```
                ;       ******************************************
                ;       * COMPUTIME MONITOR, VERSION III.6       *
                ;       * REQUIRES SBC-880 CPU BOARD             *
                ;       *  AND UFDC-1 FLOPPY DISK CONTROLLER     *
                ;       * PROVIDES FOR USE OF CT256K-1           *
                ;       *  DYNAMIC MEMORY BOARD                  *
                ;       * PROVIDES MULTIPLE FORMAT CAPABILITY    *
                ;       *  IBM COMPATIBLE FORMATS                *
                ;       *  8" OR 5 1/4" DISK                     *
                ;       *  SINGLE OR DOUBLE DENSITY              *
                ;       *  SINGLE OR DOUBLE SIDE                 *
                ;       *  128, 256, 512, OR 1024 BYTE SECTORS   *
                ;       *  1K, 2K, OR 4K ALLOCATION BLOCK SIZE   *
                ;       *  VARIABLE SECTOR SKEW                  *
                ;       * REQUIRES 2.2 CPM                       *
                ;       *  CONFIGURED FOR 60K OF RAM             *
                ;       * 2.2 CPM BIOS CONTAINED IN ROM AND      *
                ;       *  DISK PARAMETERS ARE OBTAINED FROM     *
                ;       *  TABLES ON DISK                        *
                ;       * PRINT DRIVER FOR CENTRONIX PARALLEL    *
                ;       *  INTERFACE AND CONSOLE DRIVER FOR      *
                ;       *  9600 BAUD TERMINAL ARE IN BIOS        *
                ;       * SBC880 PROCESSOR BOARD SWITCH SETTINGS*
                ;       *    SWITCH 1 POS 1,3,6 OFF              *
                ;       *             POS 2,4,5 ON               *
                ;       *    SWITCH 2 POS 1,2,3,4,6 OFF          *
                ;       *             POS 5 ON                   *
                ;       *    SWITCH 3 POS 1,2,3,4,5 OFF          *
                ;       *             POS 6 ON                   *
                ;       * WRITTEN BY R. D. CATILLER              *
                ;       * COPYRIGHT 1982 (C) COMPUTIME           *
                ;       ******************************************
                ;
                .PHEX
                .XLINK
                .PABS
                ;       MEMORY USED BY MONITOR
                ;
F800            BASE    =       0F800H  ;MONITOR BASE ADDRESS.
F7FF            STACK   =       BASE-1  ;MONITOR STACK
                ;
                ;       CONSTANTS FOR MONITOR
                ;
000D            CR      = 0DH           ;ASCII CARRIAGE RET
000A            LF      = 0AH           ;ASCII LINE FEED
                ;
                ;       I/O PORTS ON CPU BOARD
                ;
0028            T0      =       028H
0029            T1      =       T0+1
002A            T2      =       T0+2
002B            TCTL    =       T0+3
002C            INOUT   =       T0+4
002E            CONDTA  =       T0+6
002F            CONCTL  =       T0+7
002F            CONSTS  =       CONCTL
                ;
                ;       I/O PORTS ON CT256K MEMORY BOARD
                ;
0088            MAP0    =       88H     ;MEMORY MAP REGISTERS
0089            MAP1    =       89H
008A            MAP2    =       8AH
008B            MAP3    =       8BH
0088            MEMSTS  =       88H     ;MEMORY STATUS PORT
                ;
                ;       DISK DEFINES
                ;
0016            VERS    =       22      ;CP/M VERSION NUMBER
003C            MSIZE   =       60      ;CP/M MEMORY SIZE
A000            BIAS    =       (MSIZE-20)*1024
D400            CCP     =       3400H+BIAS      ;BASE OF CCP
DC06            BDOS    =       CCP+806H        ;BASE OF BDOS
EA00            BIOS    =       CCP+1600H       ;BASE OF BIOS
0000            WRALL   =       0               ;WRT TO ALOC BLK
0001            WRDIR   =       1               ;WRT TO DIRECTORY
0002            WRUAL   =       2               ;WRT TO UNALOC BLK
EA03            WBOOTE  =       BIOS+3          ;WARM BOOT ENTRY
                ;
                ;DISK I/O PORTS
                ;
009C            DSTAT   =       9CH     ;DISK STATUS PORT
009C            DCMMD   =       DSTAT   ;DISK COMMAND PORT
009D            DTRCK   =       DSTAT+1 ;DISK TRACK PORT
009E            DSCTR   =       DSTAT+2 ;DISK SECTOR PORT
009F            DDATA   =       DSTAT+3 ;DISK DATA PORT
009B            DFLAG   =       9BH     ;STATUS REGISTER 1
009B            DCNTL   =       DFLAG   ;CONTROL REGISTER 1
                ;
                ;MEMORY USED BY CPM
                ;
0004            CURDSK  =       4       ;CURRENT DISK DRIVE
0080            TBUF    =       80H     ;DEFAULT CPM BUFFER
                ;
                ;       PROGRAM CODE BEGINS:
                ;
F800            .LOC    BASE
                ;
                ;LET US BEGIN
                ;
F800  C3 F81D           JMP     BEGIN           ;RESET JUMP LATCH
                ;
                ;       MONITOR SIGN-ON MESSAGE
                ;
F803  0D0A      MSG:    .BYTE   CR,LF
F805  434F4D505554      .ASCII  'COMPUTIME III.6'
0011            MSGL    =       .-MSG
                ;
                ;       MEMORY PARITY ERROR MESSAGE
                ;
```

```
F8D7  C690          ADI   90H
F8D9  27            DAA
F8DA  CE40          ACI   40H
F8DC  27            DAA
F8DD  4F            MOV   C,A
F8DE  C9            RET
                    ;
                    ; CONSOLE CARRIAGE RETURN &
                    ; LINE FEED ROUTINE.
                    ;
F8DF  E5    CRLF:   PUSH  H       ;SAVE HL
F8E0  0602          MVI   B,2     ;CRLF LENGTH
F8E2  CD F8B8       CALL  MESG    ;SEND CRLF
F8E5  E1            POP   H
F8E6  C9            RET
                    ;
                    ;CONSOLE STATUS TEST ROUTINE.
                    ;
F8E7  DB88  CSTS:   IN    MEMSTS  ;TEST FOR PAR ERR
F8E9  E6C0          ANI   0C0H
F8EB  FE80          CPI   80H
F8ED  2017          JRNZ  CSTS1   ;NO PAR ERR
F8EF  3A F3BF       LDA   PARFLG  ;1ST PAR ERR?
