```
*****************************************************************
*                                                               *
* CP/M vers 2.2 Cold Start Loader.                              *
*                                                               *
* The following routines will boot CP/M from the Disk           *
* Jockey 2D Rev. B, or from the Disk Jockey Hard disk           *
* controller.                                                   *
*                                                               *
* Floppy boot:                                                  *
* The cold boot loader (track Ø, sector 1) is loaded into       *
* RAM on the controller by the cold boot routine in the         *
* firmware.  This cold boot loader will start loading the       *
* CCP from track Ø, sector 5 and will finish up with the        *
* last part of the CBIOS on track 1 sector 7.                   *
*                                                               *
* During a warm boot sectors 1, 2, and part of 3 will be        *
* loaded from track 1.  Track Ø loading is unaffected.          *
*                                                               *
*   track   sector    sysgen    load order     Name             *
*                                                               *
*       Ø        1       9ØØ    ffØØ     1     Boot loader      *
*       Ø        2       98Ø                   Unused           *
*       Ø        3       aØØ                                    *
*       Ø        4       a8Ø                                    *
*       Ø        5       bØØ    95ØØ     2     CCP              *
*       Ø        6       b8Ø    958Ø    13                      *
*       Ø        7       cØØ    96ØØ     3                      *
*       Ø        8       c8Ø    968Ø    14                      *
*       Ø        9       dØØ    97ØØ     4                      *
*       Ø       1Ø       d8Ø    978Ø    15                      *
*       Ø       11       eØØ    98ØØ     5                      *
*       Ø       12       e8Ø    988Ø    16                      *
*       Ø       13       fØØ    99ØØ     6                      *
*       Ø       14       f8Ø    998Ø    17                      *
*       Ø       15      1ØØØ    9aØØ     7                      *
*       Ø       16      1Ø8Ø    9a8Ø    18                      *
*       Ø       17      11ØØ    9bØØ     8                      *
*       Ø       18      118Ø    9b8Ø    19                      *
*       Ø       19      12ØØ    9cØØ     9                      *
*       Ø       2Ø      128Ø    9c8Ø    2Ø                      *
*       Ø       21      13ØØ    9dØØ    1Ø     BDOS             *
*       Ø       22      138Ø    9d8Ø    21                      *
*       Ø       23      14ØØ    9eØØ    11                      *
*       Ø       24      148Ø    9e8Ø    22                      *
*       Ø       25      15ØØ    9fØØ    12                      *
*       Ø       26      158Ø    9f8Ø    23                      *
*                                                               *
* Track 1 is recorded in double density format.  There are      *
* 1Ø24 bytes per sector.                                        *
*                                                               *
*       1        1      16ØØ    aØØØ     4                      *
*       1        2      1aØØ    a4ØØ     1                      *
*       1        3      1eØØ    a8ØØ     5     CBIOS (@ abØØh)  *
*       1        4      22ØØ    acØØ     2                      *
*       1        5      26ØØ    bØØØ     6                      *
*       1        6      2aØØ    b4ØØ     3                      *
*       1        7      2eØØ    b8ØØ     7                      *
*       1        8      32ØØ    bcØØ           Unused           *
*                                                               *
* Note that the interleave sequences for loading tracks Ø and 1 *
* are different.  This difference was designed so that the      *
* boot sequence could be done in 4 disk revolutions since the   *
* 2D Mod. B can not load consecutive sectors off of the disk.   *
*                                                               *
* Three spare sectors (track Ø, sectors 2 to 4) have been       *
```

Boot Provided with
CBIOS 2.9
8/22/82
(ABOOT&.ASM)

```
* provided for a more advanced boot loader at a later date.     *
*                                                                *
*                                                                *
* Hard boot (M10, M20, M26):                                     *
* The cold boot loader (track 0, sector 1) is loaded into        *
* RAM at either 0100h or the 2DB's RAM depending on whether     *
* this loader is assembled with a 2DB or not.  This cold         *
* boot loader will start loading the CCP from track 0,           *
* sector 2 and will finish up with the last part of the          *
* CBIOS on track 0 sector 21.                                    *
*                                                                *
*   track   sector    sysgen    load order      Name             *
*       0         1      900    fc00     1      Cold boot        *
