.MAIN. -

Format

```
0002                    version ==      2
0000                    revision==      0
                                .pabs
                                .phex

                        ;                       Update History

                        ;   Version 2.0 - (7/29)  Escape can be used for
                        ;                       program restart. See CONIN.
                        ;                       Head step delay lengthened to
                        ;                       15ms.  Head load delay of 50ms
                        ;                       inserted in ONMOTOR routine.
                        ;                       I/O errors do not terminate
                        ;                       the program anymore.
                        ;                       Conditional assembly for
                        ;                       testing (TFORMAT) installed.
                        ;                       Floppy buffer is printed to
                        ;                       console if IO error occurs
                        ;                       under TFORMAT.

                        ;----------
                        ; Data and address definitions:
                        ;
000D                    cr      =       0Dh     ; <return>
000A                    lf      =       0Ah     ; <linefeed>
0047                    userno  =       47h     ; Network user number
0004                    drivno  =       04h     ; Currently selected drive num
0016                    onprecmp=       22      ; turn on pre-compensation
003B                    maxprec =       59      ; turn off S/L bit
0900                    stepset =       900h    ; head step settle time = 15ms
1E00                    loadset =       1E00h   ; head load settle time = 50ms
0000                    RxRDY   =       0       ; receiver ready bit
0002                    TxRDY   =       2       ; transmitter ready bit
0001                    WBOOT   =       1       ; for bios calls
                        ;
                        ; Port definitions:
                        ;
0038                    DMA     =       38h
002A                    SIO1AC  =       2Ah     ; SIO-1 channel A, control
0028                    SIO1AD  =       28h     ; SIO-1 channel A, data -port 0
0008                    PIOAD   =       08h     ; PIO channel A, data
0009                    PIOBD   =       09h     ; PIO channel B, data
0018                    FLCP    =       18h     ; Floppy DMA channel
0010                    FLCPSR  =       10h     ; Floppy status register
0011                    FLCPDR  =       11h     ; Floppy data register
0003                    SETMAP  =       03h     ; Set memory map register
0003                    STOPFIOP=       03h     ; Stop floppy controller

0001                    TFMTflg =\ "
                                  Enter 0 for FORMAT  (interleaving sectors)
                                        1 for TFORMAT (linear sectors)      "

                        ;---------
                        ; Entry Points for ZDT
                        ;
```

```
0100                            .loc    100h
0100    C3 0106                 jmp     TEST
0103    C3 0142                 jmp     START

0106    3A 0047         TEST:   lda     userno  ; Make sure we're not on
0109    FE00                    cpi     00h     ; a Hinet master.
010B    2035                    jrnz    START   ; We are not. Proceed
010D    21 0116                 lxi     H,..NOGOmsg
0110    CD 0441                 call    prtmsg  ; We are. Tell user.
0113    C3 0000                 jmp     0       ; and boot.
0116                    ..NOGOmsg:
0116    0D0A464F524D            .ascii  [cr][lf]'FORMAT cannot be used on a '
0133    48694E657420            .asciz  'HiNet master.'

0142    31 0620         START:  lxi     sp,stack; Set up our stack
0145    2A 0001                 lhld    WBOOT   ; use interrupt vector in bios
0148    2E00                    mvi     L,0     ; where cold boot was
014A    36C7                    mvi     M,DMAdone&0FFh ; low
014C    23                      inx     H
014D    3603                    mvi     M,(DMAdone>8) & 0FFh ; then high
014F    CD 01D5                 call    FORMAT  ; format the disk(s)