F8F2  FE00          CPI   0
F8F4  2010          JRNZ  CSTS1   ;NOT 1ST
F8F6  3C            INR   A       ;SET PAR FLG
F8F7  32 F3BF       STA   PARFLG
F8FA  E5            PUSH  H
F8FB  C5            PUSH  B
F8FC  21 F814       LXI   H,MSG1  ;DISPLAY PAR ERR
F8FF  0609          MVI   B,MSG1L
F901  CD F8BB       CALL  MESG1
F904  C1            POP   B
F905  E1            POP   H
F906  DB2F  CSTS1:  IN    CONCTL
F908  E602          ANI   02H
F90A  3EFF          MVI   A,0FFH
F90C  C0            RNZ
F90D  2F            CMA
F90E  C9            RET
                    ;
                    ;PRINT H&L ON CONSOLE
                    ;
F90F  7C    DISPHL: MOV   A,H
F910  CD F914       CALL  DISPB
F913  7D            MOV   A,L
F914  F5    DISPB:  PUSH  PSW
F915  0F            RRC
F916  0F            RRC
F917  0F            RRC
F918  0F            RRC
F919  CD F91D       CALL  HTA2
F91C  F1            POP   PSW
F91D  CD F8D5 HTA2: CALL  HTA
F920  18A9          JMPR  CO
                    ;
                    ;MAIN KEYBOARD ROUTINE
                    ;
F922  CD F99A MAININ: CALL  CI        ;GET INPUT
F925  4F            MOV   C,A       ;ECHO IT
F926  18A3          JMPR  CO
                    ;
                    ;MAIN PARAMETER GETTING ROUTINE
                    ;
F928  21 0000 GPARAM: LXI   H,0       ;CLEAR HL
F92B  CD F922 GPNEXT: CALL  MAININ    ;GET INPUT
F92E  47      GP1:  MOV   B,A       ;SAVE IT
F92F  FE20          CPI   ' '       ;TEST FOR SPACE
F931  C8            RZ              ;RETURN IF SPACE
F932  FE2C          CPI   ','       ;TEST FOR COMMA
F934  C8            RZ              ;RETURN IF COMMA
F935  FE0D          CPI   CR        ;TEST FOR CR
F937  C8            RZ              ;RETURN IF CR
F938  D630          SUI   '0'       ;TEST < 0
F93A  DA F8AE       JC    ERROR     ;INPUT ERROR
F93D  FE17          CPI   'G'-'0'   ;TEST IF > F
F93F  D2 F8AE       JNC   ERROR     ;INPUT ERROR
F942  FE0A          CPI   10        ;TEST FOR NUMBER
F944  3807          JRC   DONE      ;GO SAVE NUMBER
F946  D607          SUI   'A'-'9'-1 ;ADJUST LETTER
F948  FE0A          CPI   0AH       ;TEST FOR . THRU 9
F94A  DA F8AE       JC    ERROR     ;INPUT ERROR
F94D  29    DONE:   DAD   H         ;SHIFT HL 1 DIGIT
F94E  29            DAD   H
F94F  29            DAD   H
F950  29            DAD   H
F951  B5            ORA   L         ;OR L WITH DIGIT
F952  6F            MOV   L,A
F953  18D6          JMPR  GPNEXT    ;GET MORE INPUT
                    ;
                    ;GETS START & END ADDRESS AND DETERMINES LENGTH
                    ;
F955  CD F928 RANGE:  CALL  GPARAM    ;GET START ADDRESS
F958  FE0D          CPI   CR        ;TEST FOR CR
F95A  CA F8AE       JZ    ERROR     ;INPUT ERROR
F95D  54            MOV   D,H       ;PUT HL IN DE
F95E  5D            MOV   E,L
F95F  CD F928       CALL  GPARAM    ;GET END ADDRESS
F962  E5            PUSH  H         ;SAVE IT
F963  B7            ORA   A         ;END - START
F964  ED52          DSBC  D
F966  44            MOV   B,H       ;PUT LENGTH IN BC
F967  4D            MOV   C,L
F968  62            MOV   H,D       ;PUT START IN HL
F969  6B            MOV   L,E
F96A  D1            POP   D         ;PUT END IN DE
F96B  C9            RET
                    ;
F96C  CD F955 RANGE2: CALL  RANGE     ;GET 2 PARAMETERS
F96F  FE0D          CPI   CR        ;TEST FOR CR
```

```
F971  C2 F8AE          JNZ     ERROR
F974  C9               RET
                       ;
F975  CD F955  RANGE3: CALL    RANGE
F978  FE0D             CPI     CR
F97A  CA F8AE          JZ      ERROR
F97D  22 F3DC          SHLD    HLSTR
F980  C5       PARAM1: PUSH    B
F981  CD F928          CALL    GPARAM
F984  C1               POP     B
F985  FE0D             CPI     CR
F987  C2 F8AE          JNZ     ERROR
F98A  C9               RET
                       ;
F98B  E5       ENDTST: PUSH    H
F98C  B7               ORA     A
F98D  ED52             DSBC    D
F98F  E1               POP     H
F990  C9               RET
                       ;
F991  CD F975  SDL:    CALL    RANGE3
F994  54               MOV     D,H
F995  5D               MOV     E,L
F996  2A F3DC          LHLD    HLSTR
F999  C9               RET
                       ;
                       ;MAIN CONSOLE INPUT ROUTINE
                       ;
F99A  CD F8E7  CI:     CALL    CSTS
F99D  B7               ORA     A
F99E  28FA             JRZ     CI
F9A0  DB2E             IN      CONDTA
F9A2  E67F             ANI     7FH
F9A4  C9               RET
                       ;
                       ;PRINTER OUTPUT ROUTINE
                       ;
F9A5  DB2C     PRINT:  IN      INOUT
F9A7  E601             ANI     1
F9A9  20FA             JRNZ    PRINT
F9AB  79               MOV     A,C
F9AC  F680             ORI     80H
F9AE  D32C             OUT     INOUT
F9B0  E67F             ANI     7FH
F9B2  D32C             OUT     INOUT
F9B4  F680             ORI     80H
F9B6  D32C             OUT     INOUT
F9B8  C9               RET
                       ;
                       ; PRINTER STATUS ROUTINE
                       ;
F9B9  DB2C     PSTS:   IN      INOUT
F9BB  E601             ANI     1
F9BD  3EFF             MVI     A,0FFH
F9BF  C8               RZ
```

```
                       ;INPUT ERROR
F9C0  2F               CMA
F9C1  C9               RET
                       ;
                       ;GET 2 PARAMETERS
                       ;TEST FOR CR
                       ;INPUT ERROR
F9C2  CD F96C  DISP:   CALL    RANGE2          ;GET PARAMETERS
F9C5  7D               MOV     A,L             ;SAVE START
F9C6  E6F0             ANI     0F0H            ;SAVE BC
F9C8  6F               MOV     L,A             ;GET 3RD PARAMETER
F9C9  7B       DISP1:  MOV     A,E             ;RESTRRE BC
F9CA  E60F             ANI     0FH             ;TEST FOR CR
F9CC  2803             JRZ     DISP2           ;INPUT ERROR
F9CE  13               INX     D
F9CF  18F8             JMPR    DISP1
F9D1  CD F8C3  DISP2:  CALL    CRLFHL          ;SAVE HL
F9D4  CD F8C9  DISP3:  CALL    SPACE           ;HL - DE
F9D7  7E               MOV     A,M             ;GET DATA
F9D8  CD F914          CALL    DISPB           ;RESTORE HL
F9DB  7D               MOV     A,L             ;RETURN FLAGS
F9DC  E60F             ANI     0FH
F9DE  FE0F             CPI     0FH             ;GET 3 PARAMETERS
F9E0  2803             JRZ     DISP4           ;DEST TO DE
F9E2  23               INX     H               ;NEXT
F9E3  18EF             JMPR    DISP3           ;SOURCE TO HL
F9E5  CD F8C9  DISP4:  CALL    SPACE           ;BC = LENGTH
F9E8  CD F8C9          CALL    SPACE           ;DISPLAY A SPACE
F9EB  7D               MOV     A,L             ;BACK UP ADR
F9EC  E6F0             ANI     0F0H
F9EE  6F               MOV     L,A             ;TEST FOR INPUT
F9EF  7E       NEXTA:  MOV     A,M             ;GET DATA
F9F0  E67F             ANI     7FH             ;KILL PARITY
F9F2  FE20             CPI     ' '             ;TEST IF >= SPACE
F9F4  3002             JRNC    DISP6           ;TEST FURTHER
F9F6  3E2E     DISP5:  MVI     A,'.'           ;REPLACE WITH PERIOD
F9F8  FE7C     DISP6:  CPI     7CH             ;> LOWER CASE Z
F9FA  30FA             JRNC    DISP5           ;REPLACE IT
F9FC  4F               MOV     C,A             ;DISPLAY IT
F9FD  CD F8CB          CALL    CO
FA00  23               INX     H               ;STEP TO NEXT
FA01  7D               MOV     A,L             ;TEST FOR END OF LINE
FA02  E60F             ANI     0FH
FA04  20E9             JRNZ    NEXTA           ;DO NEXT
FA06  CD F98B          CALL    ENDTST          ;TEST FOR END
FA09  20C6             JRNZ    DISP2           ;NEXT LINE
FA0B  C3 F85A          JMP     START
                       ;
                       ;FILL MEMORY WITH A CONSTANT
                       ;
FA0E  CD F975  FILL:   CALL    RANGE3          ;GET 2 PARAMETERS
