

```

0002 ;
0003 ; MODIFIED BY MJV 20 OCTOBER 1984 TO ALLOW FORMATTING
0004 ; SYSTEM TRACKS ONLY.
0005 ; MODIFIED 4 JUL 2014 TO CORRECT ERROR IN DOUBLE FORMATTING
0006 ;
0007 ; TO ASSEMBLE: ASMB FORMATX.AAA HEX DATE=070414 XREF WIDTH=132 PAGE=80
0008 ; TO LOAD: LOAD FORMATX.HEX
0009 ;
0010 ; THIS IS THE DISKETTE FORMATTER PROGRAM FOR THE TRIPLE DENSE BIOS SYSTEM.
0011 ; IT CAN FORMAT SINGLE DENSITY (S) IBM FORMAT, DOUBLE DENSITY (D) 26 256
0012 ; BYTE SECTORS PER TRACK, AND MAXIMUM DENSITY (X) 9 1024 BYTE SECTORS PER
0013 ; TRACK. DENSITY FLAG X=080H S=0FFH D=000H
(000D) 0014 CR EQU 0DH ; CARRIAGE RETURN DEFINITION FOR TEXT STRINGS
(000A) 0015 LF EQU 0AH ; LINE FEED DEFINITION FOR TEXT STRINGS
(001B) 0016 ESC EQU 01BH ; ESCAPE CHARACTER TO TERMINATE FORMAT
0017 ;
(0063) 0018 SELECT EQU 063H ; DRIVE, DENSITY, AND SIDE SELECT PORT
(0064) 0019 STATUS EQU 064H ; CONTROLLER STATUS PORT
(0065) 0020 TRACK EQU 065H ; CONTROLLER TRACK PORT
(0066) 0021 SECTOR EQU 066H ; CONTROLLER SECTOR PORT
(0067) 0022 DATA EQU 067H ; CONTROLLER DATA PORT
(0064) 0023 CMD EQU 064H ; CONTROLLER COMMAND PORT
0024 ;
(00FF) 0025 RWRTRY EQU 0FFH ; READ/WRITE RETRY COUNT FOR ERROR RECOVERY
(0005) 0026 SRTRY EQU 05H ; SEEK RETRY COUNT FOR ERROR RECOVERY
(FF05) 0027 RETRYC EQU RWRTRY*256+SRTRY ; READ/WRITE OPERATION RECOVERY COUNTS
0028 ;
(0005) 0029 BDOS EQU 0005H ; ADDRESS OF VECTOR TO CALL BDOS
(0000) 0030 WBOOTV EQU 0000H ; ADDRESS OF VECTOR TO WARM BOOT
0031 ;
(0000) 0032 STEPR EQU 00H ; HEAD STEP RATE BITS FOR RESTORE, SEEK,
0033 ; STEP IN, AND STEP OUT COMMANDS (3 MSEC)
(0008) 0034 RESTOR EQU 08H+STEPR ; CONTROLLER COMMAND TO RESTORE (HOME) HEAD
(0018) 0035 SKHDL EQU 018H+STEPR ; CONTROLLER COMMAND TO SEEK SPECIFIED TRACK
(005C) 0036 STEPI EQU 05CH+STEPR ; CONTROLLER COMMAND TO STEP HEAD IN 1 TRACK
(007C) 0037 STEPO EQU 07CH+STEPR ; CONTROLLER COMMAND TO STEP HEAD OUT 1 TRACK
(00D0) 0038 FORINT EQU 0D0H ; CONTROLLER COMMAND TO FORCE INTERRUPT
(0088) 0039 RDSECC EQU 088H ; CONTROLLER COMMAND TO READ SECTOR
(00F4) 0040 FORTRK EQU 0F4H ; CONTROLLER COMMAND TO FORMAT WRITE TRACK
(00FF) 0041 DESEL EQU 0FFH ; SELECT COMMAND TO DESELECT ALL DRIVES
0042 ;
(0091) 0043 STERM EQU 091H ; SEEK/STEP STATUS MASK TO CHECK NOT READY,
0044 ; SEEK ERROR, AND BUSY CONTROLLER STATUS BITS
(00F0) 0045 FORTKM EQU 0F0H ; FORMAT TRACK STATUS MASK TO CHECK NOT READY
0046 ; WRITE PROTECT, WRITE FAULT, AND RECORD NOT
0047 ; FOUND
(00FE) 0048 RDEM EQU 0FEH ; READ SECTOR STATUS MASK TO CHECK NOT READY,
