

```

*****
*
* 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 0, 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 0, 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 0 LOADING IS UNAFFECTED.

```

ABOOTH1.PRN
8/24/82
SAME AS ABOOT8. except
62K

TRACK	SECTOR	SYSGEN	LOAD ORDER	NAME
0	1	900	FF00 1	BOOT LOADER
0	2	980		UNUSED
0	3	A00		
0	4	A80		
0	5	B00	9500 2	CCP
0	6	B80	9580 13	
0	7	C00	9600 3	
0	8	C80	9680 14	
0	9	D00	9700 4	
0	10	D80	9780 15	
0	11	E00	9800 5	
0	12	E80	9880 16	
0	13	F00	9900 6	
0	14	F80	9980 17	
0	15	1000	9A00 7	
0	16	1080	9A80 18	
0	17	1100	9B00 8	
0	18	1180	9B80 19	
0	19	1200	9C00 9	
0	20	1280	9C80 20	
0	21	1300	9D00 10	BDOS
0	22	1380	9D80 21	
0	23	1400	9E00 11	
0	24	1480	9E80 22	
0	25	1500	9F00 12	
0	26	1580	9F80 23	

```

* TRACK 1 IS RECORDED IN DOUBLE DENSITY FORMAT. THERE ARE
* 1024 BYTES PER SECTOR.

```

1	1	1600	A000 4	
1	2	1A00	A400 1	
1	3	1E00	A800 5	CBIOS (@ AB00H)
1	4	2200	AC00 2	
1	5	2600	B000 6	
1	6	2A00	B400 3	

VR-1412

WISH-HEAD

```

*      1      7      2E00      B800      7
*      1      8      3200      BC00      UNUSED
*

```

```

* NOTE THAT THE INTERLEAVE SEQUENCES FOR LOADING TRACKS 0 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 0, SECTORS 2 TO 4) HAVE BEEN
* 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	COLD BOOT
0	2	B00	9500	CCP
0	3	D00	9700	
0	4	F00	9900	
0	5	1100	9B00	
0	6	1300	9D00	BDOS
0	7	1500	9F00	
0	8	1700	A100	
0	9	1900	A300	
0	10	1B00	A500	
0	11	1D00	A700	
0	12	1F00	A900	
0	13	2100	AB00	CBIOS
0	14	2300	AD00	
0	15	2500	AF00	
0	16	2700	B100	
0	17	2900	B300	
0	18	2B00	B500	
0	19	2D00	B700	
0	20	2F00	B900	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.

```

```

*****

```

```

003E = MSIZE EQU 62 ;MEMORY SIZE OF TARGET CP/M
A800 = BIAS EQU (MSIZE-20)*1024 ;MEMORY OFFSET FROM 20K SYSTEM
CD00 = CCP EQU 2500H+BIAS ;CONSOLE COMMAND PROCESSOR
E300 = BIOS EQU CCP+1600H ;CBIOS ADDRESS
E300 = CBOOT EQU BIOS ;COLD BOOT ADDRESS FOR CP/M

```

VR-1412

VISH-READ

CD00 = LOADDR EQU CCP ;LOAD ADDRESS FOR FLOPPY
000A = RETRIES EQU 10 ;MAXIMUM # OF DISK RETRIES

*
* THE FOLLOWING EQUATES SET UP THE RELATIONSHIP BETWEEN THE *
* 2D FLOPPIES AND THE HARD DISK CONTROLLERS. *
*

0001 = 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
0001 = MAXHD EQU 1 ;SET TO NUMBER OF HARD DISKS
0004 = MAXFLOP EQU 4 ;SET TO NUMBER OF FLOPPIES

