

```

1
2      ;*****
3      ;*
4      ;*      TCOPY - Example program to write the system track
5      ;*      for a Concurrent CP/M-86 Boot Disk on the
6      ;*      IBM Personal Computer
7      ;*
8      ;*****
9
10     ;      This program is used to read a binary image file of
11     ;      track 0. This track is used to bootstrap Concurrent
12     ;      CP/M-86. The file TCOPY reads has no CHD header and
13     ;      must be the same size as the track we are going
14     ;      to write.
15
16     ;      This program is intended to serve as an example
17     ;      to be modified by the OEM for differently sized loaders,
18     ;      and differently sized system track(s).
19
20     ;      Note: TCOPY must be run under CP/M-86 and not Concurrent
21     ;      CP/M-86 since TCOPY performs direct BIOS calls.
22
23     ;      The command
24     ;      GENCMD TCOPY
25     ;      is used to generate the CHD form of this program.
26
27     title "TCOPY - Copy Track 0"
28
29     ;CP/M-86, CCP/M-86 function names
30
31     ;console functions
32     0001 c_read equ 1
33     0009 c_writebuf equ 9
34
35     ;file functions
36     000F f_open equ 15
37     0021 f_readrand equ 33
38     001A f_setdma equ 26
39     0033 f_setdmaseg equ 51
40
41     ;drive functions
42     0019 drv_get equ 25
43
44     ;system functions
45     0000 s_termcpm equ 0
46     0032 s_aibios equ 50
47
48     ;direct Bios Parameter Block
49     0000 bpb_func equ byte ptr 0
50     0001 bpb_cx equ word ptr 1
51     0003 bpb_dx equ word ptr 3
52
53

```

CCP/M-86 2.0 V.2
 COPYRIGHT © 1981, 1982, 1983
 DIGITAL RESEARCH
 P. O. BOX 579
 PACIFIC GROVE, CA 93950
 SER. # CC-104

```

54
55      ;ASCII linefeed and carriage return
56      000A      lf      equ      10
57      000D      cr      equ      13
58
59      ;how many 128 byte records to read for a loader image
60      0020      records_to_read      equ      8 * 4
61
62      ;8 = number of physical sectors per track
63      ;4 = number of 128 sectors per
64      ;physical sector
65
66      cseg      ;use CCP stack
67      0000 8109      mov cl,c_writebuf      ;display sign on message
68      0002 8A0001      mov dx,offset sign_on_msg
69      0005 CDE0      int 224
70      0007 B119      mov cl,drv_get      ;get default drive number
71      0009 CDE0      int 224
72      000B 84C0      test al,al      ;must run on drive A:
73      000F 8AA001      jz drive_ok
74      0012 E9D200      mov dx,offset drive_msg
75      00E7      jmp error
76
77      drive_ok:
78      0015 810F      mov cl,f_open      ;open the file given as
79      0017 BA5C00      mov dx,offset fcb      ;the 1st command parameter,
80      001A CDE0      int 224      ;it is put at 05CH by
81      001E 7506      cmp al,0ffh      ;the program load
82      0020 BAC701      jne file_ok
83      0023 E9C100      mov dx,offset open_msg
84      00E7      jmp error
85
86      file_ok:
87      0026 C7063A123502      mov current_dma,offset track0_buffer
88      002C C7067D0000000      mov r0,0      ;start with sector 0, assume
89      0032 B92000      mov cx,records_to_read ;no CMD header in the file
90
91      file_read:
92      0035 51      push cx      ;keep the record count
93      0036 B11A      mov cl,f_setdma
94      0038 8B163A12      mov dx,current_dma
95      003C CDE0      int 224
96      003E B121      mov cl,f_readrand      ;user r0,r1,r2 for random
97      0040 BA5C00      mov dx,offset fcb      ;reads
98      0043 CDE0      int 224
99      0045 59      pop cx      ;restore the record count
100      0046 84C0      test al,al
101      0048 7406      jz read_ok
102      004A BA0402      mov dx,offset read_msg
103      004D E99700      jmp error
104
105      read_ok:
106      0050 81063A128000      add current_dma,128      ;set the DMA for the next sector
107      0056 FF057D00      inc r0      ;add one to the random record field
108      005A E2D9      loop file_read

