 3293X *        EXIT     (A) = CHARACTER
                                 3294X *        USES     A,F
                                 3295X
                                 3296X
       063.054  377 001          3297X $RCHAR   DB       SYSCALL,.SCIN
       063.056  332 054 063      3298X          JC       $RCHAR            NOT READY
       063.061  311              3299X          RET
                                 3300X
       063.062  377 002          3301X $WCHAR   DB       SYSCALL,.SCOUT
       063.064  311              3302X          RET
       063.065                   3303           XTEXT    TYPCH



                                 3305X **       $TYPCH - TYPE SINGLE CHARACTER.
                                 3306X *
                                 3307X *        ENTRY    (RET) = CHARACTER
                                 3308X *        EXIT     TO (RET)+1
                                 3309X *                 (A) = CHARACTER TYPED
                                 3310X
                                 3311X
       063.065  343              3312X $TYPCH   XTHL                       (HL) = RETURN ADDRESS
       063.066  176              3313X          MOV      A,M               (A) = CHARACTER
       063.067  043              3314X          INX      H
       063.070  343              3315X          XTHL                       RESTORE ADVANCED EXIT ADDRESS
                                 3316X
                                 3317X **       $TYPC. - TYPE SINGLE CHARACTER.
                                 3318X *
                                 3319X *        ENTRY    (A) = CHARACTER
                                 3320X *        EXIT     TO (RET)
                                 3321X
       063.071  377 002          3322X $TYPC.   DB       SYSCALL,.SCOUT
       063.073  311              3323X          RET
       063.074                   3324           XTEXT    TYPT2



                                 3326X **       $TYPTX - TYPE TEXT.
                                 3327X *
                                 3328X *        $TYPTX IS CALLED TO TYPE A BLOCK OF TEXT ON THE SYSTEM CONSOLE.
                                 3329X *
                                 3330X *        IMBEDDED ZERO BYTES INDICATE A CARRIAGE RETURN LINE FEED,
                                 3331X *        A BYTE WITH THE 200Q BIT SET IS THE LAST BYTE IN THE MESSAGE.
                                 3332X *
                                 3333X *        ENTRY    (RET) = TEXT
                                 3334X *        EXIT     TO (RET+LENGTH)
                                 3335X *        USES     A,F
                                 3336X
                                 3337X
       031.136                   3338X $TYPTX   EQU      31136A            IN H17 ROM
                                 3339X
```

```
031.144               3340X $TYPTX. EQU      31144A          IN H17 ROM
000.001               3341  $CMP$   EQU      1                       NO COMPRESSED TEXT
063.074               3342          XTEXT    TYPLN



                      3344X **      $TYPLN - TYPE LINE.
                      3345X *
                      3346X *       $TYPLN IS CALLED TO TYPE A LINE OF TEXT. ZERO BYTES ARE
                      3347X *       TAKEN AS CRLF (WITH THE PROPER PADDING)
                      3348X *
                      3349X *       CALL     $TYPLN
                      3350X *       DB       N               BYTE COUNT OF FOLLOWING MESSAGE
                      3351X *       DB       'N-CHARACTER MESSAGE'
                      3352X *
                      3353X *       ENTRY    (RET) = TEXT COUNT
                      3354X *                (RET)+1 - (RET)+N = TEXT
                      3355X *       EXIT     TO (RET)+N+1
                      3356X *       USES     A,F
                      3357X *
                      3358X
                      3359X
063.074  343          3360X $TYPLN. XTHL                     (H,L) = COUNT ADDRESS
063.075  176          3361X         MOV      A,M             (A) = COUNT
063.076  043          3362X         INX      H               (H,L)  = TEXT ADDRESS
063.077  345          3363X         PUSH     H               SAVE TEXT FWA
063.100  315 072 030  3364X         CALL     $DADA           CALCULATE RETURN ADDRESS
063.103  343          3365X         XTHL                     (HL) = TEXT ADDRE
063.104  315 112 063  3366X         CALL     $TYPL.          OUTPUT LINE
063.107  341          3367X         POP      H               (HL) = RETURN ADDRESS
063.110  343          3368X         XTHL                     RESTORE (HL), SET RETURN ADDRESS
063.111  311          3369X         RET
                      3370X
                      3371X **      $TYPL. - TYPE LINE.
                      3372X *
                      3373X *       ENTRY    (HL) = ADDRESS
                      3374X *                (A) = COUNT
                      3375X *       EXIT     NONE
                      3376X *       USES     A,F,H,L
                      3377X
063.112               3378X $TYPL.  EQU      *
063.112  247          3379X         ANA      A
063.113  310          3380X         RZ                       NOTHING TO TYPE
063.114  365          3381X         PUSH     PSW             SAVE COUNT
063.115  176          3382X         MOV      A,M             (A) = CHARACTER
063.116  043          3383X         INX      H
063.117  247          3384X         ANA      A
000.001               3385X         IF       $CMP$           IF HAVE COMPRESSED SPACES
                      3386X         JM       TPL2            IS COMPRESSED SPACE
                      3387X         ENDIF
063.120  314 304 063  3388X         CZ       $CRLF
063.123  315 071 063  3389X         CALL     $TYPC.          TYPE CHARACTER
063.126  361          3390X TPL1    POP      PSW
063.127  075          3391X         DCR      A
063.130  302 112 063  3392X         JNZ      $TYPL.
```

```
        063.133  311          3393X              RET
        000.001               3394X              IF      $CMP$           IF COMPRESSED TEXT
                              3395X
                              3396X *            HAVE COMPRESSED SPACE.
                              3397X
                              3398X TPL2         DCR     A
                              3399X              CP      $TYPCH          TYPE 00 IF CHARACTER WAS 200Q
                              3400X              DB      0
                              3401X              ANA     A               SET CODES
                              3402X TPL3         JP      TPL1            ALL EXPANDED
                              3403X              PUSH    PSW             SAVE COUNT
                              3404X              CALL    $TYPCH
                              3405X              DB      ' '
                              3406X              POP     PSW
                              3407X              DCR     A
                              3408X              JMP     TPL3
                              3409X              ENDIF
        063.134               3410               XTEXT   IDN


                              3412X **           $IDN - INPUT DECIMAL NUMBER.
                              3413X *
                              3414X *            $IDN IS CALLED TO INPUT A DECIMAL NUMBER FROM THE CONSOLE.
                              3415X *
                              3416X *            AN ENTIRE LINE IS ACCEPTED, AND THEN THE NUMBER IS DECODED.
                              3417X *
                              3418X *            RUBOUTS AND BACKSPACES MAY BE USED DURING ENTRY.
                              3419X *
                              3420X *            ENTRY   NONE
                              3421X *            EXIT    'C' CLEAR IF OK
                              3422X *                     (HL) = NUMBER
                              3423X *                    'C' SET IF SOME ERROR
                              3424X *            USES    ALL
                              3425X
                              3426X
        063.134  041 022 067  3427X $IDN         LXI     H,ITLA
        063.137  315 202 062  3428X              CALL    $RTL            READ LINE
        063.142  303 145 063  3429X              JMP     $PDD            PACK DECIMAL DIGITS
        063.145               3430               XTEXT   PDD