FA11  7D               MOV     A,L             ;PUT DATA IN A
FA12  2A F3DC          LHLD    HLSTR           ;START ADR TO HL
FA15  77               MOV     M,A             ;WRITE DATA AT START
FA16  54               MOV     D,H             ;COPY HL TO DE
FA17  5D               MOV     E,L
FA18  13               INX     D               ;DEST = SOURCE + 1
```

```
FA19  0B          DCX   B          ;ADJUST LENGTH
FA1A  EDB0        LDIR             ;WRITE DATA
FA1C  C3 F85A     JMP START
                  ;
                  ;GOTO USER PROGRAM WITH OPTIONAL BREAKPOINT
                  ;
FA1F  CD F980     GOTO: CALL  PARAM1
FA22  E9                PCHL               ;GOTO USER PROGRAM
                  ;
                  ;EXTEDED MEMORY TEST
                  ;
FA23  CD F96C     TEST: CALL  RANGE2       ;GET 2 PARAMETERS
FA26  44                MOV   B,H          ;SAVE START IN BC
FA27  4D                MOV   C,L
FA28  3E00              MVI   A,0          ;CLEAR I
FA2A  ED47        LOOP: STAI
FA2C  ED57        FILLIT: LDAI             ;BUILD DATA
FA2E  AD                XRA   L
FA2F  AC                XRA   H
FA30  77                MOV   M,A          ;WRITE DATA
FA31  23                INX   H            ;NEXT DATA
FA32  7C                MOV   A,H          ;TEST FOR END
FA33  BA                CMP   D
FA34  20F6              JRNZ  FILLIT       ;CONTINUE WRITING
FA36  7D                MOV   A,L          ;TEST FOR END
FA37  BB                CMP   E
FA38  20F2              JRNZ  FILLIT       ;CONTINUE WRITING
FA3A  60                MOV   H,B          ;RESTORE START
FA3B  69                MOV   L,C
FA3C  ED57        TEST1: LDAI              ;BUILD DATA
FA3E  AD                XRA   L
FA3F  AC                XRA   H
FA40  BE                CMP   M            ;COMPARE IT
FA41  C4 FA5B           CNZ   MERR         ;DISPLAY ERRORS
FA44  23                INX   H
FA45  7C                MOV   A,H          ;TEST FOR END
FA46  BA                CMP   D
FA47  20F3              JRNZ  TEST1        ;CONTINUE TEST
FA49  7D                MOV   A,L          ;TEST FOR END
FA4A  BB                CMP   E
FA4B  20EF              JRNZ  TEST1        ;CONTINUE TEST
FA4D  60                MOV   H,B          ;RESTORE START
FA4E  69                MOV   L,C
FA4F  DB2F              IN    CONCTL       ;TEST KEYBOARD
FA51  E602              ANI   02H
FA53  C2 F85A           JNZ   START        ;ABORT IF KEY PRESSED
FA56  ED57              LDAI               ;INCREMENT TALLY
FA58  3C                INR   A
FA59  18CF              JMPR  LOOP         ;ANOTHER PASS
                  ;
FA5B  C5          MERR: PUSH  B            ;SAVE BC
FA5C  F5                PUSH  PSW          ;SAVE DATA
FA5D  CD F8C3           CALL  CRLFHL       ;DISPLAY ADDRESS
FA60  F1                POP   PSW          ;DISPLAY DATA
FA61  CD F914           CALL  DISPB
FA64  CD F8C9           CALL  SPACE        ;DISPLAY A SPACE
FA67  7E                MOV   A,M          ;GET MEM DATA
FA68  CD F914           CALL  DISPB        ;DISPLAY IT
FA6B  C1                POP   B            ;RESTORE BC
FA6C  C9                RET                ;CONTINUE TESTING
                  ;
                  ;MOVE BLOCK OF MEMORY
                  ;
FA6D  CD F991     MOVE: CALL  SDL          ;SRC, DEST, LNGTH
FA70  EDB0              LDIR               ;DO MOVE
FA72  C3 F85A           JMP   START
                  ;
                  ;INPUT DATA FROM AN I/O PORT
                  ;
FA75  CD F980     INPUT: CALL PARAM1       ;GET PARAMETER
FA78  4D                MOV   C,L          ;PUT IO ADR IN C
FA79  ED78              INP   A            ;INPUT DATA TO A
FA7B  F5                PUSH  PSW          ;SAVE IT
FA7C  CD F8DF           CALL  CRLF         ;DISPLAY CRLF
FA7F  F1                POP   PSW          ;GET DATA
FA80  CD F914           CALL  DISPB        ;DISPLAY IT
FA83  C3 F85A           JMP   START
                  ;
                  ;OUTPUT DATA TO AN I/O PORT
                  ;
FA86  CD F96C     OUTPUT: CALL  RANGE2     ;GET 2 PARAMETERS
FA89  4D                MOV   C,L          ;OUTPUT DATA
FA8A  ED59              OUTP  E
FA8C  C3 F85A           JMP   START
                  ;
                  ;GET READY TO MOVE BIOS VECTORS
                  ;
FA8F  21 F033     BIOSMV: LXI   H,DIRBF
FA92  11 EA00     BIOSM1: LXI   D,BIOS
FA95  01 0033             LXI   B,BCDL
FA98  C9                 RET
                  ;
                  ;BOOT CPM AND RUN
                  ;
FA99  21 FBB7     BOOT:   LXI   H,TPLT     ;MOVE DOWN JMP VECTORS
FA9C  11 0000             LXI   D,0
FA9F  01 000B             LXI   B,8
FAA2  EDB0                LDIR
FAA4  21 FB64             LXI   H,BIOSCD         ;SET UP BIOS VECTO
FAA7  CD FA92             CALL  BIOSM1
FAAA  EDB0                LDIR
FAAC  31 F7FF     WBOOT:  LXI   SP,STACK         ;RESTORE STACK
FAAF  21 EA33             LXI   H,BEGDAT         ;CLEAR SCRATCH RAM
FAB2  11 EA34             LXI   D,BEGDAT+1
FAB5  01 09F1             LXI   B,DATSIZ
FAB8  3600                MVI   M,0
FABA  EDB0                LDIR
FABC  CD FA8F             CALL  BIOSMV           ;SAVE BIOS VECTORS
FABF  EB                  XCHG
FAC0  EDB0                LDIR
```

```
FAC2  21 FFBF            LXI   H,EPBASE   ;SET UP CPM TABLES
FAC5  11 F3DE            LXI   D,DPBASE
FAC8  01 0040            LXI   B,EPLGTH
FACB  EDB0               LDIR
FACD  21 D400            LXI   H,CCP      ;SET DMA ADR
FAD0  22 F3BD            SHLD  DMAAD
FAD3  3E02               MVI   A,2        ;SECTOR = 2
FAD5  32 F3B9            STA   SECTOR
FAD8  32 F3D2            STA   LUNIT
FADB  AF                 XRA   A
FADC  CD FCD9            CALL  LOGDS2     ;LOG ON DRIVE A
FADF  C2 F85A            JNZ   START      ;LOG ON ERROR
FAE2  DD2A F3D8          LIXD  CTBLP      ;GET TBL POINTER
FAE6  3E31               MVI   A,49       ;NO OF SECS
FAE8  32 F3D5            STA   BTSEC
FAEB  1619               MVI   D,25       ;8" SECS ON TRK 0
FAED  DB9B               IN    DFLAG
FAEF  E604               ANI   4          ;8" OR 5 1/4" ?
FAF1  2802               JRZ   WBOOT8     ;8"
FAF3  1611               MVI   D,17
FAF5  3A F3D5  WBOOT8:   LDA   BTSEC      ;SUBTRACT SECTORS
FAF8  92                 SUB   D
FAF9  32 F3D5            STA   BTSEC
FAFC  060A               MVI   B,10
FAFE  C5       BRWAGN:   PUSH  B
FAFF  CD FF8B            CALL  TST128     ;ADJUST SECTOR ADR
FB02  D39E               OUT   DSCTR
FB04  CD FF96            CALL  SETUP
FB07  D5                 PUSH  D          ;SAVE FOR RETRY
FB08  2A F3BD            LHLD  DMAAD      ;GET DMA ADR
FB0B  3A F3B9  BOOTRD:   LDA   SECTOR     ;TEST FOR SIDE 1
FB0E  E680               ANI   80H
FB10  3E98               MVI   A,98H      ;READ COMMAND
FB12  2802               JRZ   BTRW3      ;SIDE 0
FB14  3E9A               MVI   A,9AH      ;SELECT SIDE 1
FB16  32 F3BB  BTRW3:    STA   CMND
FB19  D39C               OUT   DCMMD
FB1B  01 809F  BTRW1:    LXI   B,(128*256)+DDATA
FB1E  EDB2               INIR
FB20  15                 DCR   D          ;LOOP CONTROL
FB21  20F8               JRNZ  BTRW1
FB23  0632               MVI   B,50
FB25  10FE     BWTDLY:   DJNZ  BWTDLY
FB27  3ED0               MVI   A,0D0H     ;FORCE INT COMMAND
FB29  D39C               OUT   DCMMD      ;EXECUTE IT
FB2B  060A               MVI   B,10       ;DELAY
FB2D  10FE     FRC1:     DJNZ  FRC1
FB2F  DB9C               IN    DSTAT      ;READ STATUS
FB31  D1                 POP   D          ;RESTORE LENGTH
FB32  CD FF31            CALL  EOJ
FB35  E69C               ANI   9CH
FB37  C1                 POP   B
FB38  2808               JRZ   WBOOTA     ;DONE, NO ERRORS
FB3A  10C2               DJNZ  BRWAGN     ;RETRY
FB3C  CD FC78  DERROR:   CALL  DERR1      ;DISPLAY ERROR
FB3F  C3 F85A            JMP   START
FB42  22 F3BD  WBOOTA:   SHLD  DMAAD      ;UPDATE DMAAD
FB45  3A F3D5            LDA   BTSEC      ;ALL SECS DONE?