0152    C3 0495                 jmp     EXIT    ; and warm boot.
```

```
                        ;----------
                        ; Home selected disk drive to track 0
0155                    HOME:
0155    3E00                    mvi     A,0     ; current track now zero
0157    32 04EC                 sta     curTRK
015A    21 04F4                 lxi     H,homeFLOP+1 ; point to drive number
015D    1845                    jmpr    seek    ; share code with SETTRK
                        ;----------
                        ; Select disk drive
015F                    SELDSK:
015F    21 017F                 lxi     H,DEVqst ; ask which disk
0162    CD 0441                 call    PRTMSG
0165    CD 047C                 call    CONIN   ; get the answer
0168    FE03                    cpi     03h     ; check for cntrl-C
016A    CA 0495                 jz      EXIT    ; Prepare to EXIT if CTRL C
016D    4F                      mov     C,A
016E    CD 048D                 call    CONOUT  ; echo it
0171    D630                    sui     '0'
0173    FA 015F                 jm      SELDSK
0176    FE08                    cpi     8
0178    F2 015F                 jp      SELDSK
017B    32 04EB                 sta     curDSK
017E    C9                      ret
017F    0D0A0A456E74 DEVqst: .asciz  [cr][lf][lf]'Enter Disk Number (0-7): '
                        ;----------
                        ; Set track
019C                    SETTRK:
019C    3A 04EC                 lda     curTRK  ; get current track
019F    21 04F8                 lxi     H,seekFLOP+2 ; point to track number
01A2    77                      mov     M,A     ; store into command
01A3    2B                      dcx     H
                        ;
                        ; Seek a track (used by SETTRK and HOME)
01A4                    seek:
01A4    3A 04EB                 lda     curDSK  ; get current disk
01A7    77                      mov     M,A     ; store into command
01A8    2B                      dcx     H
01A9    CD 04CF                 call    onMOTOR ; turn on the drive motor
01AC    54              reseek: mov     D,H     ; save ptr to seek command
01AD    5D                      mov     E,L
01AE    CD 04A7                 call    COMMAND ; seek
01B1    21 04FA         sense:  lxi     H,IsenseFLOP
01B4    CD 04A7                 call    COMMAND ; sense interrupt status
01B7    CD 04B7                 call    RESULT
01BA    62                      mov     H,D     ; restore ptr to seek command
01BB    6B                      mov     L,E
01BC    3A 04FE                 lda     resbuf  ; get first result byte
01BF    CB5F                    bit     3,A     ; check ready bit
01C1    20E9                    jrnz    reseek  ; jump if not ready
01C3    CB6F                    bit     5,A     ; mask out top 2 bits
01C5    28EA                    jrz     sense   ; loop until seek completed
01C7    E6D0                    ani     0D0h
01C9    C4 03E3                 cnz     IOERR
01CC    01 0900                 lxi     B,stepset; delay 10 ms for step settle
                        ;---------
```

```
                        ; Delay loop
01CF    0B              DELAY:  dcx     B
01D0    78                      mov     A,B
01D1    B1                      ora     C
01D2    20FB                    jrnz    DELAY
01D4    C9                      ret
```

```
                        ;----------
                        ; Format the entire disk single or double density
                        ;
01D5                    FORMAT:
01D5    21 0220                 lxi     H,FCRMstr ; print start-up message
01D8    CD 0441                 call    PRTMSG
01DB    CD 015F         ..form: call    SELDSK  ; ask which disk
01DE    C8                      rz              ; return if cntrl-C
01DF    CD 0286                 call    ASKDEN  ; ask what density
01E2    CD 02D3                 call    WAITCR  ; wait for return
01E5    CD 0155                 call    HOME    ; recalibrate the disk drive
01E8    1600                    mvi     D,0     ; track 0 is single density
01EA    CD 0302                 call    TABLIN  ; track 0 is linear
01ED    CD 0371                 call    FORM1TK ; format the first track

01F0    CD 019C         ..1:    call    SETTRK  ; seek to next track
01F3    3A 04F2                 lda     curDEN  ; rest of disk is specified
01F6    57                      mov     D,A     ; density
01F7    FE00                    cpi     0       ; single density is linear
01F9    2810                    jrz     ..2
01FB    3A 04EC                 lda     curTRK  ; so is track 1
01FE    FE01                    cpi     1
0200    2809                    jrz     ..2
0202    D5                      push    D       ; save density
0203    3E04                    mvi     A,4     ; set stagger for DD
                                .ife    TFMTflg,    [
                                call    TABSTAG ; stagger table for FORMAT
                                                                        ]
                                .ife    TFMTflg-1, [
0205    CD 0302                 call    TABLIN  ; dont stagger for TFORMAT
                                                                        ]