                              3432X **           $PDD - PACK DECIMAL DIGITS.
                              3433X *
                              3434X *            $PDD PACKS A STRING OF DECIMAL DIGITS INTO A DECIMAL INTEGER.
                              3435X *
                              3436X *            THE CHARACTERS MUST BE IN MEMORY, AND BE IMMEDIATELY FOLLOWED BY A
                              3437X *            00 BYTE.
                              3438X *
                              3439X *            ENTRY   (HL) = ADDRESS OF CHARACTERS
                              3440X *            EXIT    'C' CLEAR IF OK
                              3441X *                     (HL) = NUMBER
                              3442X *                    'C' SET IF ERROR
```

```
                             3443X *        USES     A,F,D,E,H,L
                             3444X
                             3445X
    063.145  353             3446X $PDD     XCHG                        (DE) = TEXT ADDRESS
    063.146  041 000 000     3447X          LXI      H,0                (HL) = ACCUM
                             3448X
    063.151  032             3449X $PDD1    LDAX     D
    063.152  023             3450X          INX      D                  ADVANCE ADDRESS
    063.153  247             3451X          ANA      A
    063.154  310             3452X          RZ                          ALL DONE
    063.155  326 060         3453X          SUI      '0'
    063.157  330             3454X          RC                          TOO SMALL
    063.160  376 012         3455X          CPI      10
    063.162  077             3456X          CMC
    063.163  330             3457X          RC                          TOO SMALL
    063.164  325             3458X          PUSH     D                  SAVE (DE)
    063.165  353             3459X          XCHG
    063.166  315 324 030     3460X          CALL     $MU10
    063.171  321             3461X          POP      D
    063.172  330             3462X          RC                          OVERFLOW
    063.173  205             3463X          ADD      L
    063.174  157             3464X          MOV      L,A
    063.175  076 000         3465X          MVI      A,0
    063.177  214             3466X          ADC      H
    063.200  147             3467X          MOV      H,A
    063.201  322 151 063     3468X          JNC      $PDD1              NOT OVERFLOW
    063.204  311             3469X          RET
    063.205                  3470           XTEXT    MU10



                             3472X **       $MU10 - MULTIPLY UNSIGNED 16 BIT QUANTITY BY 10.
                             3473X *
                             3474X *        (HL) = (DE)*10
                             3475X *
                             3476X *        ENTRY    (DE) = MULTIPLIER
                             3477X *        EXIT     'C' CLEAR IF OK
                             3478X *                  (HL) = PRODUCT
                             3479X *                 'C' SET IF ERROR
                             3480X *        USES     D,E,H,L,F
                             3481X
                             3482X
    030.324                  3483X $MU10    EQU      30324A             IN H17 ROM
    063.205                  3484           XTEXT    ITL



                             3486X **       $ITL - INPUT TEXT LINE.
                             3487X *
                             3488X *        $ITL INPUTS A LINE FROM THE TERMINAL.
                             3489X *
                             3490X *        CHARACTER ARE ACCEPTED FROM THE TERMINAL, RUBOUT AND BACKSPACE
                             3491X *        CHARACTERS ARE PROCESSED. WHEN A CARRIAGE RETURN IS ENTERED,
                             3492X *        $ITL RETURNS.
```

```
                           3493X *
                           3494X *        ENTRY   NONE
                           3495X *        EXIT    (HL) = $ITLA
                           3496X *                (A) = TEXT LENGTH
                           3497X *        USES    A,F,H,L
                           3498X
                           3499X
   063.205  315 213 063    3500X $ITL.    CALL    $ITL                  INPUT LINE IN UPPER CASE
   063.210  303 153 062    3501X          JMP     $MLU                  MAP LINE TO UPPER
                           3502X
   063.213  041 022 067    3503X $ITL     LXI     H,ITLA
   063.216  303 202 062    3504X          JMP     $RTL                  READ TEXT LINE
   063.221                 3505           XTEXT   TDD


                           3507X **       $TDD - TYPE DECIMAL DIGITS.
                           3508X *
                           3509X *        $TDD TYPES A 16 BIT VALUE AS 1 TO 5 DECIMAL DIGITS.
                           3510X *
                           3511X *        ENTRY   (D,E) = VALUE
                           3512X *                (A) = DIGIT COUNT
                           3513X *        EXIT    VALUE TYPED.
                           3514X *        USES    A,B,C,F
                           3515X
                           3516X
   063.221  076 005        3517X $TDD.    MVI     A,5
   063.223  345            3518X $TDD     PUSH    H
   063.224  365            3519X TDD1     PUSH    PSW
   063.225  041 270 063    3520X          LXI     H,TDDA-2
   063.230  007            3521X          RLC                           (A) = DIGIT NUMBER*2
   063.231  315 101 030    3522X          CALL    $DADA.
   063.234  176            3523X          MOV     A,M
   063.235  043            3524X          INX     H
   063.236  146            3525X          MOV     H,M
   063.237  157            3526X          MOV     L,A                   (HL) = MULTIPLE OF 10
   063.240  353            3527X          XCHG                          (DE) = DEVISOR, (HL) = VALUE
   063.241  076 377        3528X          MVI     A,377Q
   063.243  031            3529X TDD2     DAD     D
   063.244  074            3530X          INR     A
   063.245  332 243 063    3531X          JC      TDD2                  IF MORE TO GO
   063.250  306 060        3532X          ADI     '0'
   063.252  315 071 063    3533X          CALL    $TYPC.                TYPE DIGIT
   063.255  175            3534X          MOV     A,L
   063.256  223            3535X          SUB     E
   063.257  137            3536X          MOV     E,A                   REMOVE EXTRA SUBTRACTION
   063.260  174            3537X          MOV     A,H
   063.261  232            3538X          SBB     D
   063.262  127            3539X          MOV     D,A
   063.263  361            3540X          POP     PSW
   063.264  075            3541X          DCR     A
   063.265  302 224 063    3542X          JNZ     TDD1                  IF MORE DIGITS
   063.270  341            3543X          POP     H
   063.271  311            3544X          RET                           EXIT
                           3545X
```

```
    063.272              3546X TDDA     EQU      *
    063.272  377 377     3547X          DW       -1
    063.274  366 377     3548X          DW       -10
    063.276  234 377     3549X          DW       -100
    063.300  030 374     3550X          DW       -1000
    063.302  360 330     3551X          DW       -10000
    063.304              3552           XTEXT    DADA