FB48  B7                 ORA   A
FB49  CA FB72            JZ    GOCPM      ;DONE
FB4C  DDCB124E           BIT   1,18(X)    ;TEST FOR DOUBLE SIDE
FB50  3E81               MVI   A,81H      ;SIDE 1 SELECT
FB52  200B               JRNZ  WBOOT9     ;DOUBLE SIDE
FB54  0658               MVI   B,58H      ;STEP IN 1 TRACK
FB56  CD FF23            CALL  EOJA
FB59  21 F3B8            LXI   H,TRACK    ;TRK + 1
FB5C  34                 INR   M
FB5D  3E01               MVI   A,1        ;SECTOR = 1
FB5F  32 F3B9  WBOOT9:   STA   SECTOR
FB62  3A F3D5            LDA   BTSEC      ;BTSEC > SEC/TRK?
FB65  DDBE00             CMP   0(X)
FB68  57                 MOV   D,A        ;REMAINING SECS
FB69  DA FAF5            JC    WBOOT8     ;DO THE REST
FB6C  DD5600             MOV   D,0(X)     ;SECS PER TRK
FB6F  C3 FAF5            JMP   WBOOT8     ;DO NEXT TRK
FB72  CD FA8F  GOCPM:    CALL  BIOSMV     ;RESTORE BIOS VECTORS
FB75  EDB0               LDIR
FB77  21 0080            LXI   H,TBUF     ;DEFAULT CPM BUFFER
FB7A  22 F3D6            SHLD  SEKDMA     ;SET DMA ADDRESS
FB7D  3A 0004            LDA   CURDSK     ;LOG-ON DSK
FB80  4F                 MOV   C,A        ;SEND TO CPM
FB81  C3 D400            JMP   CCP        ;GO TO CP/M
                         ;
                         ;MONITOR BIOS JUMP VECTORS
                         ;
FB84  C3 FA99  BIOSCD:   JMP   BOOT       ;COLD BOOT
FB87  C3 FAAC            JMP   WBOOT      ;WARM BOOT
FB8A  C3 F8E7            JMP   CSTS       ;CONSOLE STATUS
FB8D  C3 F99A            JMP   CI         ;CONSOLE INPUT
FB90  C3 F8CB            JMP   CO         ;CONSOLE OUTPUT
FB93  C3 F9A5            JMP   PRINT      ;LIST DEVICE
FB96  C3 F8CB            JMP   CO         ;PUNCH DEVICE
FB99  C3 F99A            JMP   CI         ;READER DEVICE
FB9C  C3 FDA2            JMP   HOME       ;MOVE HEAD TO TRK 0
FB9F  C3 FCB3            JMP   SELDSK     ;SELECT DISK
FBA2  C3 FD69            JMP   SETTRK     ;SET TRACK NUMBER
FBA5  C3 FD6E            JMP   SETSEC     ;SET SECTOR NUMBER
FBA8  C3 FD9D            JMP   SETDMA     ;SET DMA ADDRESS
FBAB  C3 FDAB            JMP   HREAD      ;READ DISK
FBAE  C3 FDB7            JMP   HWRITE     ;WRITE DISK
FBB1  C3 F9B9            JMP   PSTS       ;LIST STATUS
FBB4  C3 FD73            JMP   SECTRN     ;SECTOR TRANSLATE
0033           BCDL      =     .-BIOSCD
                         ;
                         ;JUMPS AT START OF MEMORY AND IOBYTE
                         ;
FBB7  C3 EA03  TPLT:     JMP   WBOOTE
FBBA  BD       IOBYT:    .BYTE 0BDH
FBBB  00                 .BYTE 0
FBBC  C3 DC06            JMP   BDOS
```

```
                        ;
                        ;GET INPUT AND SEEK TRACK, (R = RANDOM SEEK TEST)
                        ;
FBBF  CD F980   TSEEK:  CALL    PARAM1          ;GET PARAMETER
FBC2  FE0D              CPI     CR              ;TEST FOR CR
FBC4  C2 F8AE           JNZ     ERROR           ;INPUT ERROR
FBC7  DD7E11            MOV     A,17(X)         ;GET LAST TRK
FBCA  3D                DCR     A
FBCB  BD                CMP     L               ;TEST L > LAST TRK
FBCC  7D                MOV     A,L
FBCD  DA F8AE           JC      ERROR           ;INPUT ERROR
FBD0  32 F3B8           STA     TRACK           ;SAVE TRACK
FBD3  CD FF4B   DOTRK:  CALL    SEEK2           ;SEEK TRACK
FBD6  C3 F85A           JMP     START
                        ;
                        ;READ ENTIRE TRACK TO MEMORY
                        ;
FBD9  CD F980   LDTRK:  CALL    PARAM1          ;GET PARAMETER
FBDC  AF                XRA     A               ;SELECT SIDE 0
FBDD  32 F3B9   LDTRK4: STA     SECTOR
FBE0  E5                PUSH    H
FBE1  CD FF96           CALL    SETUP           ;GET READY
FBE4  E1                POP     H
FBE5  0E9F              MVI     C,DDATA         ;POINT TO DATA PORT
FBE7  3EE4              MVI     A,0E4H          ;READ TRACK COMMAND
FBE9  D39C      LDTRK2: OUT     DCMMD           ;EXECUTE IT
FBEB  EDA2      LDTRK1: INI                     ;GET DATA
FBED  DB9B              IN      DFLAG           ;TEST FOR END
FBEF  E680              ANI     80H
FBF1  28F8              JRZ     LDTRK1          ;CONTINUE
FBF3  CD FFB7           CALL    DWAIT           ;DISABLE WAITS
FBF6  DDCB124E          BIT     1,18(X)         ;TEST FOR DBL SD
FBFA  CA F85A           JZ      START           ;NOT DBL SD
FBFD  3A F3B9           LDA     SECTOR          ;TEST IF SD 1 DONE
FC00  B7                ORA     A
FC01  C2 F85A           JNZ     START           ;DONE
FC04  3E80              MVI     A,80H           ;SELECT SIDE 1
FC06  18D5              JMPR    LDTRK4          ;READ SIDE 1
                        ;
                        ;ASSIGN DISK DRIVE AND SELECT
                        ;
FC08  21 0000   ASSIGN: LXI     H,0             ;CLEAR SELECT TBL
FC0B  22 F3B3           SHLD    DSKSEL
FC0E  22 F3B5           SHLD    DSKSEL+2
FC11  CD F980           CALL    PARAM1          ;GET DISK NO
FC14  7D                MOV     A,L
FC15  CD FCD9           CALL    LOGDS2          ;SELECT & READ TABLES
FC18  C3 F85A           JMP     START
                        ;
                        ;SUBSTITUTE AND EXAMINE MEMORY
                        ;
FC1B  CD F928   SUBS:   CALL    GPARAM          ;GET PARAMETER
FC1E  CD F8C3   SUBS1:  CALL    CRLFHL          ;DISPLAY ADDRESS
FC21  7E                MOV     A,M             ;WED DATA
FC22  CD F914           CALL    DISPB           ;DISPLAY IT
FC25  CD F8C9           CALL    SPACE           ;DISPLAY A SPACE
FC28  CD F99A           CALL    CI              ;GET INPUT
FC2B  FE0D              CPI     CR              ;TEST FOR CR
FC2D  CA F85A           JZ      START           ;DONE
FC30  FE20              CPI     ' '             ;TEST FOR SPACE
FC32  2003              JRNZ    SUBS2           ;TEST FURTHER
FC34  23        SUBS4:  INX     H               ;ADDRESS + 1
FC35  18E7              JMPR    SUBS1           ;CONTINUE DISPLAY
FC37  FE7F      SUBS2:  CPI     7FH             ;TEST FOR RUBOUT
FC39  2003              JRNZ    SUBS3           ;LOOK FOR PARAMETER
FC3B  2B                DCX     H               ;ADDRESS - 1
FC3C  18E0              JMPR    SUBS1           ;CONTINUE DISPLAY
FC3E  4F        SUBS3:  MOV     C,A             ;ECHO CHARACTER
FC3F  CD F8CB           CALL    CO
FC42  E5                PUSH    H               ;SAVE ADR
FC43  21 0000           LXI     H,0             ;CLEAR HL
FC46  CD F92E           CALL    GP1             ;GET PARAMETER
FC49  47                MOV     B,A             ;SAVE DELIMITER
FC4A  7D                MOV     A,L             ;PUT DATA IN A
FC4B  E1                POP     H               ;GET ADR
FC4C  77                MOV     M,A             ;STORE DATA
FC4D  78                MOV     A,B             ;TEST FOR CR
FC4E  FE0D              CPI     CR
FC50  CA F85A           JZ      START           ;DONE
FC53  18DF              JMPR    SUBS4           ;CONTINUE
                        ;
                        ;VERIFY BLOCK OF MEMORY AGAINST MEMORY
                        ;
FC55  CD F991   VERIFY: CALL    SDL             ;SRC, DEST, LNGTH
FC58  1A        VER1:   LDAX    D               ;GET DEST DATA
FC59  EDA1              CCI                     ;COMPARE TO SOURCE
FC5B  E2 F85A           JPO     START           ;DONE
FC5E  2B                DCX     H               ;ADJUST ADDRESS
FC5F  C4 FA5B           CNZ     MERR            ;DISPLAY ERRORS
FC62  23                INX     H               ;RESTORE ADDRESS
FC63  13                INX     D               ;DEST + 1