0208    D1                      pop     D
0209    1803                    jmpr    ..3
020B    CD 0302         ..2:    call    TABLIN  ; make table linear
020E    CD 0371         ..3:    call    FORM1TK ; format next track
0211    FE4E                    cpi     78      ; all tracks formatted?
0213    20DB                    jrnz    ..1     ; if not, loop
0215    21 0271                 lxi     H,FORMdon ; say 'done'
0218    CD 0441                 call    PRTMSG
021B    97                      sub     A       ; A = 0
021C    D309                    out     PIOBD   ; unload the head
021E    18BB                    jmpr    ..form

0220    0D0A466C6F70 FORMstr:.ascii  [cr][lf]'Floppy Disk'
022D    2054455354              .ife    TFMTflg-1, [.ascii " TEST " ]
0232    20466F726D61            .ascii  ' Format Program '
0242    322E                    .byte   version+'0','.'
0244    3030                    .byte   revision/10+'0',revision@10+'0'
0246    0D0A55736520 infomsg:.ascii  [cr][lf]'Use ESC to restart, '
025C    436F6E74726F            .asciz  'Control-C to abort. '

0271    0D0A464F524D FORMdon:.asciz  [cr][lf]'FORMAT COMPLETED'[cr][lf]

                        ;----------
                        ; Ask what density
```

```
02E6                    ASKDEN:
0286    21 02B6                 lxi     H,DENqst ; set up message
0289    CD 0441                 call    PRTMSG
028C    CD 047C                 call    CONIN   ; get answer
028F    4F                      mov     C,A
0290    CD 048D                 call    CONOUT  ; echo it
0293    FE73                    cpi     's'     ; check for single
0295    2804                    jrz     ..1
0297    FE53                    cpi     'S'     ; check for single
0299    2006                    jrnz    ..2
029B    3E00            ..1:    mvi     A,0     ; set single density
029D    32 04F2                 sta     curDEN
02A0    C9                      ret
02A1    FE64            ..2:    cpi     'd'     ; check for double
02A3    2E0B                    jrz     ..3
02A5    FE44                    cpi     'D'
02A7    2807                    jrz     ..3
02A9    FE03                    cpi     3       ; 3 is control-C
02AB    CA 0495                 jz      EXIT    ; Prepare to exit if CTRL-C
02AE    18D6                    jmpr    ASKDEN  ; Bad entry.
02B0    3E01            ..3:    mvi     A,1     ; set double density
02B2    32 04F2                 sta     curDEN
02B5    C9                      ret
02B6    0D0A53696E67 DENqst: .asciz  [cr][lf]'Single or Double density? '
                        ;----------
                        ; Wait for CR
02D3                    WAITCR:
02D3    21 02EA                 lxi     H,WCRqst ; print a message
02D6    CD 0441                 call    PRTMSG
02D9    CD 047C         ..1:    call    CONIN   ; get an answer
02DC    FE03                    cpi     3       ; 3 is control-C
02DE    CA 0495                 jz      EXIT    ; Prepare to exit if CTRL-C.
02E1    FE0D                    cpi     cr      ; is it cr?
02E3    20F4                    jrnz    ..1     ; if not loop
02E5    4F                      mov     C,A
02E6    CD 048D                 call    CONCUT  ; echo itV
02E9    C9                      ret
02EA    0D0A54797065 WCRqst: .asciz  [cr][lf]'Type return to start.'
                        ;----------
                        ; Format curTRK and increment it.