                         3554X **       $DADA - PERFORM (H,L) = (H,L) + (0,A)
                         3555X *
                         3556X *        ENTRY    (H,L) = BEFORE VALUE
                         3557X *                 (A) = BEFORE VALUE
                         3558X *        EXIT     (H,L) = (H,L) + (0,A)
                         3559X *                 'C' SET IF OVERFLOW
                         3560X *        USES     F,H,L
                         3561X
                         3562X
    030.072              3563X $DADA    EQU      30072A          IN H17 ROM
    063.304              3564           XTEXT    CRLF



                         3566X **       $CRLF - TYPE CARRIAGE RETURN/ LINE FEED
                         3567X *
                         3568X *        $CRLF IS USED TO GENERATE PADDED CRLF'S.
                         3569X *
                         3570X *        ENTRY    NONE
                         3571X *        EXIT     (A) = 0
                         3572X *        USES     A,F
                         3573X
                         3574X
    063.304  076 012     3575X $CRLF    MVI      A,NL
    063.306  377 002     3576X          DB       SYSCALL,.SCOUT
    063.310  257         3577X          XRA      A
    063.311  311         3578X          RET
```

```
                     **      CGT - COUNT GAP TIMING.
                     *
                     *       CGT COMPUTES THE GAP TIMING BY COUNTING THE NUMBER OF
                     *       TWO MILISECOND DELAY INTERVALS IN THE INTER-RECORD GAP.
                     *       THE TIME FOR THE CURRENT HOLE TO PASS IS INCLUDED IN
                     *       THE COUNT.
                     *
                     *       ENTRY   DISK OVER BEGINNING OF HOLE
                     *       EXIT    (HL) = COUNT
                     *       USES    (PSW),(DE),(HL)
                     *

063.312  052 033 040 CGT     LHLD    .TICCNT

063.315  333 177     CGT1    IN      DP.DC
000.000                      ERRNZ   DP.HD-1
063.317  037                 RAR
063.320  332 315 063         JC      CGT1             HOLE PRESENT

063.323  315 224 030         CALL    $CHL
063.326  124                 MOV     D,H              SAVE (-START) TIC COUNT
063.327  135                 MOV     E,L               WHILE WE HAVE THE TIME!

063.330  333 177     CGT2    IN      DP.DC
000.000                      ERRNZ   DP.HD-1
063.332  037                 RAR
063.333  322 330 063         JNC     CGT2             COUNT HOLE NOT PRESENT

063.336  052 033 040         LHLD    .TICCNT          GET CURRENT TIC COUNT
063.341  031                 DAD     D                DELAY = CURRENT + (-START)
063.342  311                 RET



                     **      SYDD. - CALL SYSTEM DEVICE DRIVER, WITH ERROR DETECTION.
                     *
                     *       SYDD. PASSES CALLS TO SYDD, CHECKING FOR RETURNED ERRORS.
                     *
                     *       ENTRY   FOR SYDD
                     *       EXIT    FROM SYDD
                     *       USES    AS SYDD


063.343  315 130 040 SYDD.   CALL    SYDD
063.346  320                 RNC                      ALL OK
063.347  315 136 031         CALL    $TYPTX
063.352  007 040 103         DB      BELL,' CANNOT READ/WRITE THIS DISK ! FATAL ERROR !',BELL,ENL
064.031  303 133 051         JMP     EXIT             EXIT
```

```
                        3629  **      WIH - WAIT INDEX HOLE.
                        3630  *
                        3631  *       WIH WAITS UNTIL THE INDEX HOLE HAS PASSED THE SENSOR.
                        3632  *
                        3633  *       NOTE:   THIS ROUTINE ASSUMES THAT THE DRIVE WILL NOT BE THAT FAR
                        3634  *               OFF IN TOLERANCE AS IT DOES NOT EVEN CHECK THE HIGH ORDER
                        3635  *               BYTE OF THE COUNT, WHICH SHOULD BE ZERO!
                        3636  *
                        3637  *
                        3638  *       ENTRY   DISK MOVING
                        3639  *       EXIT    INDEX HOLE JUST PAST (DISK IN INTERHOLE GAP)
                        3640  *       USES    A,F,H,L
                        3641
                        3642
064.034  325            3643  WIH     PUSH    D
064.035  315 271 036    3644          CALL    R.WNH           WAIT NO HOLE
064.040  315 235 036    3645  WIH1    CALL    R.WHD           WAIT HOLE DETECT
064.043  315 312 063    3646          CALL    CGT             COMPUTE GAP TIME
064.046  175            3647          MOV     A,L
064.047  376 006        3648          CPI     1000/100/2+1
064.051  332 040 064    3649          JC      WIH1            DONT HAVE FULL SECTOR TIMING
                        3650
                        3651  *       HAVE FULL SECTOR TIMING. WAIT FOR SHORT GAP NOW
                        3652
064.054  315 312 063    3653  WIH2    CALL    CGT
064.057  175            3654          MOV     A,L
064.060  376 006        3655          CPI     1000/100/2+1
064.062  322 054 064    3656          JNC     WIH2            NOT YET
064.065  321            3657          POP     D
064.066  303 271 036    3658          JMP     R.WNH           GOT INDEX HOLE, WAIT FOR PASS AND RETURN
```

```
                              3661
                              3662
        064.071  000          3663  GRTBLK  DB      0                  BLOCK NUMBER CONTAINING GRT
        064.072  000          3664  DIRBLK  DB      0                  BLOCK NUMBER OF FIRST DIRECTORY BLOCK
        064.073  000          3665  DIRLBLK DB      0                  BLOCK NUMBER OF LAST DIRECTORY BLOCK
                              3666
                              3667
                              3668
                              3669
                              3670  **      DISK LABEL
                              3671
        064.074               3672  LABEL   DS      0
        000.000               3673          ERRNZ   *-LABEL-LAB.SER
        064.074  000          3674          DB      0                  SERIAL NUMBER
        000.000               3675          ERRNZ   *-LABEL-LAB.IND
        064.075  000 000      3676          DW      0                  CREATION DATE
        000.000               3677          ERRNZ   *-LABEL-LAB.DIS
        064.077  000 000      3678          DW      0                  INDEX OF FIRST DIRECTORY SECTOR
        000.000               3679          ERRNZ   *-LABEL-LAB.GRT
        064.101  000 000      3680          DW      0                  GRT TABLE INDEX
        000.000               3681          ERRNZ   *-LABEL-LAB.SPG
        064.103  002          3682          DB      2                  TWO SECTORS PER GROUP
        000.000               3683          ERRNZ   *-LABEL-LAB.VLT
        064.104  000          3684          DB      LAB.DAT            VOLUME TYPE
        000.000               3685          ERRNZ   *-LABEL-LAB.VER
        064.105  026          3686          DB      VERS               VERSION OF INIT17 TO INIT DISK
                              3687
        064.106  000 000 000  3688          DB      0,0,0,0,0,0,0      UNUSED
        000.000               3689          ERRNZ   *-LABEL-LAB.LAB
        064.115               3690          DS      60                 LABEL
        064.211  000 000      3691          DB      0,0                END OF LABEL
        064.213  040 040 040  3692          DB      '      '
        064.220  015 012      3693          DB      CR,LF
        064.222  123 131 123  3694          DB      'SYSTEM COPYRIGHT HEATH CO., 10/1977, 79/4'
        064.273  015 012      3695          DB      CR,LF
        064.275  040 102 131  3696          DB      ' BY JGL, 10/1977'
        064.315  057 147 143  3697          DB      '/sc '
        064.321               3698          DS      0                  END OF LABEL
                              3699
        064.321  050          3700  MAXTRK  DB      40                 MAX # OF TRACKS
                              3701
        064.322               3702  MEML    EQU     *                  MAX MEMORY FOR LOAD
                              3703
                              3704
                              3705
        064.322               3706  PATCH   DS      64                 PATCH AREA
                              3707
                              3708  **      DIRECTORY SECTOR
                              3709
        065.022               3710  IDDA    DS      0                  DIRECTORY SECTOR FWA
                              3711
        065.022               3712  IDDB    DS      22*DIRELEN         SPACE FOR REST OF BLOCK
        001.372               3713  IDDBL   EQU     *-IDDB
        067.014               3714          DS      1                  SPACE FOR 00 BYTE
        000.000               3715          ERRNZ   *-IDDA-DIS.ENL
        067.015               3716          DS      1                  LENGTH OF ENTRYS (DIRELEN)
```