FC64  18F2              JMPR    VER1            ;CONTINUE
                        ;
FC66                    DERRM:
FC66  44534B54524B      .ASCII  'DSKTRKSCTSTSCMDDMA'
                        ;
                        ;DISPLAYS ALL DISK PARAMETERS
                        ;
FC78  C5        DERR1:  PUSH    B               ;SAVE REGISTERS
FC79  D5                PUSH    D
FC7A  E5                PUSH    H
FC7B  21 FC66           LXI     H,DERRM         ;POINT TO NAMES
FC7E  11 F3B7           LXI     D,DISKNO        ;POINT TO PARAMETER
FC81  CD F8DF           CALL    CRLF            ;DISPLAY CRLF
FC84  0E05              MVI     C,5             ;SET TALLY
FC86  0603      DERR2:  MVI     B,3
FC88  C5                PUSH    B               ;SAVE BC
FC89  CD F88B           CALL    MESG1           ;DISPLAY NAME
FC8C  CD F8C9           CALL    SPACE           ;DISPLAY SPACE
FC8F  E5                PUSH    H               ;SAVE HL
```

```
FC90  1A          LDAX    D       ;GET DATA
FC91  13          INX     D       ;STEP TO NEXT
FC92  CD F914     CALL    DISPB   ;DISPLAY DATA
FC95  CD F8C9     CALL    SPACE   ;DISPLAY SPACE
FC98  E1          POP     H       ;GET HL
FC99  C1          POP     B       ;GET BC
FC9A  0D          DCR     C       ;TALLY - 1
FC9B  20E9        JRNZ    DERR2   ;CONTINUE
FC9D  0603        MVI     B,3     ;DISPLAY NAME
FC9F  CD F8BB     CALL    MESG1
FCA2  CD F8C9     CALL    SPACE   ;DISPLAY SPACE
FCA5  2A F3BD     LHLD    DMAAD   ;LAST PARAMETER
FCA8  CD F90F     CALL    DISPHL  ;DISPLAY IT
FCAB  E1          POP     H       ;RESTORE REGISTERS
FCAC  D1          POP     D
FCAD  C1          POP     B
FCAE  3A F3BA     LDA     STATUS
FCB1  B7          ORA     A
FCB2  C9          RET
                  ;
                  ;SELECT DISK GIVEN BY REGISTER C
                  ;
FCB3  21 0000     SELDSK: LXI     H,0     ;ERROR RETURN CODE
FCB6  79                  MOV     A,C
FCB7  32 F3C9             STA     SEKDSK  ;STORE DSK NO.
FCBA  FE04                CPI     4       ;MUST BE 0,1,2,OR 3
FCBC  D0                  RNC             ;INVALID DSK NO.
FCBD  DDE5                PUSH    X       ;LOG ON DISK
FCBF  E5                  PUSH    H
FCC0  CD FCD9             CALL    LOGDS2
FCC3  E1                  POP     H
FCC4  C0                  RNZ             ;ERROR RETURN
FCC5  DDE1                POP     X
FCC7  3A F3C9             LDA     SEKDSK  ;GET DISK NO.
FCCA  6F                  MOV     L,A
FCCB  60                  MOV     H,B
FCCC  29                  DAD     H       ;*2
FCCD  29                  DAD     H       ;*4
FCCE  29                  DAD     H       ;*8
FCCF  29                  DAD     H       ;*16
FCD0  EB                  XCHG
FCD1  21 F3DE             LXI     H,DPBASE
FCD4  19                  DAD     D       ;HL=DPBASE+(DISKNO*16)
FCD5  C9                  RET
                  ;
FCD6  3A F3B7     LOGDSK: LDA     DISKNO
FCD9  4F          LOGDS2: MOV     C,A     ;SAVE DSK NO IN C
FCDA  3C                  INR     A
FCDB  DD21 EE33           LXI     X,DPO   ;BUILD TBL ADR IN X
FCDF  11 0080             LXI     D,128
FCE2  3D          DABLD1: DCR     A
FCE3  2804                JRZ     DABLD2
FCE5  DD19                DADX    D
FCE7  18F9                JMPR    DABLD1
FCE9  DD22 F3D8   DABLD2: SIXD    CTBLP
FCED  3EC0                MVI     A,0C0H
FCEF  B1                  ORA     C
FCF0  32 F3C0             STA     HSELCD  ;STORE SELECT CODE
FCF3  21 F3B3             LXI     H,DSKSEL ;SEE IF NEW DISK
FCF6  0600                MVI     B,0
FCF8  09                  DAD     B
FCF9  7E                  MOV     A,M
FCFA  B7                  ORA     A
FCFB  202D                JRNZ    DBPARM  ;NOT NEW DISK
FCFD  2F                  CMA
FCFE  77                  MOV     M,A     ;MARK AS USED
FCFF  3A F3C0             LDA     HSELCD  ;GET SEL CODE
FD02  D39B                OUT     DCNTL   ;SEL DRIVE
FD04  CD FF21             CALL    EOJB    ;RESTORE DRIVE
FD07  E680                ANI     80H     ;TEST FOR NOT REA
FD09  C2 FC78             JNZ     DERR1   ;NOT RDY ERROR
FD0C  060A                MVI     B,10    ;RETRY COUNT
FD0E  C5          NTAGN:  PUSH    B
FD0F  3E01                MVI     A,1     ;SELECT SEC 1
FD11  D39E                OUT     DSCTR
FD13  57                  MOV     D,A     ;SEC SIZE=128
FD14  2A F3D8             LHLD    CTBLP
FD17  3A F3C0             LDA     HSELCD  ;GET SELECT CODE
FD1A  D39B                OUT     DCNTL
FD1C  3E8C                MVI     A,8CH   ;READ OP
FD1E  CD FEFD             CALL    RDAT    ;READ TABLES
FD21  C1                  POP     B       ;RETRY COUNT
FD22  B7                  ORA     A       ;TEST FOR ERRORS
FD23  2805                JRZ     DBPARM  ;DONE, NO ERRORS
FD25  10E7                DJNZ    NTAGN   ;RETRY
FD27  C3 FC78             JMP     DERR1   ;DISP PARAMETERS
FD2A  DD6E0F      DBPARM: MOV     L,15(X) ;GET BYTES/SEC
FD2D  DD6610              MOV     H,16(X)
FD30  29                  DAD     H
FD31  7C                  MOV     A,H
FD32  32 F3C1             STA     HSTBLK  ;CPM SEC/PHY SEC
FD35  3D                  DCR     A
FD36  32 F3C3             STA     SECMSK  ;SECTOR MASK
FD39  DD4602              MOV     B,2(X)  ;GET BLK SHIFT FA
FD3C  3E01                MVI     A,1     ;CPM ALOC SIZE/12
FD3E  07          BLKCAL: RLC
FD3F  10FD                DJNZ    BLKCAL
FD41  32 F3C2             STA     BLKCNT
FD44  3A F3B7             LDA     DISKNO  ;GET DISK NO
FD47  21 F3D2             LXI     H,LUNIT ;POINT TO LAST UN
FD4A  BE                  CMP     M       ;SEE IF SAME
FD4B  77                  MOV     M,A     ;SAVE THIS UNIT
FD4C  C8                  RZ
FD4D  CD FF96             CALL    SETUP
FD50  21 F41E             LXI     H,IDSV
FD53  1601                MVI     D,1
FD55  3EC4                MVI     A,0C4H  ;READ ADR COMMAND
FD57  D39C                OUT     DCMND
FD59  32 F3BB             STA     CMND
FD5C  01 069F             LXI     B,(6*256)+DDATA
```

```
FD5F    CD FF05          CALL    READ2
FD62    3A F41E  IDRD1:  LDA     IDSV
FD65    D39D             OUT     DTRCK          ;SET THE TRACK
FD67    AF               XRA     A              ;RET, NO ERRORS
FD68    C9               RET
                         ;
                         ;SET TRACK GIVEN BY REGISTER C
                         ;
FD69    79       SETTRK: MOV     A,C
FD6A    32 F3CA          STA     SEKTRK
FD6D    C9               RET
                         ;
                         ;SET SECTOR GIVEN BY REGISTER C
                         ;
FD6E    79       SETSEC: MOV     A,C
FD6F    32 F3CB          STA     SEKSEC
FD72    C9               RET
                         ;
                         ;TRANSLATE THE SECTOR GIVEN BY BC USING THE
                         ;TRANSLATE TABLE GIVEN BY DE
                         ;
FD73    DDE5     SECTRN: PUSH    X              ;SAVE X
FD75    21 FFEC          LXI     H,0FFECH       ;TBL POINTER-20
FD78    19               DAD     D
FD79    E5               PUSH    H              ;POINTER TO X
FD7A    DDE1             POP     X
FD7C    DD7E12           MOV     A,18(X)        ;TEST FOR DBL SIDE
FD7F    E602             ANI     2
FD81    2811             JRZ     SECTR1         ;SINGLE SIDE
FD83    AF               XRA     A              ;DOUBLE SIDE
FD84    DD6E00           MOV     L,0(X)         ;GET CPM SEC/TRK
FD87    67               MOV     H,A
FD88    CB3D             SRLR    L              ;DIVIDE BY 2
FD8A    2D               DCR     L              ;SUBTRACT 1
FD8B    ED42             DSBC    B              ;SEC ON SD 0 OR 1?