0049 ; RECORD TYPE, RECORD NOT FOUND, CRC ERROR,
0050 ; LOST DATA, AND DRQ
(002F) 0051 WCLDBM EQU 02FH ; MASK TO CHECK FOR WRITE FAULT, CRC ERROR,
0052 ; LOST DATA, DRQ, OR BUSY
(007F) 0053 NRDYM EQU 07FH ; MASK TO CHECK ALL STATUS BITS EXCEPT NOT
0054 ; READY
(FF80) 0055 DNRDYM EQU NOT(NRDYM) ; MASK TO CHECK DRIVE NOT READY
(0010) 0056 RNFM EQU 010H ; MASK TO CHECK RECORD NOT FOUND STATUS BIT
(0010) 0057 SEEKEM EQU 010H ; MASK TO CHECK SEEK ERROR STATUS BIT
(0020) 0058 HDNTLM EQU 020H ; MASK TO CHECK HEAD NOT LOADED STATUS BIT
(0001) 0059 BUSYM EQU 001H ; MASK TO CHECK CONTROLLER BUSY STATUS BIT
0060 ;
(0020) 0061 STEPDM EQU 020H ; MASK TO CHANGE STEP DIRECTION IN STEP
0062 ; COMMAND
(0004) 0063 EBITM EQU 004H ; MASK TO SET E BIT IN READ AND WRITE COMMANDS
0064 ; TO DELAY 15 MSEC IF HEAD NOT LOADED
(0010) 0065 SIDEM EQU 010H ; MASK TO CHANGE SIDE IN DRIVE SELECT COMMAND
(007F) 0066 EWAITM EQU 07FH ; MASK TO ENABLE WAIT IN DRIVE SELECT COMMAND
(FF80) 0067 DWAITM EQU NOT(EWAITM) ; MASK TO DISABLE WAIT IN DRIVE SELECT COMMAND
(000F) 0068 DRIVM EQU 00FH ; MASK TO OBTAIN DRIVE # FROM SELECT STATUS
0069 ;
(0004) 0070 SIDSLB EQU 004H ; BIT NUMBER OF SIDE SELECT BIT IN DRIVE SELECT
0071 ; COMMAND
(0006) 0072 DENSB EQU 006H ; BIT NUMBER OF DENSITY SELECT IN DRIVE SELECT
0073 ; COMMAND
(0001) 0074 F1B EQU 001H ; BIT NUMBER OF F1 BIT IN CONTROLLER COMMAND
0075 ; TO CHECK THAT THE SELECTED SIDE FLAG (F2)
0076 ; AND THE SIDE (HEAD) ADDRESS BYTE AGREE
0077 ;
(0100) 0078 ORG 0100H
    
```

```

0079 ;
0100 C35701 R 0080 JP FORMAT ; GO TO START OF FORMAT PROGRAM
0103 ODOA2020 0081 MCL: DB CR,LF, ' '
012D 20202020 0082 DB ' $'
0157 313E0B 0083 FORMAT: LD SP,STACK ; INITIALIZE STACK POINTER
015A 0E0C 0084 LD C,00CH ; GET CODE TO REQUEST CPM VERSION NUMBER
015C CD0500 0085 CALL CALL BDOS ; CALL BDOS TO GET VERSION NUMBER
015F 22090B 0086 LD (CPMVER),HL ; SAVE CPM VERSION
0162 CDC80A 0087 CALL LKBIOS ; INITIALIZE LINKAGE TO BIOS
0165 210301 0088 LD HL,MCL ; GET ADDRESS OF CR, LF, AND 81 SPACES
0168 CDC20A 0089 CALL MSGCON ; SEND CR, LF, AND 81 SPACES TO CONSOLE
016B CDE02 0090 RDENCU: CALL GTNBRT ; GET NUMBER OF TRACKS TO FORMAT
016E CDDF01 0091 CALL GTFMAT ; GET AND SET UP OPERATOR DESIRED FORMAT
0092 ; DENSITY I. E. S, D, OR X
0171 32080B 0093 LD (DENSFL),A ; SAVE SELECTED FORMAT DENSITY
0174 CDBC01 0094 CALL GTDRIV ; SET UP DRIVE TO FORMAT
0177 CD3802 0095 FORMT1: CALL OFORCU ; DISPLAY 'TO FORMAT XXXXXX ON DRIVE X
0096 ; TYPE CR, ^C TO END, ESC FOR NEW OPTIONS.'