```
     000.000                  3717          ERRNZ   *-IDDA-DIS.SEC
     067.016                  3718          DS      2               SECTOR NUMBER OF BEGINNING OF THIS BLOCK
     000.000                  3719          ERRNZ   *-IDDA-DIS.LNK
     067.020                  3720          DS      2               SECTOR NUMBER OF BEGINNING OF NEXT BLOCK
     000.000                  3721          ERRNZ   *-IDDA-512      SHOULD FILL BLOCK
                              3722
                              3723  **      RGTAB AND GRTAB MUST BE ON EVEN PAGE BOUNDARYS.
     067.022                  3724  LINE    DS      81
     000.120                  3725  LINEL   EQU     *-LINE-1
     067.022                  3726  ITLA    EQU     LINE            USE LINE FOR BUFFER
                              3727
     067.143                  3728          DS      *+255/256*256-*
                              3729
     070.000                  3730  GRTAB   DS      256             GRT TABLE
     071.000                  3731  RGTAB   DS      256             RGT TABLE
                              3732
     072.000                  3733  RMEML   DS      0               RUNNING MEMORY MAX LEN
                              3734
                              3735
     072.000                  3736          END
ASSEMBLY COMPLETE
 3736 STATEMENTS
    0 ERRORS DETECTED
 9558 BYTES FREE
```

```
$BCRC    044373     1204L     1246      1422
$CCO     062075     2090      2112      2126      2154      2180      2334      2357      2371      2527      2552      2556      3036L
$CDEHL   030216     3052E     3089
$CHL     030224     2243      3064E     3601
$CMP$    000001     3341E     3385      3394
$COMP    030060     1392E     1458
$CRLF    063304     3388      3575L
$DADA    030072     3364      3563E
$DADA.   030101     1402E     1559      3522
$DOS     062240     2053      3223L
$DTB     062112     3076L
$DTB1    062116     3080L     3083
$DTB2    062125     3088L     3093
$DTB3    062142     3090      3097L
$HLIHL   030211     3203E
$ICTT    044257      947      1121L     1122
$ICTT.   044271     1121      1126L     1980
$ICTT..  044317     1123      1148L     1982
$IDN     063134     2360      2559      3427L
$ITL     063213     2188      2374      2530      3500      3503L
$ITL.    063205     2129      2157      2337      3500L
$MCU     050007      948      1910L     3121
$MLU     062153     3116L     3148      3501
$MLU1    062156     3119L     3124
$MOVE    030252     1028      1371E     1666      2741      2882
$MU10    030324     3460      3483E
$MU86    031007     1063      1261E
$PDD     063145     3429      3446L
$PDD1    063151     3449L     3468
$RCHAR   063054     3152      3245      3297L     3298
$RSTALL  031047     3010E     3040
$RTL     062202     3146      3150E     3428      3504
$RTL.    062173     3146L
$RTL1    062203     3152L     3158
$RTL2    062235     3154      3175L
$SAVALL  031054     3024E     3036
$TDD     063223     2323      3518L
$TDD.    063221     3517L
$TYPC.   063071     3322L     3389      3533
$TYPCH   063065     3312L
$TYPEC.  045112     1177      1182      1288      1312L     1342      1570
$TYPET   045055      938       960       974       986       993      1043      1046      1083      1089      1095      1281L     1298
                    1498      1598      1949      1954      1960      1995      2011      2021
$TYPET.  045063     1282      1286L     1294
$TYPET1  045103     1290      1298L
$TYPL.   063112     2328      3366      3378E     3392
$TYPLN.  063074     3360L
$TYPTX   031136     2091      2113      2127      2151      2155      2181      2208      2300      2311      2317      2324      2329
                    2335      2358      2372      2389      2528      2553      2557      2564      2693      3223      3243      3338E     3625
$TYPTX.  031144     3340E
$UDD     031157     3192E
$WCHAR   063062     3301L
$WDR     031222     3287E
$WER     031241     3275E
.        000310     1822S     1823      2517S     2518      2519
..       000000     2272S     2273      2274
.ABUSS   040024      116E
.ALARM   002136       89E
```

```
        .ALEDS  040013       114E      1665     1667     1738
        .CHFLG  000060       596L
        .CLEAR  000055       593L
        .CLEARA 000056       594L
        .CLOSE  000046       586L      2050
        .CLRCO  000007       570L
        .CONSL  000006       569L      3039
        .CRC    002347        97E
        .CRCSUM 040027       117E
        .CTC    002172        91E
        .CTLC   000041       581L
        .CTLFLG 040011       113E
        .DECODE 000053       591L
        .DELET  000050       588L
        .DISMT  000061       597L
        .DLEDS  040021       115E
        .DLY    000053        86E      1692     1730     1892
        .DMNMS  000203       608L
        .DMOUN  000201       606L      3253
        .DOD    003122       100E
        .DODA   003356       102E
        .DSPMOD 040007       111E
        .DSPROT 040006       110E
        .DUMP   001374        88E
        .ERROR  000057       595L      2095
        .EXIT   000000       563L      2085
        .HORN   002140        90E      1669
        .IDENT  000000        85E
        .IOWRK  040002       108E
        .LINK   000040       580L
        .LOAD   001267        87E
        .LOADD  000062       598L
        .LOADO  000010       571L      3227     3230
        .MFLAG  040010       112E      1659     1662     1752
        .MONMS  000202       607L
        .MOUNT  000200       605L
        .NAME   000054       592L
        .OPENC  000045       585L
        .OPENR  000042       582L
        .OPENU  000044       584L
        .OPENW  000043       583L
        .PCHL   002264        93E
        .POSIT  000047       587L
        .PRINT  000003       566L
        .RCK    003260       101E
        .READ   000004       567L
        .REGI   040005       109E
        .REGPTR 040035       120E
        .RENAM  000051       589L
        .RESET  000204       609L
        .RNB    002331        96E
        .RNP    002325        95E
        .SCIN   000001       564L      3297
        .SCOUT  000002       565L      3301     3322     3576
        .SETTP  000052       590L      2057
        .SRS    002265        94E
        .START  040000       107E
        .SYSRES 000012       573L
```