FD8D    3005             JRNC    SECTR1         ;SECTOR ON SIDE 0
FD8F    7D               MOV     A,L            ;BUILD ADR FOR SD 1
FD90    2F               CMA
FD91    4F               MOV     C,A
FD92    3E80             MVI     A,80H          ;SIDE SELECT = SD 1
FD94    EB       SECTR1: XCHG                   ;HL=TRANS TBL
FD95    09               DAD     B              ;ADD SEC INDEX
FD96    B6               ORA     M              ;OR SIDE SELECT
FD97    6F               MOV     L,A            ;HL = TRANS(SECTOR)
FD98    2600             MVI     H,0
FD9A    DDE1             POP     X              ;RESTORE X
FD9C    C9               RET
                         ;
                         ;SET DMA ADDRESS GIVEN BY REGISTERS B AND C
                         ;
FD9D    ED43 F3D6SETDMA: SBCD    SEKDMA         ;SAVE THE ADDRESS
FDA1    C9               RET
                         ;
                         ;MOVE TO THE TRACK 00 POSITION OF CURRENT DRIVE
                         ;
FDA2    3A F3CE          LDA     HSTWRT         ;TEST FOR WRITE PEND
FDA5    B7               ORA     A
FDA6    C0               RNZ
FDA7    32 F3CF          STA     HSTACT         ;CLEAR HSTACT FLAG
FDAA    C9               RET
                         ;
                         ;DEBLOCKING ROUTINES
                         ;
FDAB    3E01     HREAD:  MVI     A,1            ;SET READOP=READ
FDAD    32 F3D1          STA     READOP
FDB0    3C               INR     A
FDB1    32 F3C4          STA     WRTYPE         ;TREAT AS UNALOC
FDB4    C3 FE15          JMP     ALLOC
FDB7    AF       HWRITE: XRA     A              ;SET READOP=WRITE
FDB8    32 F3D1          STA     READOP
FDBB    79               MOV     A,C            ;SAVE WRITE TYPE
FDBC    32 F3C4          STA     WRTYPE
FDBF    FE02             CPI     WRUAL          ;UNALOC WRITE?
FDC1    2018             JRNZ    CHKUNA         ;NOT UNALOC WRT
FDC3    2A F3C9          LHLD    SEKDSK         ;UNADSK=SEKDSK
FDC6    22 F3C6          SHLD    UNADSK         ;UNATRK=SEKTRK
FDC9    3A F3C2          LDA     BLKCNT         ;RECORDS/CPM BLOCK
FDCC    32 F3C5          STA     UNACNT
FDCF    3A F3CB          LDA     SEKSEC         ;UNASEC=SEKSEC
FDD2    32 F3C8          STA     UNASEC
FDD5    2A F3D8          LHLD    CTBLP          ;SAVE TBL ADR
FDD8    22 F3DA          SHLD    UTBLP
FDDB    3A F3C5  CHKUNA: LDA     UNACNT         ;ANY UNALOC REMAIN?
FDDE    B7               ORA     A
FDDF    2834             JRZ     ALLOC          ;NONE REMAIN
FDE1    3D               DCR     A              ;UNACNT-1
FDE2    32 F3C5          STA     UNACNT
FDE5    2A F3C6          LHLD    UNADSK         ;SEK = UNA?
FDE8    3A F3CB          LDA     SEKSEC
FDEB    4F               MOV     C,A
FDEC    3A F3C8          LDA     UNASEC
FDEF    CD FECB          CALL    CMP3
FDF2    2021             JRNZ    ALLOC          ;NOT EQUAL
FDF4    2A F3DA          LHLD    UTBLP          ;GET TBL ADR
FDF7    4E               MOV     C,M            ;CPM SEC/TRK
FDF8    0600             MVI     B,0
FDFA    11 0014          LXI     D,20           ;SEC TRANSL TBL
FDFD    19               DAD     D
FDFE    3A F3C8          LDA     UNASEC         ;SEARCH FOR UNASEC
FE01    EDB1             CCIR                   ;END OF TRK?
FE03    EA FE0E          JO      NOOVF          ;NOT END
FE06    21 F3C7          LXI     H,UNATRK       ;UNATRK + 1
FE09    34               INR     M
FE0A    3E01             MVI     A,1            ;UNASEC = 1
FE0C    1801             JMPR    OVF
FE0E    7E       NOOVF:  MOV     A,M            ;GET NEXT SECTOR
FE0F    32 F3C8  OVF:    STA     UNASEC         ;STORE NEXT UNASEC
FE12    AF               XRA     A              ;RESET RSFLAG
FE13    1805             JMPR    ALLOC2
FE15    AF       ALLOC:  XRA     A              ;SET UNACNT=0
```

```
FE16  32 F3C5          STA     UNACNT
FE19  3C               INR     A               ;SET RSFLAG
FE1A  32 F3D0  ALLOC2: STA     RSFLAG          ;NEED PREREAD
FE1D  21 EA33          LXI     H,DBUF          ;SET DMA ADR
FE20  22 F3BD          SHLD    DMAAD
FE23  DDE5     RWOPER: PUSH    X
FE25  3A F3C9          LDA     SEKDSK          ;LOG ON SEKDSK
FE28  CD FCD9          CALL    LOGDS2
FE2B  3A F3CB          LDA     SEKSEC          ;COMPUTE PHY SEC ADR
FE2E  4F               MOV     C,A
FE2F  E680             ANI     80H             ;GET SIDE SELECT
FE31  32 F3CD          STA     SIDSEL          ;SAVE IT
FE34  79               MOV     A,C             ;MASK OFF SIDE SEL
FE35  E67F             ANI     7FH
FE37  4F               MOV     C,A
FE38  3A F3C1          LDA     HSTBLK
FE3B  0D               DCR     C
FE3C  1F       PSEC1:  RAR
FE3D  3804             JRC     SECDN
FE3F  CB39             SRLR    C
FE41  18F9             JMPR    PSEC1
FE43  0C       SECDN:  INR     C
FE44  3A F3CD          LDA     SIDSEL          ;GET SIDE SELECT
FE47  B1               ORA     C               ;OR IT WITH SECTOR
FE48  32 F3CC          STA     SEKHST          ;SAVE IT
FE4B  21 F3CF          LXI     H,HSTACT        ;TEST & SET HSTACT FLG
FE4E  7E               MOV     A,M
FE4F  3601             MVI     M,1             ;SET FLAG
FE51  B7               ORA     A               ;TEST FLAG
FE52  2816             JRZ     FILHST          ;NOT SET
FE54  2A F3B7          LHLD    DISKNO          ;SEEK=HOST?
FE57  3A F3CC          LDA     SEKHST
FE5A  4F               MOV     C,A
FE5B  3A F3B9          LDA     SECTOR
FE5E  CD FECB          CALL    CMP3
FE61  2825             JRZ     MATCH           ;SAME
FE63  3A F3CE          LDA     HSTWRT          ;HOST WRITTEN?