017A CDFC09 0097 CALL GTOPRP ; GET OPERATOR RESPONSE TO ABOVE CUE
017D FE03 0098 CP 'C'-040H ; CHECK IF ^C ENTERED
017F CAB901 R 0099 JP Z,REBOOT ; GO JUMP TO BOOT IF ^C
0182 FE0D 0100 CP CR ; CHECK IF CARRIAGE RETURN ENTERED
0184 C26B01 R 0101 JP NZ,RDENCU ; GO REDISPLAY DENSITY CUE IF NOT CR
0187 215509 0102 LD HL,TKNHDR ; GET ADDRESS OF TRACK # HEADER DISPLAY
018A CDC20A 0103 CALL MSGCON ; DISPLAY ' 0 | 1 | ETC. '
018D CD7602 0104 CALL FORDSK ; FORMAT DISKETTE ON SPECIFIED DRIVE
0190 313E0B 0105 REDRRN: LD SP,STACK ; RESET STACK POINTER TO START OF STACK
0193 212C08 0106 LD HL,RERNCU ; GET ADDRESS OF RERUN CUE
0196 CDC20A 0107 CALL MSGCON ; DISPLAY 'TO RUN AGAIN TYPE CR, ESC FOR
0108 ; OPTIONS, ^C TO ABORT'
0199 CDFC09 0109 CALL GTOPRP ; GET OPERATOR RESPONSE TO ABOVE CUE
019C FE1B 0110 CP ESC ; CHECK IF ESC ENTERED
019E CA6B01 R 0111 JP Z,RDENCU ; GO DISPLAY DENSITY CUE IF ESC
01A1 FE0D 0112 CP CR ; CHECK IF CARRIAGE RETURN ENTERED
01A3 CA7701 R 0113 JP Z,FORMT1 ; GO DISPLAY PREVIOUSLY SELECTED DENSITY
0114 ; IF CARRIAGE RETURN ENTERED
01A6 FE03 0115 CP 'C'-040H ; CHECK IF ^C ENTERED
01A8 CAB901 R 0116 JP Z,REBOOT ; GO REBOOT CPM IF ^C
01AB C39001 R 0117 JP REDRRN ; GO REDISPLAY RERUN CUE IF NOT CR, ESC, OR ^C
0118 ;
0119 ; THIS SUBROUTINE CHECK IF THE OPERATOR HAS TYPED ESC. IF YES, IT GOES TO
0120 ; RESET THE STACK POINTER AND DISPLAY THE RERUN CUE.