```
.TICCNT 040033     119E    2240     2262     3594     3610
.TPERR  002205      92E
.TPERRX 040031     118E
.UIVEC  040037     121E
.VERS   000011     572L    2045
.WNB    003024      99E
.WNP    003017      98E
.WRITE  000005     568L
AAL     054230    2071     2298L
AAL1    055002    2307     2317L
AAL2    055066    2313     2334L    2342
ABR     046324    1525     1655E
ABR.A   047155    1664     1755L
ABR0    047036    1699L    1704
ABR0.1  046324    1659L
ABR0.3  046364    1673L    1725
ABR0.5  047021    1688     1690L
ABR1    047040    1700L    1708
ABR2    047103    1724L    1734
ABR3    047107    1716     1729L
ABR5    047124    1720     1738L
ABR5.1  047133    1741L    1744
ABS.COD 000010     368L     910
ABS.ENT 000006     366L
ABS.ID  000000     362L
ABS.LDA 000002     364L
ABS.LEN 000004     365L
AC.DLY  000156     270E    1691     1891
ADB     060147    2614     2670E
ADB1    060152    2677L    2690     2692
ADB2    060154    2678L    2683
ADB3    060171    2681     2688L
ADB4    060267    2684     2699L
AIO.CGN 041047     843L
AIO.CHA 041116     858L
AIO.CNT 041111     854L
AIO.CSI 041050     844L
AIO.DDA 041041     839E
AIO.DES 041055     848L    1445     1471
AIO.DEV 041057     849L
AIO.DIR 041062     852L     937     1027     1173     1455
AIO.DTA 041053     847L
AIO.EOF 041113     856L
AIO.EOM 041112     855L
AIO.FLG 041043     840L
AIO.GRT 041044     841L
AIO.LGN 041051     845L
AIO.LSI 041052     846L
AIO.SPG 041046     842L
AIO.TFP 041114     857L
AIO.UNI 041061     850L    2067
AIO.VEC 041040     838L
AMW     052317    2063     2151L
AMW1    052362    2154L    2164
AMW2    053101    2162     2165L
BELL    000007     888E    1044     1047     1084     1084     1090     1090     1096     1097     1499     1500     1950
                  1950     2092     2209     2694     2694     3224     3626     3626
BGT     060324    2641     2738L
```

```
BGT2     061035     2797L     2803
BGT3     061045     2799      2802L
BKSP     000010      890E
BLABEL   051010     1029      1037      1062      1616      1624      2034E     2035
BOOEND   064010     2036E
BOOT.P   000001      818E      977       996
BUFF     052010     1418      1444      1451      1470      1941      2035E     2036
C.DSYN   000375      218E
C.STX    000002      892E
C.SYN    000026      891E
CB.CLI   000100       55E       70
CB.MTL   000040       54E
CB.SPK   000200       56E
CB.SSI   000020       53E
CDB.H84  000001      761E     1127      1149      1314      1532      1841
CDB.H85  000000      760E     1522
CDC      050020      988      1930L
CDC1     050043     1940L     2007
CDC2     050110     1945      1954L     1988
CDC3     050240     1981      1985L
CDC4     050367     1984      2021L
CDCA     051000     1931      1966      1969      2013      2025L
CDCB     051002     1932      1962      1965      1997      2000      2026L
CDCD     051004     1936      1940      1956      1976      1978      2004      2027L
CDCE     051006     1934      1963      1967      1971      1979      2028L
CGT      063312     3594L     3646      3653
CGT1     063315     3596L     3599
CGT2     063330     3605L     3608
CIM      054136     2201      2203      2239L
CIM.CNT  000334     2235E     2271      2272
CIM.DLY  000002     2234E     2235      2236      2237      2241
CIM.MAX  000346     2237E     2274      2280
CIM.MIN  000320     2236E     2273      2278
CIM1     054164     2255L     2266
CIM2     054201     2259      2262L
CIMA     054227     2251      2255      2284L
CO.FLG   000001      738E     3038
CR       000015      884E      939       949      1299      1334      1346      2536      3693      3695
CRCSUM   046222     1632L
CS.FLG   000200      739E
CSC      045174     1417L     1964      1968      1972
CSL.CHR  000001      716E
CSL.ECH  000200      714E
CSL.WRP  000002      715E
CTLA     000001      899E
CTLB     000002      900E
CTLC     000003      901E     1983
CTLD     000004      902E     3153
CTLO     000017      903E
CTLP     000020      904E
CTLQ     000021      905E
CTLS     000023      906E
CTLZ     000032      907E
CTP.2SB  000010      724E     1530      1854      1855      1885      1887
CTP.BKM  000002      725E
CTP.BKS  000200      721E
CTP.MLI  000040      722E
CTP.MLO  000020      723E
CTP.TAB 000001       726E
D.ABORT 040141       668L
D.CDE   040160       673L
D.CON   040110       535L
D.DLY   040235       688L
D.DLYHS 040244       627L
D.DLYMO 040243       626L      2436
D.DRVTB 040251       632L      2299    2619    2632    2643
D.DTS   040163       674L
D.DVCTL 040242       624L      2423    2470
D.E.CHK 040267       643L
D.E.HCK 040270       644L
D.E.HSY 040266       642L
D.E.MDS 040265       641L
D.E.TRK 040272       646L
D.E.VOL 040271       645L
D.ERR   040265       640L
D.ERRL  040273       647L
D.ERRT  040232       687L
D.HECNT 040261       634L
D.LPS   040177       678L
D.MAI   040171       676L      2484
D.MAO   040174       677L
D.MOUNT 040133       666L
D.OECNT 040264       636L
D.OPR   040273       651L
D.OPW   040275       652L
D.RAM   040240       538L       619     654
D.RAML  000037       654E
D.RDB   040202       679L
D.READ  040147       670L
D.READR 040152       671L
D.SDP   040205       680L
D.SDT   040166       675L
D.SECNT 040262       635L
D.STS   040210       681L
D.STZ   040213       682L
D.SYDD  040130       665L
D.TRKPT 040245       629L
D.TS    040241       622L
D.TT    040240       621L
D.UDLY  040216       683L
D.VEC   040130       537L       663
D.VOLPT 040247       630L
D.WNB   040227       686L      2445    2447    2449    2451    2453    2461    2463    2467    2468    2469
D.WRITE 040155       672L
D.WSC   040221       684L
D.WSP   040224       685L      2441    2457
D.XIT   040144       669L
D.XOK   040136       667L
DC.ABT  000007       341L      1613    2069    2410    2489
DC.CLO  000006       340L
DC.LOD  000011       343L
DC.MAX  000012       344L
DC.MOU  000010       342L      1627
DC.OPR  000003       337L
DC.OPU  000005       339L
DC.OPW  000004       338L
```