                        ;
                        ; Build Sector Table (linear)
0302                    TABLIN:
0302    21 3000                 lxi     H,bufFLOP ; build sector table in buf
0305    3A 04EC                 lda     curTRK  ; C should have curTRK
0308    4F                      mov     C,A
0309    3E01                    mvi     A,1     ; A has sector number

030B    71              ..1:    mov     M,C     ; fill in TRACK
030C    23                      inx     H
030D    F5                      push    PSW
030E    3A 04EB                 lda     curDSK
0311    0F                      rrc
0312    0F                      rrc
0313    E601                    ani     1       ; compute head (ie, side)
```

```
0315    77              mov     M,A     ; fill in HEAD (0 or 1)
0316    F1              pop     PSW
0317    23              inx     H
0318    77              mov     M,A     ; fill in SECTOR (counting)
0319    23              inx     H
031A    72              mov     M,D     ; code for BYTES/SECTOR
031B    23              inx     H
031C    3C              inr     A       ; set to next sector
031D    FE1B            cpi     27      ; end of list?
031F    20EA            jrnz    ..1     ; no, then loop
0321    C9              ret

                ;
                ; Build staggered sector table
0322                    TABSTAG:
0322    D5              push    D       ; save density
0323    21 3000         lxi     H,bufFLOP ; zero out the output buffer

0326    3600            mvi     M,0
0328    11 3001         lxi     D,bufFLOP+1
032B    01 00FF         lxi     B,255
032E    EDB0            ldir

0330    21 3000         lxi     H,bufFLOP ; build the sector table
0333    47              mov     B,A     ; B has the stagger count
0334    3A 04EC         lda     curTRK  ; C has the track number
0337    4F              mov     C,A
0338    D1              pop     D       ; restore density
0339    3E01            mvi     A,1     ; first sector is 1

033B    71              ..1:    mov     M,C     ; fill in TRACK
033C    23              inx     H
033D    F5              push    PSW
033E    3A 04EB         lda     curDSK
0341    0F              rrc
0342    0F              rrc
0343    E601            ani     1
0345    77              mov     M,A     ; fill in HEAD (0 or 1)
0346    F1              pop     PSW
0347    23              inx     H
0348    77              mov     M,A     ; fill in SECTOR (counting)
0349    23              inx     H
034A    72              mov     M,D     ; fill in BYTES/SECTOR
034B    23              inx     H

034C    FE1A            cpi     26      ; last sector?
034E    C8              rz              ; return if yes
034F    3C              inr     A       ; set next sector
0350    C5              push    B       ; save stagger count
0351    23              ..2:    inx     H       ; increment HL by 4 for each
0352    23              inx     H       ; 1 sector to stagger
0353    23              inx     H
0354    23              inx     H
0355    10FA            djnz    ..2     ; repeat for stagger count
0357    C1              pop     B       ; restore stagger counter
```

```
0358    F5              push    PSW     ; save A
0359    7D      ..3:    mov     A,L     ; check for HL overflow
035A    FE68            cpi     26*4
035C    FA 0362         jm      ..4
035F    DE68            sbi     26*4    ; if overflow subtract extra
0361    6F              mov     L,A     ; from HL
0362    23      ..4:    inx     H       ; check if this sector already
0363    23              inx     H       ; filled
0364    7E              mov     A,M
0365    B7              ora     A
0366    2804            jrz     ..5
0368    23              inx     H       ; if so skip it, and try next
0369    23              inx     H       ; next sector
036A    18ED            jmpr    ..3
036C    2B      ..5:    dcx     H       ; restore HL
036D    2B              dcx     H
036E    F1              pop     PSW     ; restore A
036F    18CA            jmpr    ..1     ; fill next sector

0371            FORM1TK:
                ;