0121 ;
01AE 1EFF 0122 CKESC: LD E,0FFH ; SET FLAG FOR DIRECT CONSOLE INPUT
01B0 CDF209 0123 CALL DCONIO ; GET CONSOLE CHARACTER/STATUS
01B3 FE1B 0124 CP ESC ; CHECK IF ESC ENTERED
01B5 CA9001 R 0125 JP Z,REDRN ; GO RESET STACK AND DISPLAY RERUN CUE IF ESC
01B8 C9 0126 RET
0127 ;
0128 ; GO TO CP/M WARM BOOT ROUTINE TO RETURN TO CP/M
0129 ;
01B9 C30000 0130 REBOOT: JP WBOOTV ; GO TO CP/M WARM BOOT
0131 ;
0132 ; SUBROUTINE TO GET DRIVE WITH DISKETTE TO FORMAT
0133 ;
01BC 21EC07 0134 GTDRIV: LD HL,EDRVCU ; GET ADDRESS OF CUE TO ENTER DRIVE
01BF CDC20A 0135 CALL MSGCON ; DISPLAY ' ENTER DRIVE (a B c or d) -'
01C2 CDFC09 0136 CALL GTOPRP ; GET OPERATOR RESPONSE TO ABOVE CUE
01C5 FE1B 0137 CP ESC ; CHECK IF ESC ENTERED
01C7 CAB901 R 0138 JP Z,REBOOT ; GO REBOOT IF ESC
01CA FE0D 0139 CP CR ; CHECK IF CARRIAGE RETURN
01CC C2D101 R 0140 JP NZ,GTDRE ; GO CHECK DRIVE ENTERED IF NOT CR
01CF 3E42 0141 LD A,'B' ; DEFAULT TO DRIVE B IF CARRIAGE RETURN
01D1 321909 0142 GTDRE: LD (DDCUS),A ; SAVE DRIVE CHARACTER IN DISK ON DRIVE CUE
01D4 D641 0143 SUB 'A' ; CONVERT TO HEX DRIVE NUMBER (0,1,2, OR 3)
01D6 32FF0A 0144 LD (DRIVFT),A ; SAVE DRIVE NUMBER FOR FORMAT
01D9 FE04 0145 CP 004H ; CHECK IF DRIVE SELECTION VALID
01DB D2BC01 R 0146 JP NC,GTDRIV ; GO REDISPLAY DRIVE SELECTION CUE IF NOT
01DE C9 0147 RET
0148 ;
0149 ; THIS SUBROUTINE GETS THE FORMATTING DENSITY FROM THE OPERATOR AND SETS UP TO
0150 ; FORMAT WITH THE SELECTED DENSITY
0151 ;
01DF 210C08 0152 GTFMAT: LD HL,EDENCU ; GET ADDRESS OF CUE TO ENTER DENSITY
01E2 CDC20A 0153 CALL MSGCON ; DISPLAY 'ENTER DENSITY (s d or X) -'
01E5 CDFC09 0154 CALL GTOPRP ; GET OPERATOR RESPONSE TO ABOVE CUE
01E8 FE1B 0155 CP ESC ; CHECK IF ESC ENTERED
01EA CAB901 R 0156 JP Z,REBOOT ; GO REBOOT IF ESC
01ED FE0D 0157 CP CR ; CHECK IF CARRIAGE RETURN ENTERED
01EF C2F401 R 0158 JP NZ,GTFMXX ; GO CHECK DENSITY ENTERED IF NOT CR

```

```

01F2 3E58      0159      LD      A,'X'      ; DEFAULT TO X (MAXIMUM DENSITY, 9 1024 BYTE
                   0160      ; SECTORS PER TRACK)
01F4 FE58      0161      GTFCMX: CP      'X'      ; CHECK IF X (MAXIMUM DENSITY)
01F6 0680      0162      LD      B,080H    ; SET MAXIMUM DENSITY INDICATOR
01F8 CA0902    R 0163      JP      Z,GTFFST  ; GO SET MAXIMUM DENSITY IN DENSITY FLAG, X
01FB FE53      0164      CP      'S'      ; CHECK IF S (SINGLE DENSITY) ENTERED