```
  DC.REA  000000       334L       1492     1943
  DC.REC  000002       336L       1618     2305
  DC.WRI  000001       335L       2497     2623     2636     2647     2886     2901
  DDF.BOL 000011       460E
  DDF.BOO 000000       459L
  DDF.LAB 000011       461L       1617     2304     2635
  DDF.RBT 000012       462L       2605     2606     2622     2743     2925     2927
  DDF.USR 000014       463L
  DEV.DDA 000004       424L
  DEV.DVG 000016       436L
  DEV.DVL 000014       435L
  DEV.FLG 000006       425L
  DEV.JMP 000003       423L
  DEV.MNU 000011       432L
  DEV.MUM 000010       431L
  DEV.NAM 000000       415L
  DEV.RES 000002       419L
  DEV.SPG 000007       430L
  DEV.UNT 000012       433L
  DEVELEN 000017       438E
  DF.CLR  000376       377E       2892     2950
  DF.DI   000040       194E
  DF.DS0  000002       190E       2192     2421     2426
  DF.DS1  000004       191E
  DF.DS2  000010       192E
  DF.EMP  000377       376E       2827
  DF.HD   000001       184E       2250     2257     3597     3606
  DF.MO   000020       193E       2192     2421     2426
  DF.SD   000010       187E
  DF.ST   000100       195E
  DF.TO   000002       185E
  DF.WG   000001       189E       2426
  DF.WP   000004       186E
  DF.WR   000200       196E
  DIF.CNT 000020       491E       1055     2922     2937
  DIF.LOC 000100       489E       2922     2937     2946
  DIF.SYS 000200       488E       2922     2937     2946
  DIF.WP  000040       490E       2922     2937     2946
  DIR.ALD 000025       392L
  DIR.CLU 000015       385L       2919
  DIR.CRD 000023       391L       2930
  DIR.EXT 000010       380L       2915
  DIR.FGN 000020       388L       1057     2924
  DIR.FLG 000016       386L       1052     1057     2921
  DIR.LGN 000021       389L       2926
  DIR.LSI 000022       390L       2928
  DIR.NAM 000000       379L       1173     2913
  DIR.PRO 000013       381L       2917
  DIR.VER 000014       382L
  DIRBLK  064072      2714        2758     3664L
  DIRELEN 000027       394E        402      852     1462     2832     2952     2955     2981     3712
  DIRIDL  000015       383E       1026     1036
  DIRLBLK 064073      2786        2874     3665L
  DIRSTRT 000001      2715        2754E    2759
  DIS.ENL 001373       406L       2833     3715
  DIS.ENT 000000       401E       1451
  DIS.LNK 001376       408L       1470     2846     2903     3719
  DIS.SEC 001374       407L       2835     2885     2900     3717
  DM.MR    000000        60E
  DM.MW    000001        61E
  DM.RR    000002        62E
  DM.RW    000003        63E
  DOS.     063025      3234       3237     3240     3253L
  DOS1     063013      3245L      3247
  DOSA     063035      3239       3260L
  DOSB     063042      3236       3261L
  DOSC     063047      3233       3262L
  DP.DC    000177       182E      2193     2249     2256     2422     2427     2471     3596     3605
  DR.IM    000001       420E
  DR.PR    000002       421E
  DT.CR    000002       427E
  DT.CW    000004       428E
  DT.DD    000001       426E
  DV.EL    000000       416E
  DV.NU    000001       417E
  EC.CNA   000004       138L
  EC.DDA   000027       157L
  EC.DIF   000017       149L
  EC.DIW   000035       163L
  EC.DNI   000045       171L
  EC.DNR   000046       172L
  EC.DNS   000005       139L
  EC.DSC   000047       173L
  EC.EOF   000001       135L
  EC.EOM   000002       136L
  EC.FAO   000031       159L
  EC.FAP   000026       156L
  EC.FL    000030       158L
  EC.FNF   000014       146L      1475
  EC.FNO   000011       143L
  EC.FNR   000034       162L
  EC.FOD   000043       169L
  EC.FUC   000013       145L
  EC.ICN   000016       148L
  EC.IDN   000006       140L
  EC.IFC   000020       150L
  EC.IFN   000007       141L
  EC.ILC   000003       137L
  EC.ILO   000040       166L
  EC.ILR   000012       144L
  EC.ILV   000037       165L
  EC.IOI   000052       176L
  EC.IS    000032       160L
  EC.NCV   000050       174L      2078
  EC.NEM   000021       151L
  EC.NOS   000051       175L
  EC.NPM   000044       170L
  EC.NRD   000010       142L
  EC.NVM   000042       168L      3255
  EC.OTL   000053       177L
  EC.RF    000022       152L
  EC.UNA   000036       164L
  EC.UND   000015       147L
  EC.UUN   000033       161L
  EC.VPM   000041       167L
  EC.WF    000023       153L
  EC.WP    000025        155L
  EC.WPV   000024        154L
  ENL      000212        897E      2125      2152      2210      2301      2312      2373      2390      2555      2565      2694      3224
                        3626
  ERROR    051136       2054       2058      2080      2089L
  ESC      000033        895E
  EXIT     051133       2064       2084L     2132      2134      3627
  EXIT.    051134       2085L      2097
  FCU      046000        932       1512L
  FCU0     046046       1528       1532L
  FCU1     046050       1524       1538L
  FF       000014        898E
  FOV      060013       2075       2599E
  FOV1     060020       2607L      2610
  FT.ABS   000000        352E       913
  FT.BAC   000003        355E
  FT.PIC   000001        353E
  FT.REL   000002        354E
  GBL      056346       2074       2510E
  GBL1     056353       2513L      2516
  GBL1.5   056364       2520L      2523
  GBL2     056373       2525L      2539
  GBL3     057110       2545L      2548
  GBL4     057117       2526       2533      2535      2537      2552L
  GBL5     057272       2556L      2566      2587
  GBL6     057323       2564L      2573
  GBL7     057361       2563       2568L
  GRTAB    070000       2645       2740      2743      2757      2796      2842      3730L
  GRTBLK   064071       2701       2787      2863      3663L
  GVI      055224       2072       2356E     2361      2364      2366
  GVI1     055340       2371L      2391
  GVI2     056035       2381L      2388
  GVI3     056071       2385       2393L     2396
  HOS.SPG  000002        456E
  I.CONFL  000004        741E       742      3037
  I.CONTY  000001        728E       729
  I.CONWI  000003        734E       735
  I.CSLMD  000000        718E
  I.CUSOR  000002        731E       732
  ICTT1    044301       1132L
  ICTT2    044310       1128       1141L
  ICTT3    044334       1150       1160L
  IDD      061053       2652       2825L
  IDD2     061075       2835L      2906
  IDD4     061257       2849       2898L
  IDD5     061275       2894       2903L
  IDDA     065022       2833       2835      2846      2884      2885      2899      2900      2903      3710L     3715      3717      3719
                        3721
  IDDB     065022       2881       2973      3712L     3713
  IDDBL    001372       2881       2974      3713E
  IDDC     061306       2880       2912L     2913      2915      2917      2919      2921      2924      2926      2928      2930      2954
  IDDC2    061331       2854       2855      2931L
  IDDC3    061355       2864       2865      2867      2939L
  IDDC4    061360       2856       2857      2940L
  IDDC5    062004       2873       2875      2877      2948L
  IDDC6    062007       2858       2859      2949L
  IDDCL    000134       2879       2881      2954E     2955
  IDDD     062042       2826       2847      2957L
```