```

COPYRIGHT © 1981, 1982, 1983
 DIGITAL RESEARCH
 P. O. BOX 579
 PACIFIC GROVE, CA 93950

SER. # _____

```

107
108
109 ; we have the Track 0 image in RAM
110 ; Ask for destination diskette
111
112 next_diskette:
113
114 005C B109      mov cx,c_writebuf
115 005E BA4A01    mov dx,offset new_disk_msg
116 0061 CDE0      int 224
117
118 0063 B101      mov cx,c_read      ;wait for a keystroke
119 0065 CDE0      int 224
120
121 ; Using CP/M-86 function 50, Direct bios call,
122 ; write the track image in TRACK0_BUFFER to
123 ; track 0, on drive A:.
124
125 0067 E82F00    0099      call select_disk      ;select A:
126 006A E83400    00A1      call set_track      ;set track to 0
127 006D E83700    00A7      call set_dmaseg      ;set DMA segment = 05
128
129 0070 C7053C120000      mov current_sector,0      ;sectors are relative to 0 in BIOS
130 0076 C7063A123502      mov current_dma,offset track0_buffer
131 007C B92000      mov cx,32      ;number of 128 byte sectors to write
132
133 next_sector:
134 007F 51          push cx      ;save sector count
135 0080 E82400      00A0      call set_dmaoff
136 0083 E82F00      00A5      call set_sector
137 0086 E83400      00B0      call write_sector
138 0089 81063A128000      add current_dma,128      ;next area of memory to write
139 008F FF063C12      inc current_sector      ;next sector number
140 0093 59          pop cx      ;restore sector count
141 0094 E2E9      007F      loop next_sector
142 0096 E93A00      00D3      jmp track_ok
143
144 select_disk:
145 0099 B009      mov al,9      ;BIOS function number of seldsk
146 009B 33C9      xor cx,cx      ;always drive A:
147 009D 88D1      mov dx,cx
148 009F E820      00C1      jmps bios
149
150 set_track:
151 00A1 B00A      mov al,10      ;BIOS function number of settrk
152 00A3 33C9      xor cx,cx      ;go to track 0
153 00A5 EB1A      00C1      jmps bios
154
155 set_dmaseg:
156 00A7 B011      mov al,17      ;BIOS function number of setdma
157 00A9 8CD9      mov cx,dx      ;dma segment we want to use
158 00AB EB14      00C1      jmps bios
159
160 set_dmaoff:
161 00AD B00C      mov al,12      ;BIOS function number of setdma
162 00AF 8B0E3A12      mov cx,current_dma
163 00B3 EB0C      00C1      jmps bios

```

COPYRIGHT © 1981, 1982, 1983
 DIGITAL RESEARCH
 P. O. BOX 579
 PACIFIC GROVE, CA 93950

SER. # _____

```

160
161      set_sector:
162      00B5 800B      mov al,11      ;BIOS function number of setsec
163      00B7 8B0E3C12  mov cx,current_sector
164      00B8 EB04      jmps bios      00C1
165      write_sector:
166      00B9 B00E      mov al,14      ;BIOS function number of write sector
167      00BF EB00      jmps bios      00C1      ;error checking can be added here
168
169      bios:
170      00C1 B83512     mov bx,offset bpb      ;fill in BIOS Parameter Block
171      00C4 8807      mov bpb_func[bx],al
172      00C6 894F01     mov bpb_cx[bx],cx
173      00C9 895703     mov bpb_dx[bx],dx
174      00CC 8132      mov cl,s_dirbios
175      00CE 8B93      mov dx,bx
176      00D0 CDE0      int 224
177      00D2 C3        ret
178
179      track_ok:
180      00D3 8109      mov cl,c_writebuf      ;does the user want to write
181      00D5 BA8701     mov dx,offset continue_msg      ;to another diskette ?
182      00D8 CDE0      int 224
183      00DA 8101      mov cl,c_read      ;get response
184      00DC CDE0      int 224
185      00DE 245F      and al,05FH      ;make upper case
186      00E0 3C59      cmp al,"Y"
187      00E2 750C      jne done      00F0
188      00E4 E975FF     jmp next_diskette      005C
189
190      error:
191      00E7 52        push dx
192      00E8 EB0C00     call crlf      00F7
193      00EB 5A        pop dx
194      00EC 8109      mov cl,c_writebuf
195      00EE CDE0      int 224
196
197      done:
198      00F0 B90000     mov cx,s_termcpm
199      00F3 8B01      mov dx,cx
200      00F5 CDE0      int 224
201
202      crlf:
203      00F7 BAA801     mov dx,offset crlf_msg
204      00FA 8109      mov cl,c_writebuf
205      00FC CDE0      int 224
206      00FE C3        ret
207
208
209
210      dseg
211
212      org      5ch