01FD 06FF      0165      LD      B,0FFH    ; SET SINGLE DENSITY INDICATOR
01FF CA0902    R 0166      JP      Z,GTFFST  ; GO SET SINGLE DENSITY IN DENSITY FLAG, S
0202 FE44      0167      CP      'D'      ; CHECK IF D (DOUBLE DENSITY, 26 256 BYTE
                   0168      ; SECTORS PER TRACK) ENTERED
0204 0600      0169      LD      B,000H    ; SET DOUBLE DENSITY INDICATOR
0206 C2DF01    R 0170      JP      NZ,GTFFMAT ; GO REDISPLAY DENSITY CUE IF NOT X, S, OR D
0209 78        0171      GTFFST: LD     A,B      ; SELECTED DENSITY FLAG TO A
020A 32080B    0172      LD      (DENSFL),A ; SAVE SELECTED DENSITY FLAG
020D C9        0173      RET
                   0174      ;
                   0175      ; THIS SUBROUTINE GETS THE NUMBER OF TRACKS TO FORMAT (SYSTEM OR ALL)
020E 3E4D      0176      GTNBRT: LD     A,04DH   ; LOAD DEFAULT NUMBER OF TRACKS TO FORMAT
0210 32000B    0177      LD      (NUMTRK),A ; DEFAULT IS 77 TRACKS
0213 21CD07    0178      LD      HL,TRKCUE ; GET ADDRESS OF CUE
0216 CDC20A    0179      CALL   MSGCON     ; DISPLAY 'ENTER TRACKS (s or A)
0219 CDFC09    0180      CALL   GTOPRP     ; GET OPERATOR RESPONSE
021C FE1B      0181      CP      ESC       ; ESC ENTERED ?
021E CAB901    R 0182      JP      Z,REBOOT  ; YES
0221 FE0D      0183      CP      CR       ; CR ENTERED ?
0223 C22802    R 0184      JP      NZ,GTNTRK ; NO, FIND WHAT WAS ENTERED
0226 3E41      0185      LD      A,'A'     ; YES, ENTER DEFAULT
0228 FE41      0186      GTNTRK: CP     'A'   ; ALL TRACKS ?
022A CA3702    R 0187      JP      Z,GTTRK2  ; YES
022D FE53      0188      CP      'S'      ; SYSTEM ONLY
022F 3E02      0189      LD      A,02H    ; 2 TRACKS
0231 32000B    0190      LD      (NUMTRK),A ; SYSTEM ONLY HAS 2 TRACKS
0234 C20E02    R 0191      JP      NZ,GTNBRT ; NOT S OR A, TRY AGAIN
0237 C9        0192      GTTRK2: RET
                   0193      ;
                   0194      ; THIS SUBROUTINE DISPLAYS THE CUE ' TO FORMAT XXXXXXXX DENSITY ON DRIVE X
                   0195      ; TYPE CR, ^C TO END, ESC FOR NEW OPTIONS.'
                   0196      ;
0238 21C108    0197      OFORCU: LD     HL,TFMTCU ; GET ADDRESS OF 'TO FORMAT' MESSAGE
023B CDC20A    0198      CALL   MSGCON     ; DISPLAY 'TO FORMAT'
023E 21D108    0199      LD      HL,SYSCU  ; GET ADDRESS OF 'SYSTEM FILES'