```
IDS       056101   2073     2410L
IDS1      056116   2420L    2485
IDS2      056150   2435L    2475
IDS3      056231   2460L    2465
IDS4      056323   2483     2489L
IDSA      056172   2413     2444E    2480
INIT      051010    916     2044E
INIT0     051064   2059     2065L
INIT1     051125   2046     2048     2078L
IP.PAD    000360     46E
ITLA      067022   3427     3503     3726E
LAB.DAT   000000    475E    1031     3684
LAB.DIS   000003    471L    1037     2719     2834     2868     3677
LAB.GRT   000005    472L    2646     2706     3679
LAB.IND   000001    470L    2630     3675
LAB.LAB   000021    482L     483     2327     2376     3689
LAB.LBL   000074    483E    2326
LAB.NOD   000002    477E
LAB.SER   000000    469L    1624     2319     2367     2479     2618     2642     3673
LAB.SPG   000007    473L    1062     3681
LAB.SYS   000001    476E    1034
LAB.VER   000011    480L    3685
LAB.VLT   000010    479L    1029     3683
LABEL     064074   2367     2376     2479     2618     2630     2634     2642     2646     2706     2719     2834     2868
                   3672L    3673     3675     3677     3679     3681     3683     3685     3689
LDE..     045212   1038     1442L    1474
LDE3      045233   1455L    1466
LEP       043151    980     1025L
LEP1      043262   1039     1052L
LEP3      043307   1067L
LEP4      043343   1056     1083L
LEP5      044032   1032     1089L
LEP6      044131   1035     1085     1095L
LF        000012    885E     939      939     1299     3693     3695
LINE      067022   3724L    3725     3726
LINEL     000120   3725E
M.FOX     000303     80E
M.PAM8    000021     79E
MAXTRK    064321   2481     3700L
MEML      064322    915     3702E
MI.JMP    000303     31E
MSD       046155    933      979     1613L
NL        000012    896E     897     2092     2094     2114     2115     2116     2117     2118     2120     2122     2124
                   2128     2152     2156     2182     2183     2184     2185     2186     2209     2210     2301     2325     2336
                   2359     2529     2554     2555     3157     3224     3224     3244     3246     3575
NUL2      000000    887E
NULL      000200    886E
OP.CTL    000360     47E
OP.DIG    000360     48E
OP.SEG    000361     49E
OVL.COD   000000    508L
OVL.ENS   000010    513E
OVL.ENT   000004    510L
OVL.FLB   000006    511L
OVL.IN    000001    785E
OVL.NUM   000014    787E
OVL.RES   000002    786E
OVL.SIZ   000002    509L
```

```
  OVL.UCS 000200        788E
  OVL0    000000        519L      3226
  OVL1    000001        520L      3229
  PATCH   064322       3706L
  PDS     062043       2828       2893     2973L
  PDS1    062051       2975L      2989
  PDS2    062056       2982L      2986
  PIC.COD 000006        879L      1079
  PIC.ID  000000        874L
  PIC.LEN 000002        876L      1071
  PIC.PTR 000004        877L
  PRS     051172       2055       2112L    2136
  PUBLIC  000001          1E       966
  QUOTE   000047        893E
  R.DLY   035303        867E      2219     2221     2425     2429
  R.WHD   036235        865E      2196     2437     3645
  R.WNH   036271        866E      2195     2197     3644     3658
  READD   045276       1070       1078     1446     1492L
  READERR 045304       1498L      1620     1629
  RESTART 051056       2063L      2076     2340     2695
  RGTAB   071000       2302       2319     2327     2511     2543     2584     2604     2612     2621     2676     2688     2739
                       3731L
  RMEML   072000       2056       3733L
  RMI     053104       2068       2180L    2211
  ROMBOOT 030000        530E      1501
  RUBOUT  000177        889E
  S.BAUD  040344        762L      1538     1876
  S.BOOTF 041034        817L       976      978      995      997
  S.CAADR 040333        745L
  S.CACC  041006        801L
  S.CCTAB 040335        746L
  S.CDB   040343        759L      1126     1148     1313     1539     1840
  S.CFWA  040352        769L
  S.CODE  041007        802L
  S.CONFL 040332        743L
  S.CONTY 040327        730L      1529     1531     1853     1884
  S.CONWI 040331        736L
  S.CSLMD 040326        719L       729      732      735      742     2052
  S.CUSOR 040330        733L
  S.DATC  040310        701L      2629     2853
  S.DATE  040277        700L
  S.DCS   041033        815L
  S.DDDTA 040366        780L
  S.DDGRP 040364        777L
  S.DDLDA 040360        775L
  S.DDLEN 040362        776L
  S.DDOPC 040370        781L
  S.DFWA  040354        770L
  S.DIREA 041016        809L
  S.DLINK 040346        767L
  S.FASER 041013        808L
  S.FCI   041021        810L
  S.GRT0  024000        526E
  S.GRT1  025000        527E
  S.GRT2  026000        528E
  S.GUP   041027        812L
  S.HIMEM 040316        703L
  S.INT   040343        540L       755
```