*
* THE FOLLOWING EQUATES ARE FOR THE DISKUS HARD DISK IF WANTED. *
*

0050 = HDORG EQU 50H ;HARD DISK CONTROLLER
0050 = HDSTAT EQU HDORG ;HARD DISK STATUS
0050 = HDCNTL EQU HDORG ;HARD DISK CONTROL
0053 = HDDATA EQU HDORG+3 ;HARD DISK DATA
0052 = HDFUNC EQU HDORG+2 ;HARD DISK FUNCTION
0051 = HDCMND EQU HDORG+1 ;HARD DISK COMMAND
0051 = HDRESLT EQU HDORG+1 ;HARD DISK RESULT
0002 = RETRY EQU 2 ;RETRY BIT OF RESULT
0001 = TKZ EQU 1 ;TRACK ZERO BIT OF STATUS
0002 = OPDONE EQU 2 ;OPERATION DONE BIT OF STATUS
0004 = COMPLT EQU 4 ;COMPLETE BIT OF STATUS
0008 = TMOU EQU 8 ;TIME OUT BIT OF STATUS
0010 = WFAULT EQU 10H ;WRITE FAULT BIT OF STATUS
0020 = DRVRDY EQU 20H ;DRIVE READY BIT OF STATUS
0040 = INDX EQU 40H ;INDEX BIT OF STATUS
0004 = PSTEP EQU 4 ;STEP BIT OF FUNCTION
00FB = NSTEP EQU 0FBH ;STEP BIT MASK OF FUNCTION
0004 = HDRLN EQU 4 ;SECTOR HEADER LENGTH
0200 = SECLN EQU 512 ;SECTOR DATA LENGTH
000F = WENABL EQU 0FH ;WRITE ENABLE
000B = WRESET EQU 0BH ;WRITE RESET OF FUNCTION
0005 = SCENBL EQU 5 ;CONTROLLER CONTROL
0007 = DSKCLK EQU 7 ;DISK CLOCK FOR CONTROL
00F7 = MDIR EQU 0F7H ;DIRECTION MASK FOR FUNCTION
00FC = NULL EQU 0FCH ;NULL COMMAND
0000 = IDBUFF EQU 0 ;INITIALIZE DATA COMMAND
0008 = ISBUFF EQU 8 ;INITIALIZE HEADER COMMAND
0001 = RSECT EQU 1 ;READ SECTOR COMMAND
0005 = WSECT EQU 5 ;WRITE SECTOR COMMAND

ENDIF

VR-1412

*
 * THE FOLLOWING EQUATES ARE FOR THE DISK JOCKEY 2D/B IF WANTED. *
 *

```

F800 = ORIGIN EQU MAXFLOP NE 0 ;ORIGIN OF DJ 2D MOD B PROM
FC00 = DJRAM EQU ORIGIN+400H ;DISK JOCKEY 2D MOD B ROUTINES

F809 = TKZERO EQU ORIGIN+9H ;TRACK 0 SEEK
F80C = TRKSET EQU ORIGIN+0CH ;SET TRACK
F80F = SETSEC EQU ORIGIN+0FH ;SET SECTOR
F812 = SETDMA EQU ORIGIN+12H ;SET DMA ADDRESS
F815 = DREAD EQU ORIGIN+15H ;READ SECTOR
F824 = DMAST EQU ORIGIN+24H ;GET DMA ADDRESS
F827 = STATUS EQU ORIGIN+27H ;DISK STATUS
F82A = DSKERR EQU ORIGIN+2AH ;FLASH ERROR LIGHT
F82D = SETDEN EQU ORIGIN+2DH ;SET DENSITY
      ENDIF
  
```

```

      IF FIRST ;DEFINE START ADDRESS IF HARD DISK
FC00 = BOOT EQU MAXFLOP NE 0 ;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
  
```

```

0D00 = OFFSET EQU 900H-BOOT ;DDT OFFSET
  
```

 *
 * COLD BOOT LOADER FOR DISCUS M10, M20, OR M26.
 *

```

      IF FIRST ;FIRST = 1 IS HARD DISK
FC00 ORG BOOT

FC00 31FEFD BOOTHDX LXI SP,CSTKHD ;SET UP STACK AT END OF THIS SECTOR
FC03 011401 LXI B,1*100H+20 ;B = SECTOR COUNT, C = SECTOR #
FC06 CD28FC CALL CLODHD ;LOAD SECTOR 20 INTO CCP
FC09 2100EB LXI H,CCP+1E00H ;DESTINATION OF MOVE
FC0C 1100CD LXI D,CCP ;SOURCE OF MOVE
FC0F 0E00 MVI C,0
FC11 1A CMOVHDX LDAX D ;GET A BYTE OF SOURCE
FC12 77 MOV M,A ;MOVE IT
FC13 23 INX H ;BUMP DESTINATION
FC14 13 INX D ;BUMP SOURCE
FC15 0D DCR C ;ALL DONE WITH THIS PAGE ?
FC16 C211FC JNZ CMOVHDX
FC19 2100CB LXI H,CCP-200H ;INITIAL DMA ADDRESS
FC1C 222EFC SHLD CDMAHD
  
```

VR-1412

VISI-READ

```

FC1F 010212      LXI      B,18*100H+2      ;B = SECTOR COUNT, C = SECTOR #
FC22 CD28FC      CALL     CLODHD
FC25 C300E3      JMP      CBOOT           ;GO TO CP/M