                ; Set up the DMA chip
0371    3E00            mvi     A,0     ; multiplex Floppy to DMA
0373    D308            out     PIOAD
0375    21 050C         lxi     H,DMAfmt
0378    01 1438         lxi     B,DMAfm$<8+DMA
037B    EDB3            outir           ; program the chip
                ;
                ; Set up the floppy format command
037D    3A 04EB         lda     curDSK  ; set up current disk
0380    32 0506         sta     FORMdsk
0383    7A              mov     A,D     ; set up floppy density
0384    32 0507         sta     FORMmod
0387    FE00            cpi     0       ; check for density
0389    200C            jrnz    ..2     ; and fill in special fields
038B    3E0D            mvi     A,0Dh   ; set up for single density
038D    32 0505         sta     FORMcom ; format command
0390    3E1B            mvi     A,1Bh   ; set gap size
0392    32 0509         sta     FORMgap
0395    180A            jmpr    ..3
0397    3E4D    ..2:    mvi     A,4Dh   ; set up for double density
0399    32 0505         sta     FORMcom ; format command
039C    3E36            mvi     A,36h   ; set gap size
039E    32 0509         sta     FORMgap
03A1            ..3:
                ;
                ; Execute the floppy command
03A1    21 0505         lxi     H,FORMcom
03A4    CD 03AF         call    EXflop
                ;
                ; Increment to next track
03A7    3A 04EC         lda     curTRK
03AA    3C              inr     A
03AB    32 04EC         sta     curTRK
```

```
03AE    C9              ret             ; return
                ;
                ; Execute the floppy command
03AF    FB      EXflop: ei              ; turn on interupts
03B0    CD 04A7         call    COMMAND
03B3    CD 04B7         call    RESULT
03B6    3A 04FE         lda     resbuf  ; get first result byte
03B9    E6C0            ani     0C0h    ; mask out top 2 bits
03BB    C4 03E3         cnz     ICERR   ; jump if abnormal termination
03BE    3A 04FF         lda     resbuf+1 ; get second result byte
03C1    E633            ani     33h     ; check for CRC error
03C3    C4 03E3         cnz     IOERR
03C6    C9              ret
                ;
                ; Handle the floppy-done interrupt from the DMA chip
03C7    F5      DMAdone:push    PSW
03C8    DB03            in      STOPFLOP; reset the floppy chip
03CA    3EC3            mvi     A,0C3h  ; reset the DMA chip
03CC    D338            out     DMA
03CE    F1              pop     PSW
03CF    FB              ei
03D0    ED4D            reti
                ;----------
                ; Handle an I/O error
                ;
                ; Print a message of the form 'I/O Error '.  If we
                ; are assembling  under the  Tformat option, then
                ; print the address from which the call here was
                ; made, and print the floppy result buffer.
                ; Next, space down and await a control C or ESC.
03D2    0D0A2A2A2A20 errmsg: .asciz  [cr][lf]'*** I/O Error '
03E3            IOERR:
03E3    21 03D2         lxi     H,errmsg
03E6    CD 0441         call    PRTMSG          ; Print error message
03E9    E1              pop     H
03EA    22 2FFE         shld    bufFLOP-2
03ED    22 04FC         shld    erradr          ; save our error addr
                ;Print out error details if this is TFORMAT
                .ife    TFMTflg-1, [
03F0    21 0424         lxi     H,..frommsg
03F3    CD 0441         call    prtmsg
03F6    21 04FD         lxi     H,erradr+1
03F9    3E01            mvi     A,1
03FB    CD 044D         call    PRTBYTS
03FE    21 04FC         lxi     H,erradr
0401    3E01            mvi     A,1             ; print addr where
0403    CD 044D         call    PRTBYTS         ; error occured
0406    21 042F         lxi     H,..resmsg
0409    CD 0441         call    prtmsg
040C    21 04FE         lxi     H,resbuf
040F    3E07            mvi     A,7
0411    CD 044D         call    PRTBYTS         ; print result buf
                                                ]
                ;Tell user what his options are now and wait for him.