```
S.JUMPS  041010      806L
S.MOUNT  041032      814L
S.OFWA   040350      768L
S.OMAX   040324      709L
S.OSN    041004      797L
S.OVLE   041000      794L
S.OVLFL  040371      790L
S.OVLS   040376      793L
S.OVSTK  041035      822L
S.RFWA   040356      771L
S.SCI    041024      811L
S.SCR    041120      860L
S.SDD    041010      807L
S.SOVR   041146      542L       544
S.SSN    041002      796L
S.SYSM   040320      705L
S.TIME   040312      702L
S.UCSF   040372      791L
S.UCSL   040374      792L
S.USRM   040322      707L
S.VAL    040277      539L       698
SOBOOT   042200      930L      2032      2495
SOBOOT2  043042      950        952       974L
SOBOOT3  043062      979L       998
SOBOOT4  043105      954        986L
SOBOOT5  043124      956        993L
SOBOOTL  007000     2032E      2494
SOBOOTX  042211      935L       970       989      1048      1091      1098      2017      2023
SB.ORG   047000      126E      1068      1071      1077      1079      1640
SB.OVMX  014000      127E
SC.ACE   000350      269E      1141      1160      1328      1332      1513      1519      1520      1674      1676      1678      1681
                    1684       1690      1693      1695      1699      1700      1717      1731      1732      1750      1868      1873      1875
                    1878       1880      1883      1890      1893      1894      1896
SC.UART  000372      230E      1132      1154      1319      1323      1514      1847      1848      1849      1850      1852      1861
                    1863
SCU      047252     1540       1840L
SCU1     047322     1842       1868L     1870
SPG      000002       33E      2512      2517      2605      2606      2606      2674      2688      2704      2743      2756      2836
                    2925       2927
STACK    042200      546E       935      2065
STACKL   001032      544E
SYDD     040130      536E      1493      1614      1619      1628      1944      2306      3623
SYDD.    063343     2070       2411      2490      2498      2624      2637      2648      2887      2902      3623L
SYSCALL  000377      556E      2045      2050      2057      2085      2095      3039      3227      3230      3253      3297      3301
                    3322       3576
T110     047236     1792       1808L
T1200    047212     1773       1786L
T150     047242     1809       1814L
T1800    047232     1787       1804L
T19200   047202     1769       1778L
T300     047216     1774       1790L
T3600    047226     1784       1800L
T4800    047206     1770       1782L
T600     047176     1764       1772L
T7200    047222     1783       1796L
T75      047246     1810       1818L
T9600    047172     1763       1768L
TAB      000011      894E      2114      2114      2114      2115      2115      2115      2116      2116      2118
```

```
 TABLE   047166      1671       1760L    1779     1780     1788     1791     1797     1798     1801     1802     1805     1806
                     1815       1816     1819     1820     1823
 TDD1    063224      3519L      3542
 TDD2    063243      3529L      3531
 TDDA    063272      3520       3546E
 TFN     044341      1045       1173L
 TFN1    044360      1175       1180L    1185
 TPL1    063126      3390L
 TTDD    046064      1555L      1597     1959
 TTDD.   046062      1554L
 TTDD1   046065      1556L      1579
 TTDD2   046104      1566L      1568
 TTDDA   046133      1557       1583E
 TTDDCR  046145      1597L      1975     2002     2016
 TYPEC1  045123      1319L      1321
 TYPEC2  045140      1315       1328L    1330
 TYPEC3  045152      1324       1334L
 TYPEC4  045157      1340L      1345
 UC.2SB  000004       295E      1689     1887     1888
 UC.5BW  000000       291E
 UC.6BW  000001       292E
 UC.7BW  000002       293E
 UC.8BW  000003       294E      1518     1521     1687     1689     1889
 UC.BI   000020       314E
 UC.CTS  000020       323E
 UC.DCS  000001       319E
 UC.DDR  000002       320E
 UC.DLA  000200       300E      1677     1877
 UC.DR   000001       310E      1142     1707
 UC.DRL  000010       322E
 UC.DSR  000040       324E
 UC.DTR  000001       303E
 UC.EDA  000001       281E
 UC.EPS  000020       297E
 UC.FE   000010       313E      1707     1715
 UC.IID  000006       288E
 UC.IIP  000001       287E
 UC.LOO  000020       307E      1675     1694     1874     1895
 UC.MSI  000010       284E
 UC.OR   000002       311E      1701
 UC.OU1  000004       305E
 UC.OU2  000010       306E
 UC.PE   000004       312E      1707
 UC.PEN  000010       296E
 UC.RI   000100       325E
 UC.RLS  000200       326E
 UC.RSI  000004       283E
 UC.RTS  000002       304E
 UC.SB   000100       299E
 UC.SKP  000040       298E
 UC.TER  000004       321E
 UC.THE  000040       315E      1329
 UC.TRE  000002       282E
 UC.TSE  000100       316E      1869
 UCI.ER  000020       252E      1862
 UCI.IE  000002       254E
 UCI.IR  000100       250E      1851
 UCI.RE  000004       253E      1862
 UCI.RO   000040       251E
 UCI.TE   000001       255E     1862
 UDR      000000       227E     1154      1323
 UF.FCT   000100       211E
 UF.RDA   000001       208E
 UF.ROR   000002       209E
 UF.RPE   000004       210E
 UF.TBM   000200       212E
 UMI.16X  000002       245E     1860
 UMI.1B   000100       235E     1855      1860
 UMI.1X   000001       244E
 UMI.2B   000300       237E     1855
 UMI.64X  000003       246E
 UMI.HB   000200       236E
 UMI.L5   000000       240E
 UMI.L6   000004       241E
 UMI.L7   000010       242E
 UMI.L8   000014       243E     1860
 UMI.PA   000020       239E
 UMI.PE   000040       238E
 UNT.DIS  000005       447L
 UNT.FLG  000000       444L
 UNT.GRT  000001       445L
 UNT.GTS  000003       446L
 UNT.SIZ  000007       449E
 UO.CLK   000001        72E
 UO.DDU   000002        71E     1661
 UO.HLT   000200        69E
 UO.NFR   000100        70E
 UP.DP    000174       202E
 UP.FC    000175       203E      931      2414
 UP.SC    000176       205E
 UP.SR    000176       206E
 UP.ST    000175       204E
 UR.DLL   000000       276E     1681      1880
 UR.DLM   000001       278E     1684      1883
 UR.IER   000001       280E     1513      1674      1873
 UR.IIR   000002       286E
 UR.LCR   000003       290E     1519      1520      1678      1690      1878      1890
 UR.LSR   000005       309E     1141      1328      1700      1732      1868
 UR.MCR   000004       302E     1676      1693      1695      1875      1894      1896
 UR.MSR   000006       318E
 UR.RBR   000000       272E     1160      1699      1717      1731      1750      1893
 UR.THR   000000       274E     1332
 USERFWA  042200       547E      910       914       915
 USR      000001       228E     1132      1319      1514      1847      1848      1849      1850      1852      1861      1863
 USR.FE   000040       259E
 USR.OE   000020       260E
 USR.PE   000010       261E
 USR.RXR  000002       263E     1133
 USR.TXE  000004       262E
 USR.TXR  000001       264E     1320
 VERS     000026       554E     2047      2115      2115      3686
 VOLSIZ   001220        34E     2688
 W1S      054124      2194      2218L
 WIH      064034      2430      3643L
 WIH1     064040      3645L     3649
 WIH2     064054      3653L     3656

15924 BYTES FREE
```