0241 3A000B    0200      LD      A,(NUMTRK) ; NUMBER OF TRACKS TO FORMAT
0244 FE02      0201      CP      02H     ; CHECK IF ALL TRACKS
0246 CA4C02    R 0202      JP      Z,ONUMTR  ; GO DISPLAY 'SINGLE TRACKS...'
0249 21E108    0203      LD      HL,FILECU ; GET ADDRESS OF 'ALL TRACKS '
024C CDC20A    0204      ONUMTR: CALL  MSGCON   ; DISPLAY NUMBER OF TRACKS TO FORMAT
024F 3A080B    0205      LD      A,(DENSFL) ; GET SELECTED DENSITY FLAG
0252 FEFF      0206      CP      0FFH    ; CHECK IF SINGLE DENSITY
0254 21EE08    0207      LD      HL,SINGCU ; GET ADDRESS OF 'SINGLE'
0257 CA6502    R 0208      JP      Z,OFORD   ; GO DISPLAY 'SINGLE' IF SINGLE SELECTED
025A FE80      0209      CP      080H    ; CHECK IF MAXIMUM DENSITY SELECTED
025C 21FE08    0210      LD      HL,MAXCU  ; GET ADDRESS OF 'MAXIMUM'
025F CA6502    R 0211      JP      Z,OFORD   ; GO DISPLAY 'MAXIMUM' IF MAXIMUM SELECTED
0262 21F608    0212      LD      HL,DOUBCU ; GET ADDRESS IF 'DOUBLE'
0265 CDC20A    0213      OFORD: CALL  MSGCON   ; DISPLAY TEXT OF SELECTED DENSITY
0268 210709    0214      LD      HL,DONDCU ; GET ADDRESS OF 'DENSITY ON DRIVE ETC.'
026B CDC20A    0215      CALL   MSGCON     ; DISPLAY 'DENSITY ON DRIVE X
                   0216      ; TYPE CR, ^C TO END, ESC FOR NEW OPTIONS.'
026E C9        0217      RET
                   0218      ;
                   0219      ; THIS SUBROUTINE SENDS CARRIAGE RETURN AND LINE FEED TO THE CONSOLE
                   0220      ;
026F 21CA07    0221      OTCRLF: LD     HL,CRLFMSG ; GET ADDRESS OF THE CR, LF MESSAGE
0272 CDC20A    0222      CALL   MSGCON     ; SEND CARRIAGE RETURN AND LINE FEED
                   0223      ; TO THE CONSOLE
0275 C9        0224      RET
                   0225      ;
                   0226      ; THIS SUBROUTINE FORMATS THE DISKETTE ON THE SELECTED DRIVE IN THE SELECTED
                   0227      ; FORMAT.
                   0228      ;
0276 CD6F02    0229      FORDSK: CALL  OTCRLF  ; SEND CARRIAGE RETURN, LINE FEED TO CONSOLE
0279 CDAE01    0230      CALL   CKESC     ; CHECK IF ESC ENTERED, DISPLAY 'TO RUN AGAIN
                   0231      ; TYPE.....' IF ENTERED, DO NOT RETURN
027C CD1406    0232      CALL   BSDTKB    ; BUILD SINGLE DENSITY TRACK FORMAT BUFFER
027F CD8F06    0233      CALL   BDDTKB    ; BUILD DOUBLE DENSITY OR MAX DENSITY BUFFER
0282 0E00      0234      LD      C,000H   ; INITIALIZE CYLINDER (TRACK) TO 0
0284 0600      0235      LD      B,000H   ; INITIALIZE HEAD (SIDE) TO 0
0286 CD7B06    0236      CALL   STIDSD    ; STORE ID'S (HEAD, CYLINDER, AND SECTOR)
                   0237      ; IN SINGLE DENSITY TRACK FORMAT BUFFER
0289 0E00      0238      LD      C,000H   ; INITIALIZE CURRENT CYLINDER (TRACK) TO 0