*       0         2      b00    9500     3      CCP              *
*       0         3      d00    9700     4                       *
*       0         4      f00    9900     5                       *
*       0         5     1100    9b00     6                       *
*       0         6     1300    9d00     7      BDOS             *
*       0         7     1500    9f00     8                       *
*       0         8     1700    a100     9                       *
*       0         9     1900    a300    10                       *
*       0        10     1b00    a500    11                       *
*       0        11     1d00    a700    12                       *
*       0        12     1f00    a900    13                       *
*       0        13     2100    ab00    14      CBIOS            *
*       0        14     2300    ad00    15                       *
*       0        15     2500    af00    16                       *
*       0        16     2700    b100    17                       *
*       0        17     2900    b300    18                       *
*       0        18     2b00    b500    19                       *
*       0        19     2d00    b700    20                       *
*       0        20     2f00    b900     2      Partial load     *
*       0        21     3000                    Unused           *
*                                                                *
* The warm boot load sequence starts at track 0, sector 2        *
* and goes straight through to sector 12.  There is still        *
* plenty of room left in this loader for more advanced           *
* things like sector interleaving although this is hardly        *
* necessary on a hard disk.                                      *
*                                                                *
******************************************************************

msize   equ     48              ;Memory size of target CP/M
bias    equ     (msize-20)*1024 ;Memory offset from 20k system
ccp     equ     2500h+bias      ;Console command processor
bios    equ     ccp+1600h       ;CBIOS address
cboot   equ     bios            ;Cold boot address for CP/M
loaddr  equ     ccp             ;Load address for floppy
retries equ     10              ;Maximum # of disk retries

******************************************************************
*                                                                *
* The following equates set up the relationship between the      *
* 2D floppies and the Hard Disk Controllers.                     *
*                                                                *
******************************************************************

first   equ     1               ;0 = Floppies are A-D drives and
                                ;       Hard Disk are E-P
                                ;1 = Hard Disks are A-L drives and
                                ;       Floppies are M-P
maxhd   equ     1               ;Set to number of hard disks
maxflop equ     4               ;Set to number of floppies

******************************************************************
*                                                                *
* The following equates are for the Diskus Hard disk if wanted. *
*                                                               *
*****************************************************************

        if      (maxhd ne 0) and first  ;Want Hard Disk included ?
hdorg   equ     50h             ;Hard Disk Controller
hdstat  equ     hdorg           ;Hard Disk Status
hdcntl  equ     hdorg           ;Hard Disk Control
hddata  equ     hdorg+3         ;Hard Disk Data
hdfunc  equ     hdorg+2         ;Hard Disk Function
hdcmnd  equ     hdorg+1         ;Hard Disk Command
hdreslt equ     hdorg+1         ;Hard Disk Result
retry   equ     2               ;Retry bit of result
tkz     equ     1               ;Track zero bit of status
opdone  equ     2               ;Operaction done bit of status
complt  equ     4               ;Complete bit of status
tmout   equ     8               ;Time out bit of status
wfault  equ     10h             ;Write fault bit of status
drvrdy  equ     20h             ;Drive ready bit of status
indx    equ     40h             ;Index bit of status
pstep   equ     4               ;Step bit of function
nstep   equ     0fbh            ;Step bit mask of function
hdrlen  equ     4               ;Sector header length
secln   equ     512             ;Sector data length
wenabl  equ     0fh             ;Write enable
wreset  equ     0bh             ;Write reset of function
scenbl  equ     5               ;Controller control
dskclk  equ     7               ;Disk clock for control
mdir    equ     0f7h            ;Direction mask for function
null    equ     0fch            ;Null command
idbuff  equ     0               ;Initialize data command
isbuff  equ     8               ;Initialize header command
rsect   equ     1               ;Read sector command
wsect   equ     5               ;Write sector command
        endif

*****************************************************************
*                                                               *
* The following equates are for the Disk Jockey 2D/B if wanted. *