```

COPYRIGHT © 1981, 1982, 1983
 DIGITAL RESEARCH
 P. O. BOX 579
 PACIFIC GROVE, CA 93950

SER. # _____

```

213
214 005C          fcb      rb      33
215 007D 0000      r0       dw      0
216 007F 00        r3       db      0
217
218          org      100h
219 0100 4578616D706C sign_on_msg db      "Example TCOPY for IBM PC", cr, lf
220      652054434F50
221      5920666F7220
222      494240205043
223      000A
224 011A 525561647320 db      "Reads track image file and writes "
225      747261636820
226      696361676520
227      66596C652061
228      6E6420777269
229      74657320
230 013C 6974206F6E20 db      "it on track 0:"
231      747261636820
232      3024
233 014A 000A50757420 new_disk_msg db      cr,lf,"Put destination diskette in A:"
234      64657374696E
235      6174696F6E20
236      646973686574
237      746520696E20
238      413A
239 016A 000A          db      cr,lf
240 016C 537472696E65          db      "Strike any key when ready :"
241      20616E792068
242      657920776865
243      6E2072656164
244      792024
245 0187 000A57726974 continue_msg db      cr,lf,"Write another Track 0 (Y/N) ? :"
246      6520616E6F74
247      686572205472
248      616368203020
249      28592F4E2920
250      3F2024
251
252 01A8 000A24          crlf_msg db      cr,lf,"$"
253
254
255 01AB 54434F505920 drive_msg db      "TCOPY runs only on drive A: $"
256      72756E73206F
257      6E6C79206F6E
258      206472697665
259      20413A24
260 01C1 476976652066 open_msg db      "Give file name containing track 0 "
261      696C65206E61
262      6D6520636F6E
263      7461696E696E
264      672074726163
265      68203020

```

COPYRIGHT © 1981, 1982, 1983
 DIGITAL RESEARCH
 P. O. BOX 579
 PACIFIC GROVE, CA 93950

SER. # _____

```
266
267 01E9 69506167652C      db      "image, after TCDPY command!"
268      206166740572
269      2054434F5059
270      20636F6D6D61
271      6E6424
272 0204 46696C652069      read_msg  db      "File is not long enough!"
273      73206E6F7420
274      6C6F6E672065
275      6E6F75676824
276 021C 4572726F7220      write_msg db      "Error writing on track 0!"
277      77726974696E
278      67206F6E2074
279      725163682030
280      24
281
282 0235      track0_buffer  rb      1000H      ;4K tracks
283
284 1235      bpb            rb      5          ;direct Bios Parameter Block
285
286 123A 0000      current_dma  dw      0
287 123C 0000      current_sector dw      0
288
289
290 END OF ASSEMBLY.  NUMBER OF ERRORS:  0.  USE FACTOR:  3%
```

COPYRIGHT © 1981, 1982, 1983
DIGITAL RESEARCH
P. O. BOX 579
PACIFIC GROVE, CA 93950

SER. # _____

BIOS	00C1 L	147	151	155	159	164	167	168#
BPA	1235 V	169	284#					
BPBCX	0001 N	50#	171					
BPBDX	0003 N	51#	172					
BPBFUNC	0000 N	49#	170					
CONTINUEMSG	0187 V	181	245#					
CR	000D N	57#	219	233	239	245	252	
CREAD	0001 N	32#	118	183				
CRLF	00F7 L	192	202#					
CRLFMSG	01A8 V	203	252#					
CS	SREG V							
CURRENTDMA	123A V	86	93	104	130	137	158	286#
CURRENISECTOR	123C V	129	130	163	287#			
CWRITEBUF	0009 N	33#	66	114	180	194	204	
DONE	00F0 L	187	197#					
DRIVMSG	01AB V	73	255#					
DRIVEOK	0015 L	72	76#					
DRVGET	0019 N	42#	69					
DS	SREG V	154						
ERROR	00E7 L	74	83	102	190#			
ES	SREG V							
FCB	005C V	78	96	214#				
FILEOK	0026 L	81	85#					
FILEREAD	0035 L	90#	106					
FOPEN	000F N	36#	77					
FREADRAND	0021 N	37#	95					
FSETDMA	001A N	38#	92					
FSETDMASEG	0033 N	39#						
LF	000A N	56#	219	233	239	245	252	
NEWDISKMSG	014A V	115	233#					
NEXTDISKETTE	005C L	112#	188					
NEXTSECTOR	007F L	132#	140					
OPENMSG	01C7 V	82	260#					
RO	007D V	87	105	215#				
R3	007F V	216#						
READMSG	0204 V	101	272#					
READOK	0050 L	100	103#					
RECORDSTOREAD	0020 N	60#	88					
SDIRBIOS	0032 N	46#	173					
SELECTDISK	0099 L	125	143#					
SETDMAOFF	00AD L	134	156#					
SETDMASEG	00A7 L	127	152#					
SEI SECTOR	00B5 L	135	161#					
SETTRACK	00A1 L	126	148#					
SIGNONMSG	0100 V	67	219#					
SS	SREG V							
STERNCPM	0000 N	45#	198					
TRACKOBUFER	0235 V	86	130	282#				
TRACKOK	0003 L	141	179#					
WRITEMSG	021C V	276#						
WRITESECTOR	00BD L	136	165#					

COPYRIGHT © 1981, 1982, 1983
 DIGITAL RESEARCH
 P. O. BOX 579
 PACIFIC GROVE, CA 93950
 SER. # _____