FE66  B7               ORA     A
FE67  C4 FED9          CNZ     DWRITE          ;WRITE HOST BUF
FE6A  2A F3C9  FILHST: LHLD    SEKDSK          ;GET SET TO FILL BUF
FE6D  22 F3B7          SHLD    DISKNO
FE70  3A F3CC          LDA     SEKHST          ;PHYSICAL SEC ADR
FE73  DDCB1256         BIT     2,18(X)         ;1ST SEC = 0?
FE77  2801             JRZ     FLHST2          ;1ST SEC = 1
FE79  3D               DCR     A               ;1ST SEC = 0
FE7A  32 F3B9  FLHST2: STA     SECTOR
FE7D  3A F3D0          LDA     RSFLAG          ;RSFLAG SET?
FE80  B7               ORA     A
FE81  C4 FED5          CNZ     DREAD           ;YES DO READ
FE84  AF               XRA     A
FE85  32 F3CE          STA     HSTWRT          ;NO PENDING WRT
FE88  3A F3CB  MATCH:  LDA     SEKSEC          ;MASK BUF SEL BITS
FE8B  3D               DCR     A
FE8C  6F               MOV     L,A
FE8D  3A F3C3          LDA     SECMSK
FE90  A5               ANA     L
FE91  67               MOV     H,A             ;BUF SEL BITS X 12
FE92  2E00             MVI     L,0
FE94  CB3C             SRLR    H
FE96  CB1D             RARR    L
FE98  11 EA33          LXI     D,DBUF          ;ADD BUFFER INDEX
FE9B  19               DAD     D
FE9C  ED5B F3D6        LDED    SEKDMA          ;GET SEEK DMA ADR
FEA0  01 0080          LXI     B,128           ;SIZE OF MOVE
FEA3  3A F3D1          LDA     READOP          ;READOP=1?
FEA6  B7               ORA     A
FEA7  2006             JRNZ    RWMOVE          ;YES MOVE READ DAT
FEA9  3E01             MVI     A,1
FEAB  32 F3CE          STA     HSTWRT          ;SET HOST WRITTEN
FEAE  EB               XCHG                    ;MOVE WRITE DATA
FEAF  EDB0     RWMOVE: LDIR                    ;MOVE THE DATA
FEB1  3A F3C4          LDA     WRTYPE          ;WRITE TO DIRECTOR
FEB4  FE01             CPI     WRDIR
FEB6  3A F3D3          LDA     ERFLAG          ;GET ERROR FLAG
FEB9  200D             JRNZ    RWD             ;NOT DIR WRT
FEBB  B7               ORA     A               ;ERRORS?
FEBC  200A             JRNZ    RWD             ;DONE IF ERRORS
FEBE  AF               XRA     A
FEBF  32 F3CE          STA     HSTWRT          ;RESET HSTWRT FLAG
FEC2  CD FED9          CALL    DWRITE          ;WRITE TO DIRECTORY
FEC5  3A F3D3          LDA     ERFLAG          ;GET ERROR FLAG
FEC8  DDE1     RWD:    POP     X
FECA  C9               RET                     ;DONE
FECB  ED5B F3C9 CMP3:  LDED    SEKDSK          ;GET SEKDSK & SEKTR
FECF  B7               ORA     A
FED0  ED52             DSBC    D               ;COMPARE HL & DE
FED2  C0               RNZ                     ;RETURN NOT EQUAL
FED3  B9               CMP     C               ;COMPARE A & C
FED4  C9               RET                     ;DONE
                       ;
                       ;DISK READ AND WRITE ROUTINES
                       ;
FED5  3E01     DREAD:  MVI     A,1             ;SET READ FLAG
FED7  1801             JMPR    STRFLG
FED9  AF       DWRITE: XRA     A               ;SET WRITE FLAG
FEDA  32 F3BC  STRFLG: STA     RWFLG           ;SAVE IT FOR LATER USE
FEDD  060A     DORDWT: MVI     B,10            ;NUMBER OF RETRIES
FEDF  C5       AGN:    PUSH    B               ;SAVE BC
FEE0  CD FF42          CALL    SEEK            ;SEEK THE TRACK
FEE3  CC FEF5          CZ      RDWR            ;NO ERROR
FEE6  32 F3D3          STA     ERFLAG          ;STORE ERROR FLAG
FEE9  C1       READ3:  POP     B               ;GET ERROR RETRY CO
FEEA  C8               RZ                      ;RETURN IF NO ERROR
FEEB  3EFF             MVI     A,0FFH          ;CAUSE IDRD ON RETR
FEED  32 F3D2          STA     LUNIT
FEF0  10ED             DJNZ    AGN             ;DO A RETRY
FEF2  C3 FC78          JMP     DERR1           ;DISP PARAMETERS
                       ;
FEF5  5F       RDWR:   MOV     E,A             ;SAVE COMMAND
FEF6  3A F3BC          LDA     RWFLG
```

```
FEF9  B7                ORA     A
FEFA  7B                MOV     A,E     ;REGET THE COMMAND
FEFB  2813              JRZ     WRDAT   ;WRITE IF ZERO
FEFD  D39C      RDAT:   OUT     DCMMD   ;DISK COMMAND PORT
FEFF  32 F3BB           STA     CMND
FF02  01 809F   READ1:  LXI     B,(128*256)+DDATA
FF05  EDB2      READ2:  INIR
FF07  15                DCR     D       ;LOOP CONTROL
FF08  20F8              JRNZ    READ1
FF0A  CD FF31           CALL    EOJ
FF0D  E69C              ANI     9CH     ;ISOLATE READ ERROR BITS
FF0F  C9                RET
                ;
FF10  F620      WRDAT:  ORI     20H     ;ADD WRITE COMMAND
FF12  D39C              OUT     DCMMD   ;DISK COMMAND PORT
FF14  33 F3BB           STA     CMND
FF17  01 809F   WRT1:   LXI     B,(128*256)+DDATA
FF1A  EDB3              OUTIR
FF1C  15                DCR     D       ;LOOP CONTROL
FF1D  20F8              JRNZ    WRT1
FF1F  1810.             JMPR    EOJ
FF21  0608      EOJB:   MVI     B,8     ;BASIS OF RESTORE COMMAND
FF23  DB9B      EOJA:   IN      DFLAG   ;STEP RATE BITS
FF25  E603              ANI     3
FF27  B0                ORA     B       ;ADD ON THE COMMAND
FF28  32 F3BB           STA     CMND
FF2B  D39C              OUT     DCMMD   ;DO THE COMMAND
FF2D  0680              MVI     B,128   ;DELAY
FF2F  10FE      SKDLY:  DJNZ    SKDLY
FF31  DB9C      EOJ:    IN      DSTAT   ;TEST FOR INTRQ
FF33  E601              ANI     1
FF35  20FA              JRNZ    EOJ             ;WAIT FOR INTRQ
FF37  CD FFB7           CALL    DWAIT           ;DISABLE WAITS
FF3A  DB9C      EOJ1:   IN      DSTAT   ;GET THE DISK STATUS
FF3C  32 F3BA           STA     STATUS
FF3F  E6FC              ANI     0FCH
FF41  C9                RET
                ;
FF42  CD FCD6   SEEK:   CALL    LOGDSK          ;LOG ON DISKNO
FF45  C0                RNZ                     ;ERROR RETURN
FF46  CD FF8B           CALL    TST128          ;ADJUST SECTOR ADR
FF49  D39E              OUT     DSCTR           ;DISK SECTOR PORT
FF4B  DB9D      SEEK2:  IN      DTRCK           ;DISK TRACK PORT
FF4D  4F                MOV     C,A             ;SAVE IT
FF4E  3A F3BB           LDA     TRACK           ;GET DESIRED TRACK
FF51  B9                CMP     C
FF52  280A              JRZ     RDWRT           ;NO SEEK NEEDED
FF54  D39F              OUT     DDATA           ;SET THE TRACK
FF56  0618              MVI     B,18H           ;SEEK COMMAND
FF58  CD FF23           CALL    EOJA            ;DO THE SEEK
FF5B  E680              ANI     80H             ;ERROR MASK
FF5D  C0                RNZ                     ;SEEK ERROR
FF5E  CD FF96   RDWRT:  CALL    SETUP           ;GET READY
FF61  DB9B              IN      DFLAG
FF63  E610              ANI     10H             ;HEAD LOADED?