0414    21 0246         lxi     H,infomsg
```

```
0417    CD 0441             call    prtmsg          ; print CTRL-C,ESC info
041A    CD 047C     ..wait: call    CONIN
041D    FE33                cpi     '3'     ;loop until ctrl-c
041F    20F9                jrnz    ..wait  ;(or ESCAPE from conin)
0421    C3 0495             jmp     EXIT    ;exit if ctrl-c

0424                ..frommsg:
0424    66726F6D2061        .asciz  'from addr '
042F                ..resmsg:
042F    2E2052657375        .asciz  '. Result buffer: '

2FFE                locerr  =       bufFIOP-2
```

```
                        ;----------
                        ; Utility routines
                        ;
                        ; PRTMSG  - print a message
                        ; PRTBYTS - print specified number of bytes
                        ; COMMAND - send a command to the floppy controller
                        ; RESULT  - get a result from the floppy controller
                        ; WAITDR  - wait for floppy data register
                        ; MOTORon - turn on the drive motor
                        ;----------
                        ; Print a message
                        ;  Regs in:   HL = address of message (ended by null)
                        ;  Regs out:  none
                        ;  Destroyed: A, HL
0441                    PRTMSG:
0441    7E                      mov     A,M     ; get a byte
0442    B7                      ora     A       ; if null, then return
0443    C8                      rz
0444    4F                      mov     C,A     ; put it where CONOUT wants it
0445    E5                      push    H       ; save HL
0446    CD 048D                 call    CONOUT  ; output the byte
0449    E1                      pop     H       ; restore HL
044A    23                      inx     H       ; point to next byte
044B    18F4                    jmpr    PRTMSG
                        ;----------
                        ; Print a specified number of bytes
                        ;  Regs  in:    A = number of bytes to  print
                        ;               HL= addr of 1st byte
                        ;  Regs out:    none
                        ;  Destoyed:    any/all
044D    FE01            PRTBYTS:cpi     1
044F    D8                      rc              ; nothing to print
0450    280C                    jrz     ..prtbyt; print just 1 char

0452    F5              ..loop: push    PSW     ; save number of chrs
0453    E5                      push    H
0454    CD 045E                 call    ..prtbyt; print 1 byte to console
0457    E1                      pop     H
0458    F1                      pop     PSW     ; restore number of chrs
0459    23                      inx     H       ; get to next byte
045A    3D                      dcr     A       ; Reduce our count
045B    C8                      rz              ; and return when finished.
045C    18F4                    jmpr    ..loop

                        ; Regs  in:     HL=addr of byte to be printed
045E                    ..prtbyt:
045E    7E                      mov     A,M
045F    F5                      push    PSW     ; save the chr
0460    07                      rlc
0461    07                      rlc
0462    07                      rlc
0463    07                      rlc
0464    CD 046C                 call    ..prtnbl
0467    F1                      pop     PSW
0468    CD 046C                 call    ..prtnbl
```

```
046B    C9              ret
046C                    ..prtnbl:
046C    E60F            ani     0Fh
046E    C630            adi     '0'
0470    FE3A            cpi     '9'+1
0472    4F              mov     C,A     ; chr goes out in C
0473    DA 048D         jc      CONOUT
0476    C607            adi     'A'-('9'+1)
0478    4F              mov     C,A     ; chr goes out in C
0479    C3 048D         jmp     CONOUT

                        ;----------
                        ; Console input
047C    CD 0485         CONIN:  call    ..1
047F    FE1B            cpi     1Bh     ; Is entry ESCAPE?
0481    C0              rnz             ; No. Return the chr.
0482    C3 01D5         jmp     FORMAT  ; Yes. Restart program.

0485    2A 0001         ..1:    lhld    WBOOT   ; call the bios
0488    11 0006         lxi     D,06h
048B    19              dad     D
048C    E9              pchl
                        ;----------
                        ; Console output
048D                    CONOUT:
048D    2A 0001         lhld    WBCOT   ; call the bios
0490    11 0009         lxi     D,09h
0493    19              dad     D
0494    E9              pchl
                        ;----------
                        ; Select original drive then exit
0495                    EXIT:
0495    CD 049B         call    ..seldrv; Select orig driv
0498    C3 0000         jmp     0       ; and warm boot

049B                    ..seldrv:
049B    3A 0004         lda     drivno  ; Get orig driv num
049E    4F              mov     C,A     ; in C.