```

```

FC28 C5          CLODHD  PUSH   B           ;SAVE SECTOR AND COUNT

```

```

FC29 79          MOV     A,C
FC2A 32ACFC      STA     HDSEC
FC2D 2100CB      LXI     H,CCP-200H      ;GET DMA ADDRESS (SELF MODIFYING)
FC2E =           CDMAHD  EQU     $-2       ;STORAGE FOR PREVIOUS DMA ADDRESS
FC30 110002      LXI     D,200H         ;OFFSET TO NEW DMA ADDRESS
FC33 19          DAD     D             ;ADD IN OFFSET, HL = NEW DMA ADDRESS

```

```

FC34 222EFC      SHLD   CDMAHD         ;SAVE NEW DMA ADDRESS
FC37 CD41FC      CALL   CRDHD         ;ATTEMPT A READ
FC3A C1          POP     B             ;RECOVER SECTOR NUMBER AND COUNT

```

```

;           B = COUNT, C = NUMBER
FC3B 05          DCR     B             ;UPDATE SECTOR COUNT
FC3C C8          RZ
;           ;ALL DONE ?

```

```

FC3D 0C          INR     C
FC3E C328FC      JMP     CLODHD        ;CONTINUE READING

```

```

*****
*
* RDHD DOES THE ACTUAL READ FROM THE CONTROLLER, THE DMA
* ADDRESS AND SECTOR # HAVE ALREADY BEEN SET UP.
*
*****

```

```

FC41 01010A      CRDHD  LXI     B,RETRIES*100H+1 ;MAXIMUM # OF ATTEMPTS
FC44 C5          CRHD   PUSH   B           ;SAVE ERROR COUNT
FC45 CD51FC      CALL   HDREAD        ;ATTEMPT THE READ
FC48 C1          POP     B           ;RESTORE THE ERROR COUNT
FC49 D0          RNC
;           ;RETURN IF NO ERROR
FC4A 05          DCR     B           ;UPDATE ERROR COUNT
FC4B C244FC      JNZ    CRHD         ;TRY AGAIN IF NOT TO MANY ERRORS
FC4E C34EFC      JMP     $           ;DYNAMIC ERROR HALT

```

```

FC51 CD98FC      HDREAD CALL   HDPREP        ;PREPARE THE SECTOR HEADER IMAGE
FC54 DB          RC
;           ;ERROR EXIT
FC55 3E01        MVI     A,RSECT      ;READ SECTOR COMMAND
FC57 D351        OUT    HDCMND
FC59 CD7EFC      CALL   PROCESS      ;PROCESS THE READ
FC5C DB          RC
;           ;ERROR EXIT
FC5D AF          XRA     A           ;POINTER TO DATA BUFFER
FC5E D351        OUT    HDCMND

```

```

FC60 0680        MVI     B,SECLN/4    ;NUMBER OF BYTES TO READ
FC62 2A2EFC      LHL    CDMAHD       ;GET DESTINATION OF DATA
FC65 DB53        IN     HDDATA       ;TWO DUMMY DATA BYTES

```

```

FC67 DB53        IN     HDDATA
FC69 DB53        RTLOOP IN     HDDATA      ;MOVE FOUR BYTES
FC6B 77          MOV     M,A         ;BYTE ONE

```

```

FC6C 23          INX    H
FC6D DB53        IN     HDDATA       ;BYTE TWO
FC6F 77          MOV     M,A

```

```

FC70 23          INX    H
FC71 DB53        IN     HDDATA       ;BYTE THREE
FC73 77          MOV     M,A

```

VR-1412

```

FC74 23      INX      H
FC75 DB53    IN      HDDATA      ;BYTE FOUR
FC77 77      MOV      M,A
FC78 23      INX      H
FC79 05      DCR      B      ;UPDATE BYTE COUNT
FC7A C269FC  JNZ      RTLOOP
FC7D C9      RET

```

```

FC7E DB50    PROCESS IN      HDSTAT      ;WAIT FOR COMMAND TO FINISH
FC80 47      MOV      B,A
FC81 E602    ANI      OPDONE

```

```

FC83 CA7EFC  JZ       PROCESS
FC86 3E07    MVI      A,DSKCLK      ;TURN ON DISK CLOCK
FC88 D350    OUT     HDCNTL

```

```

FC8A DB50    IN      HDSTAT
FC8C E608    ANI      TMOUT      ;TIMED OUT ?
FC8E 37      STC

```

```

FC8F C0      RNZ
FC90 DB51    IN      HDRESLT
FC92 E602    ANI      RETRY      ;ANY RETRIES ?