```

```
FF65  3E04              MVI     A,4
FF67  2801              JRZ     RDWRT1          ;JUMP IF NOT
FF69  AF                XRA     A               ;RESET HEAD LD FLAG
FF6A  C688      RDWRT1: ADI     88H             ;READ COMMAND
FF6C  4F                MOV     C,A             ;SAVE IT
FF6D  3A F3BB           LDA     TRACK           ;TEST FOR TRK 0
FF70  B7                ORA     A
FF71  1601              MVI     D,1
FF73  2808              JRZ     RDWRT2          ;TRK 0 = SD
FF75  DD6E0F            MOV     L,15(X)         ;GET SECTOR SIZE
FF78  DD6610            MOV     H,16(X)
FF7B  29                DAD     H
FF7C  54                MOV     D,H
FF7D  2A F3BD   RDWRT2: LHLD    DMAAD           ;GET DMA ADDRESS
FF80  3A F3B9   RDWRT3: LDA     SECTOR          ;TEST FOR SIDE 1
FF83  E680              ANI     80H
FF85  79                MOV     A,C             ;REGET COMMAND
FF86  C8                RZ                      ;SIDE 0
FF87  F602              ORI     2               ;SELECT SIDE 1
FF89  BF                CMP     A               ;CLEAR THE FLAGS
FF8A  C9                RET
FF8B  3A F3B9   TST128: LDA     SECTOR
FF8E  DDCB125E          BIT     3,18(X)         ;SD 1 < 1287
FF92  C8                RZ                      ;SD 1 > 128
FF93  E67F              ANI     7FH             ;SD 1 < 128
FF95  C9                RET
                ;
                ;
                ;SELECTS DRIVE
                ;
FF96  21 F3B9   SETUP:  LXI     H,SECTOR        ;POINT TO SECTOR
FF99  DB9B              IN      DFLAG
FF9B  E640              ANI     40H             ;TEST FOR TRK 0
FF9D  3A F3C0           LDA     HSELCD          ;GET SEL CODE
FFA0  2004              JRNZ    SETUP2          ;NOT TRK 0
FFA2  CB7E              BIT     7,M             ;TEST FOR SIDE 1
FFA4  280E              JRZ     SETUP3          ;TRK 0, SIDE 0 IS SD
FFA6  DDCB127E  SETUP2: BIT     7,18(X)         ;TEST FOR DD/SD
FFAA  2802              JRZ     SETUP4          ;THIS DSK IS SD
FFAC  CBB7              RES     6,A             ;SET DD
FFAE  CB7E      SETUP4: BIT     7,M             ;TEST FOR SIDE 1
FFB0  2802              JRZ     SETUP3          ;SIDE 0
FFB2  CBEF              SET     5,A             ;SELECT SIDE 1
FFB4  D39B      SETUP3: OUT     DCNTL
FFB6  C9                RET
                ;
                ;
                ;DISABLE WAIT STATES
                ;
FFB7  3A F3C0   DWAIT:  LDA     HSELCD          ;RESET AUTO-WAIT
FFBA  E67F              ANI     7FH
FFBC  D39B              OUT     DCNTL
FFBE  C9                RET
                ;
                ;
```

```
FFBF                    EPBASE:
                        ;DISK PARAMETER HEADER FOR DISK 00
FFBF    EE47    0000      WORD  TD0,0
FFC3    0000    0000      WORD  0,0
FFC7    F033    EE33      WORD  DIRBF,DPB0
FFCB    F2B3    F0B3      WORD  CHK00,ALL00
                        ;DISK PARAMETER HEADER FOR DISK 01
FFCF    EEC7    0000      WORD  TD1,0
FFD3    0000    0000      WORD  0,0
FFD7    F033    EEB3      WORD  DIRBF,DP1
FFDB    F2F3    F133      WORD  CHK01,ALL01
                        ;DISK PARAMETER HEADER FOR DISK 02
FFDF    EF47    0000      WORD  TD2,0
FFE3    0000    0000      WORD  0,0
FFE7    F033    EF33      WORD  DIRBF,DP2
FFEB    F333    F1B3      WORD  CHK02,ALL02
                        ;DISK PARAMETER HEADER FOR DISK 03
FFEF    EFC7    0000      WORD  TD3,0
FFF3    0000    0000      WORD  0,0
FFF7    F033    EFB3      WORD  DIRBF,DP3
FFFB    F373    F233      WORD  CHK03,ALL03
0040                    EPLGTH  =       .-EPBASE
                        ;
                        ;SCRATCH RAM AREA FOR BDOS USE
                        ;
EA33                    BEGDAT  =       BIOS+BCDL       ;BEGINNING OF DATA AREA
EA33                    .LOC    BEGDAT
EA33                    DBUF:   .BLKB   1024            ;DEBLOCK BUFFER
EE33                    DPB0:   .BLKB   20
EE47                    TD0:    .BLKB   108
EEB3                    DP1:    .BLKB   20
EEC7                    TD1:    .BLKB   108
EF33                    DP2:    .BLKB   20
EF47                    TD2:    .BLKB   108
EFB3                    DP3:    .BLKB   20
EFC7                    TD3:    .BLKB   108
F033                    DIRBF:  .BLKB   128     ;SCRATCH DIRECTORY AREA
F0B3                    ALL00:  .BLKB   128     ;ALLOCATION VECTOR 0
F133                    ALL01:  .BLKB   128     ;ALLOCATION VECTOR 1
F1B3                    ALL02:  .BLKB   128     ;ALLOCATION VECTOR 2
F233                    ALL03:  .BLKB   128     ;ALLOCATION VECTOR 3
F2B3                    CHK00:  .BLKB   64      ;CHECK VECTOR 0
F2F3                    CHK01:  .BLKB   64      ;CHECK VECTOR 1
F333                    CHK02:  .BLKB   64      ;CHECK VECTOR 2
F373                    CHK03:  .BLKB   64      ;CHECK VECTOR 3
F3B3                    DSKSEL: .BLKB   4       ;DISK ACCESS TABLE
F3B7                    DISKNO: .BLKB   1       ;DISK PARAMATERS
F3B8                    TRACK:  .BLKB   1
F3B9                    SECTOR: .BLKB   1
F3BA                    STATUS: .BLKB   1
F3BB                    CMND:   .BLKB   1
F3BC                    RWFLG:  .BLKB   1
F3BD                    DMAAD:  .BLKB   2
F3BF                    PARFLG: .BLKB   1       ;PAR ERROR FLAG
F3C0                    HSELCD: .BLKB   1       ;HOST SELECT CODE
F3C1                    HSTBLK: .BLKB   1       ;CPM SEC/PHY SEC
F3C2                    BLKCNT: .BLKB   1       ;REC/CPMBLK
F3C3                    SECMSK: .BLKB   1       ;SECTOR MASK
F3C4                    WRTYPE: .BLKB   1       ;WRITE TYPE
F3C5                    UNACNT: .BLKB   1       ;UNALOC REC COUNT
F3C6                    UNADSK: .BLKB   1       ;LAST UNALOC DISK
F3C7                    UNATRK: .BLKB   1       ;LAST UNALOC TRACK
F3C8                    UNASEC: .BLKB   1       ;LAST UNALOC SECTOR
F3C9                    SEKDSK: .BLKB   1       ;SEEK DISK
F3CA                    SEKTRK: .BLKB   1       ;SEEK TRACK
F3CB                    SEKSEC: .BLKB   1       ;SEEK SECTOR
F3CC                    SEKHST: .BLKB   1       ;PHYSICAL SEC ADR
F3CD                    SIDSEL: .BLKB   1       ;SIDE SELECT STORAGE
F3CE                    HSTWRT: .BLKB   1       ;HOST WRITTEN FLAG
F3CF                    HSTACT: .BLKB   1       ;HOST ACTIVE FLAG
F3D0                    RSFLAG: .BLKB   1       ;READ SECTOR FLAG
F3D1                    READOP: .BLKB   1       ;1=READ, 0=WRITE
F3D2                    LUNIT:  .BLKB   1       ;LAST SELECTED DRIVE
F3D3                    ERFLAG: .BLKB   1       ;ERROR FLAG
F3D4                    RFLAG:  .BLKB   1       ;CPM READ FLAG
F3D5                    BTSEC:  .BLKB   1       ;NO OF BOOT SECS
F3D6                    SEKDMA: .BLKB   2       ;SEEK DMA ADDRESS
F3D8                    CTBLP:  .BLKB   2       ;CURRENT TBL ADR
F3DA                    UTBLP:  .BLKB   2       ;UNALOC TBL ADR
F3DC                    HLSTR:  .BLKB   2
F3DE                    DPBASE: .BLKB   EPLGTH  ;CPM DISK PARAMETER HEA
F41E                    IDSV:   .BLKB   6
                        ;
09F1                    DATSIZ  =       .-BEGDAT        ;SIZE OF SCRATC
                        ;END OF PROGRAM
                        .END
```