*                                                               *
*****************************************************************

        if      maxflop ne 0
origin  equ     0f800h          ;Orgin of DJ 2D Mod B PROM
djram   equ     origin+400h
                                ;Disk Jockey 2D Mod B routines
tkzero  equ     origin+9h       ;Track 0 seek
trkset  equ     origin+0ch      ;Set track
setsec  equ     origin+0fH      ;Set sector
setdma  equ     origin+12h      ;Set DMA address
dread   equ     origin+15h      ;Read sector
dmast   equ     origin+24h      ;Get DMA address
status  equ     origin+27h      ;Disk status
dskerr  equ     origin+2ah      ;Flash error light
setden  equ     origin+2dh      ;Set density
        endif

if      first                   ;Define start address if hard disk
        if      maxflop ne 0
boot    equ     djram           ;If floppy is there then use its RAM
        else
boot    equ     0100h           ;Otherwise start at 0100h
        endif
        else                    ;Define start address if floppy
boot    equ     djram+0300h     ;Upper quarter of floppy RAM
        endif
```

```
offset  equ     9ØØh-boot       ;DDT offset

*****************************************************************
*                                                               *
* Cold Boot loader for Discus M1Ø, M2Ø, or M26.                 *
*                                                               *
*****************************************************************

        if      first           ;first = 1 is hard disk

        org     boot

boothd  lxi     sp,cstkhd       ;Set up stack at end of this sector
        lxi     b,1*1ØØh+2Ø     ;B = sector count, C = sector #
        call    clodhd          ;Load sector 2Ø into CCP
        lxi     h,ccp+1eØØh     ;Destination of move
        lxi     d,ccp           ;Source of move
        mvi     c,Ø
cmovhd  ldax    d               ;Get a byte of source
        mov     m,a             ;Move it
        inx     h               ;Bump destination
        inx     d               ;Bump source
        dcr     c               ;All done with this page ?
        jnz     cmovhd
        lxi     h,ccp-2ØØh      ;Initial DMA address
        shld    cdmahd
        lxi     b,18*1ØØh+2     ;B = sector count, C = sector #
        call    clodhd
        jmp     cboot           ;Go to CP/M

clodhd  push    b               ;Save sector and count
        mov     a,c
        sta     hdsec
        lxi     h,ccp-2ØØh      ;Get DMA address (self modifying)
cdmahd  equ     $-2             ;Storage for previous DMA address
        lxi     d,2ØØh          ;Offset to new DMA address
        dad     d               ;Add in offset, HL = new DMA address
        shld    cdmahd          ;Save new DMA address
        call    crdhd           ;Attempt a read
        pop     b               ;Recover sector number and count
                                ;       B = count, C = number
        dcr     b               ;Update sector count
        rz                      ;All done ?
        inr     c
        jmp     clodhd          ;Continue reading

*****************************************************************
*                                                               *
* Rdhd does the actual read from the controller, the DMA        *
* address and sector # have already been set up.                *
*                                                               *
*****************************************************************

crdhd   lxi     b,retries*1ØØh+1 ;Maximum # of attempts
crhd    push    b               ;Save error count
        call    hdread          ;Attempt the read
        pop     b               ;Restore the error count
        rnc                     ;Return if no error
        dcr     b               ;Update error count
        jnz     crhd            ;Try again if not to many errors
        jmp     $               ;Dynamic error halt

hdread  call    hdprep          ;Prepare the sector header image
        rc                      ;Error exit
        mvi     a,rsect         ;Read sector command
        out     hdcmnd
        call    process         ;Process the read
        rc                      ;Error exit
        xra     a               ;Pointer to data buffer
        out     hdcmnd
        mvi     b,secln/4       ;Number of bytes to read
        lhld    cdmahd          ;Get destination of data
        in      hddata          ;Two dummy data bytes
        in      hddata
rtloop  in      hddata          ;Move four bytes
        mov     m,a             ;Byte one
        inx     h
        in      hddata          ;Byte two
        mov     m,a
        inx     h
        in      hddata          ;Byte three
        mov     m,a
        inx     h
        in      hddata          ;Byte four
        mov     m,a
        inx     h
        dcr     b               ;Update byte count
        jnz     rtloop
        ret

process in      hdstat          ;Wait for command to finish
        mov     b,a
        ani     opdone
        jz      process
        mvi     a,dskclk        ;Turn on Disk Clock
        out     hdcntl
        in      hdstat
        ani     tmout           ;Timed out ?