049F    2A 0001         lhld    WBCCT   ; Send it to BIOS
04A2    11 001B         lxi     D,1Bh   ;
04A5    19              dad     D
04A6    E9              pchl

                        ;----------
                        ; Send a command to the floppy controller
                        ;  Regs in:   HL = address of command string
                        ;  Regs out:  none
                        ;  Destroyed: A, HL
04A7                    COMMAND:
04A7    7E              mov     A,M
04A8    FEFF            cpi     endcom
04AA    C8              rz              ; return if end-of-command
04AB    CD 04C8         call    WAITDR
04AE    DC 03E3         cc      IOERR   ; direction is wrong
04B1    7E              mov     A,M
```

```
04B2    D311            out     FLOPDR  ; send a byte to controller
04B4    23              inx     H       ; point to next byte
04B5    18F0            jmpr    COMMAND
                ;----------
                ; Collect a result from the floppy controller
                ;  Regs in:   none
                ;  Regs out:  B = length of result string
                ;  Destroyed: A, HL
04B7            RESULT:
04B7    21 04FE         lxi     H,resbuf
04BA    CD 04C8 readDR: call    WAITDR
04BD    D0              rnc
04BE    3EC3            mvi     A,0C3h
04C0    D338            out     DMA     ; reset DMA chip
04C2    DB11            in      FLCPDR  ; get a result byte
04C4    77              mov     M,A     ; store it away
04C5    23              inx     H
04C6    18F2            jmpr    readDR
                ;----------
                ; Wait until floppy data register is ready
                ;  Regs in:   none
                ;  Regs out:  A = status register, shifted left by 1
04C8            WAITDR:
04C8    DB10            in      FLOPSR
04CA    07              rlc
04CB    30FB            jrnc    WAITDR
04CD    07              rlc
04CE    C9              ret
                ;----------
                ; onMOTOR - turn on a drive motor
                ;  Regs in:   A = drive number
                ;  Regs out:  none
                ;  Destroyed: A, B
04CF            onMOTOR:
04CF    47              mov     B,A
04D0    3A 04EC         lda     curTRK  ; set precomp
04D3    FE16            cpi     onprecmp
04D5    3808            jrc     ..1
04D7    CBE8            set     5,B     ; bit 5 = pre-comp
04D9    FE3B            cpi     maxprec
04DB    3002            jrnc    ..1
04DD    CBE0            set     4,B     ; bit 4 = S/L
04DF    CBD0    ..1:    set     2,B     ; bit 2 = head load
04E1    78              mov     A,B
04E2    D309            out     PIOBD
04E4    01 1E00         lxi     B,loadset; delay 50ms after
04E7    CD 01CF         call    DELAY   ; head load
04EA    C9              ret
```

```
                        ;----------
                        ; Current disk position information
04EB    00              curDSK: .byte   0       ; current physical drive
04EC    00              curTRK: .byte   0       ; current physical track
04ED    01              curSEC: .byte   1       ; current physical sector
04EE    0100            curBYT: .word   256     ; current length of operation
04F0    3000            curDMA: .word   bufFLCP ; I/C buffer
04F2    01              curDEN: .byte   1       ; current density

                        ;----------
                        ; Floppy controller commands
00FF                    endcom  =       0FFh    ; command terminator
04F3                    homeFLOP:
04F3    07                      .byte   7       ; recalibrate command
04F4    00                      .byte   0       ; curDSK
04F5    FF                      .byte   endcom
04F6                    seekFLOP:
04F6    0F                      .byte   15      ; seek command