```

```

FC94 37      STC
FC95 C0      RNZ
FC96 AF      XRA      A      ;NO ERROR EXIT
FC97 C9      RET

```

```

FC98 DB50    HDPREP  IN      HDSTAT      ;IS DRIVE READY ?
FC9A E620    ANI      DRVRDY
FC9C 37      STC
FC9D C0      RNZ

```

```

FC9E 3E08    MVI      A,ISBUFF      ;INITIALIZE POINTER TO HEADER BUFFER
FCA0 D351    OUT     HDCMND
FCA2 3EFC    MVI      A,NULL

```

```

FCA4 D352    OUT     HDFUNC      ;SELECT DRIVE A
FCA6 AF      XRA      A
FCA7 D353    OUT     HDDATA      ;FORM HEAD BYTE

```

```

FCA9 D353    OUT     HDDATA      ;FORM TRACK BYTE
FCAB 3E00    MVI      A,0      ;FORM SECTOR BYTE
FCAC =      HDSEC  EQU     $-1

```

```

FCAD D353    OUT     HDDATA
FCAF 3E80    MVI      A,80H      ;FORM KEY
FCB1 D353    OUT     HDDATA

```

```

FCB3 3E07    MVI      A,DSKCLK      ;TURN ON DISK CLOCK
FCB5 D350    OUT     HDCNTL
FCB7 3E0F    MVI      A,WENABL      ;WRITE ENABLE ON

```

```

FCB9 D350    OUT     HDCNTL
FCBB C9      RET

```

```

FDFE          ORG      BOOTHD+200H-2

```

```

FDFE =      CSTKHD EQU     $
FDFE 00FC    DW      BOOTHD

```

```

ELSE          ;FIRST = 0 IS FLOPPY DISK

```

```

*****
*

```

VR-1412

VISI-READ

* COLD BOOT LOADER FOR THE DISK JOCKEY 2D REVISION B CONTROLLER *
 *

	ORG	BOOT	
T0BOOT	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	
BACKUP	SUI	27-6	;BACK UP TO SECTOR 6
	EQU	\$-1	
NXTDMA	LXI	H,LOADDR-80H	;MEMORY ADDRESS OF SECTOR - 100H
	EQU	\$-2	
	SHLD	NEWDMA	
NOWRAP	STA	NEWSEC	;SAVE THE UPDATED SECTOR #
	MOV	C,A	
	CALL	SETSEC	;SET UP THE SECTOR
NEWDMA	LXI	H,LOADDR-100H	;MEMORY ADDRESS OF SECTOR - 100H
	EQU	\$-2	
	LXI	D,100H	;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
NXTRTY	LXI	B,RETRIES*100H+0	;MAXIMUM # OF ERRORS, TRACK #
FREAD	EQU	\$-2	
	PUSH	B	
	CALL	TRKSET	;SET UP THE PROPER TRACK
	CALL	DREAD	;READ THE SECTOR
	POP	B	
	JNC	T0BOOT	;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+0700H	;DMA START ADDRESS FOR FIRST REVOLUTION - 2048
	SHLD	NEWDMA	
	LXI	H,LOADDR+0300H	;DMA START ADDRESS FOR SECOND REVOLUTION - 2048

VR-1412

VISI-HEAD

```

SHLD  NXTDMA
LXI   H,2048           ;DIFFERENCE BETWEEN DMA ADDRESSES
SHLD  SECSIZ
LXI   H,RETRIES*100H+1;MAXIMUM # OF ERRORS, TRACK #
SHLD  NXTRTY
JMP   T0BOOT          ;GO LOAD IN TRACK 1
ENDIF

```

```
FE00      END
```

A800 BIAS	E300 BIOS	FC00 BOOT	FC00 BOOTH	E300 CBOOT
CD00 CCP	FC2E CDMAHD	FC28 CLODHD	FC11 CMOVHD	0004 COMPLT
FC41 CRDHD	FC44 CRHD	EDFE CSTKHD	FC00 DJRAM	F824 DMAST
F815 DREAD	0020 DRVRDY	0007 DSKCLK	F82A DSKERR	0001 FIRST
0051 HDCMND	0050 HDCNTL	0053 HDDATA	0052 HDFUNC	0050 HDORG
FC98 HDPREP	FC51 HDREAD	0051 HDRESLT	0004 HDRLEN	FCAC HDSEC
0050 HDSTAT	0000 IDBUFF	0040 INDX	0008 ISBUFF	CD00 LOADDR
0004 MAXFLOP	0001 MAXHD	00F7 MDIR	003E MSIZE	00FB NSTEP
00FC NULL	0D00 OFFSET	0002 OPDONE	F800 ORIGIN	FC7E PROCESS
0004 PSTEP	000A RETRIES	0002 RETRY	0001 RSECT	FC69 RTLOOP
0005 SCENBL	0200 SECLN	F82D SETDEN	F812 SETDMA	F80F SETSEC
F827 STATUS	0001 TKZ	F809 TKZERO	0008 TMOUT	F80C TRKSET
000F WENABL	0010 WFAULT	000B WRESET	0005 WSECT	