        stc
        rnz
        in      hdreslt
        ani     retry           ;Any retries ?
        stc
        rnz
        xra     a               ;No error exit
        ret

hdprep  in      hdstat          ;Is Drive ready ?
        ani     drvrdy
        stc
        rnz
        mvi     a,isbuff        ;Initialize pointer to header buffer
        out     hdcmnd
        mvi     a,null
        out     hdfunc          ;Select drive A
        xra     a
        out     hddata          ;Form head byte
        out     hddata          ;Form track byte
        mvi     a,Ø             ;Form sector byte
hdsec   equ     $-1
        out     hddata
        mvi     a,8Øh           ;Form Key
        out     hddata
        mvi     a,dskclk        ;Turn on Disk clock
        out     hdcntl
        mvi     a,wenabl        ;Write enable on
        out     hdcntl
        ret

        org     boothd+2ØØh-2
```

```
cstkhd  equ     $
        dw      boothd

        else                    ;first = Ø is floppy disk

*****************************************************************
*                                                               *
* Cold boot loader for the Disk Jockey 2D Revision B controller *
*                                                               *
*****************************************************************

        org     boot

tØboot  mvi     a,5-2           ;First sector - 2
newsec  equ     $-1
        inr     a               ;Update sector #
        inr     a
        cpi     27              ;Size of track in sectors + 1
trksiz  equ     $-1
        jc      nowrap          ;Skip if not at end of track
        jnz     t1boot          ;Done with this track
exit    equ     $-2
        sui     27-6            ;Back up to sector 6
backup  equ     $-1
        lxi     h,loaddr-8Øh    ;Memory address of sector - 1ØØh
nxtdma  equ     $-2
        shld    newdma
nowrap  sta     newsec          ;Save the updated sector #
        mov     c,a
        call    setsec          ;Set up the sector
        lxi     h,loaddr-1ØØh   ;Memory address of sector - 1ØØh
newdma  equ     $-2
        lxi     d,1ØØh          ;Update DMA address
secsiz  equ     $-2
        dad     d
nowrp   shld    newdma          ;Save the updated DMA address
        mov     b,h
        mov     c,l
        call    setdma          ;Set up the new DMA address
        lxi     b,retries*1ØØh+Ø;Maximum # of errors, track #
nxtrty  equ     $-2
fread   push    b
        call    trkset          ;Set up the proper track
        call    dread           ;Read the sector
        pop     b
        jnc     tØboot          ;Continue if no error
        dcr     b
        jnz     fread           ;Keep trying if error
        jmp     dskerr          ;Too many errors, flash the light

t1boot  lxi     h,cboot         ;We jump to cboot next time
        shld    exit
        mvi     c,1             ;Select double density
        call    setden
        xra     a               ;First sector - 2
        sta     newsec
        mvi     a,8             ;Size of (logical) track + 1
        sta     trksiz
        dcr     a               ;Number of sectors to back up
        sta     backup
        lxi     h,loaddr+Ø7ØØh  ;DMA start address for first revolution - 2Ø48
        shld    newdma
        lxi     h,loaddr+Ø3ØØh  ;DMA start address for second revolution - 2Ø48
        shld    nxtdma
        lxi     h,2Ø48          ;Difference between DMA addresses
        shld    secsiz
        lxi     h,retries*1ØØh+1;Maximum # of errors, track #
        shld    nxtrty
        jmp     tØboot          ;Go load in track 1
        endif

        end
```