04F7    00                      .byte   0       ; curDSK
04F8    00                      .byte   0       ; curTRK
04F9    FF                      .byte   endcom
04FA                    IsenseFLOP:
04FA    08                      .byte   8       ; sense interrupt status
04FB    FF                      .byte   endcom
0046                    readFLOP =      46h     ; DOUBLE DENSITY
0045                    writeFLOP=      45h
04FC    0000            erradr: .word   0               ; err addr holder
04FE    000000000000    resbuf: .byte   0,0,0,0,0,0,0   ; result buffer

                        ;----------
                        ; FORMAT command for Floppy Controller
0505                    FORMcom:
0505    4D                      .byte   4Dh     ; format command (SD/DD)
0506    00              FORMdsk:.byte   0       ; curDSK
0507    01              FORMmod:.byte   1       ; 0=SD, 1=DD
0508    1A                      .byte   26      ; 26 sectors/track
0509    36              FORMgap:.byte   36h     ; gap length
050A    E5                      .byte   0E5h    ; fill byte
050B    FF                      .byte   endcom


                        ;----------
                        ; DMA commands for FORMAT command
050C                    DMAfmt:
050C    C3                      .byte   0C3h    ; master reset
050D    C7                      .byte   0C7h    ; reset port A
050E    CB                      .byte   0CBh    ; reset port B
050F    79                      .byte   079h    ; read from memory
0510    3000                    .word   bufFLOP ; address
0512    0067                    .word   103     ; size - 1
0514    14                      .byte   14h     ; port A inc, memory
0515    28                      .byte   28h     ; port B fixed, I/O
0516    95                      .byte   95h     ; byte mode
0517    18                      .byte   FLOP    ; port B
0518    12                      .byte   12h     ; interrupt at end of block
```

```
0519    00                      .byte   00      ; interrupt vector
051A    9A                      .byte   9Ah     ; stop at end of block
051B    CF                      .byte   0CFh    ; load starting address
051C    05                      .byte   05      ; write to I/O
051D    CF                      .byte   0CFh    ; load starting address
051E    AB                      .byte   0ABh    ; enable interrupts
051F    87                      .byte   87h     ; enable DMA
0014                    DMAfm$  =       .-DMAfmt
3000                    bufFLOP =       3000h   ; location of I/O buffer
0520                            .blkb   100h    ; Room for the stack.
0620                            stack   =       .
                        .end
```

+++++ SYMBOL TABLE +++++

ASKDEN 0286	BUFFLO 3000	COMMAN 04A7	CONIN 047C
CONOUT 048D	CR 000D	CURBYT 04EE	CURDEN 04F2
CURDMA 04F0	CURDSK 04EB	CURSEC 04ED	CURTRK 04EC
DELAY 01CF	DENQST 02B6	DEVQST 017F	DMA 0038
DMADON 03C7	DMAFMT 050C	DMAFM$ 0014	DRIVNO 0004
ENDCOM 00FF	ERRADR 04FC	ERRMSG 03D2	EXFLOP 03AF
EXIT 0495	FLOP 0018	FLOPDR 0011	FLOPSR 0010
FORM1T 0371	FORMAT 01D5	FORMCO 0505	FORMDO 0271
FORMDS 0506	FORMGA 0509	FORMMO 0507	FORMST 0220
HOME 0155	HOMEFL 04F3	INFOMS 0246	IOERR 03E3
ISENSE 04FA	LF 000A	LOADSE 1E00	LOCERR 2FFE
MAXPRE 003B	ONMOTO 04CF	ONPREC 0016	PIOAD 0008
PIORD 0009	PRTBYT 044D	PRTMSG 0441	READDR 04BA
READFL 0046	RESBUF 04FE	RESEEK 01AC	RESULT 04B7
REVISI 0000	RXRDY 0000	SEEK 01A4	SEEKFL 04F6
SELDSK 015F	SENSE 01B1	SETMAP 0003	SETTRK 019C
SIO1AC 002A	SIO1AD 0028	STACK 0620	START 0142
STEPSE 0900	STOPFL 0003	TABLIN 0302	TABSTA 0322
TEST 0106	TFMTFL 0001	TXRDY 0002	USERNO 0047
VERSIO 0002	WAITCR 02D3	WAITDR 04C8	WBOOT 0001
WCRQST 02EA	WRITEF 0045	.BLNK. 0000:03 X	.DATA. 0000* X
.PROG. 0000' X			