```

028B	79	0239	LD	A,C	; SET UP TO SAVE CURRENT TRACK BEING FORMATTED
028C	32010B	0240	LD	(TKNUMF),A	; SAVE CYLINDER (TRACK) BEING FORMATTED
028F	C5	0241	PUSH	BC	; SAVE CYLINDER AND HEAD ON THE STACK
0290	CD6405	0242	CALL	CKDS	; CHECK IF DRIVE IS DOUBLE SIDED
0293	DB63	0243	IN	A,(SELECT)	; GET DRIVE SELECT STATUS
0295	E610	0244	AND	SIDEM	; ISOLATE SIDE 2 BIT (BIT IS SET UP BY CKDS
		0245			; TO INDICATE THAT THE SELECTED DRIVE IS DOUBLE
		0246			; SIDED)
0297	3EFF	0247	LD	A,0FFH	; SET DOUBLE SIDED FLAG
0299	2801	0248	JR	Z,STSDFG	; GO STORE FLAG IF DOUBLE SIDED DISKETTE
029B	2F	0249	CPL		; SET FLAG TO SINGLE SIDED STATE
029C	32050B	0250	STSDFG: LD	(SSDSFG),A	; SAVE SINGLE/DOUBLE SIDED FLAG
029F	DB63	0251	IN	A,(SELECT)	; GET CURRENT SELECT STATUS
02A1	F610	0252	OR	A,SIDEM	; SELECT SIDE 1 IN CASE
02A3	D363	0253	OUT	(SELECT),A	; SIDE 2 WAS SELECTED
02A5	3E08	0254	LD	A,RESTOR	; GET RESTORE COMMAND
02A7	CD1605	0255	CALL	SCOMME	; SEND RESTORE COMMAND
02AA	3E5C	0256	LD	A,STEPI	; GET STEP IN, HEAD LOAD
02AC	320B0B	0257	LD	(LSSTEP),A	; SET LAST STEP COMMAND TO STEP IN
02AF	213E0F	0258	LD	HL,SDBUFF	; GET ADDRESS OF SINGLE DENSITY FORMAT BUFFER
02B2	22FD0A	0259	LD	(ACTBFA),HL	; SET AS ACTIVE BUFFER ADDRESS FOR STIORG
02B5	22060B	0260	LD	(TKFBA),HL	; INITIALIZE CURRENT TRACK FORMAT BUFFER ADDR
02B8	21F009	0261	LD	HL,ATSKMG	; GET ADDRESS OF '*' MESSAGE
02BB	CDC20A	0262	CALL	MSGCON	; SEND '*' TO CONSOLE TO INDICATE NEXT TRACK
		0263			; IS BEING FORMATTED
02BE	CDAE01	0264	CALL	CKESC	; GO CHECK IF ESC ENTERED, IF ENTERED, GO
		0265			; REDISPLAY CUES, DO NOT RETURN
02C1	CD5504	0266	CALL	WRITK	; WRITE SPECIFIED TRACK
02C4	C20804	0267	JP	NZ,ERRWTK	; IF ERROR, GO DISPLAY ERROR WRITING TRACK
		0268			; AND RERUN IF ERROR
02C7	CD7703	0269	CALL	READV	; READ EACH SECTOR ON TRACK JUST FORMATTED
02CA	C22004	0270	JP	NZ,ERRRD	; GO DISPLAY READ ERROR AND RERUN IF ERROR
02CD	3A050B	0271	LD	A,(SSDSFG)	; GET SINGLE/DOUBLE SIDED FLAG
02D0	B7	0272	OR	A	; CHECK IF DOUBLE SIDED
02D1	2828	0273	JR	Z,FORNXT	; GO INCREMENT TO NEXT CYLINDER (TRACK) IF
		0274			; SINGLE SIDED
02D3	C1	0275	POP	BC	; GET CYLINDER (TRACK) AND HEAD
02D4	C5	0276	PUSH	BC	; SAVE TRACK AND HEAD AGAIN ON THE STACK
02D5	0601	0277	LD	B,001H	; SET HEAD (SIDE) TO 1 FOR FORMATTING
		0278			; OTHER SIDE
02D7	CD7B06	0279	CALL	STIDSD	; STORE ID BYTES (CYLINDER, HEAD, AND SECTOR)
		0280			; IN SINGLE DENSITY FORMAT BUFFER
02DA	CD1607	0281	CALL	STIDDD	; STORE ID BYTES IN DOUBLE DENSITY FORMAT BUFFR
02DD	DB63	0282	IN	A,(SELECT)	; GET DRIVE SELECT STATUS
02DF	EE10	0283	XOR	SIDEM	; CHANGE SIDE SELECT TO SELECT OTHER SIDE
02E1	D363	0284	OUT	(SELECT),A	; SELECT OTHER SIDE FOR FORMATTING
02E3	2A060B	0285	LD	HL,(TKFBA),A	; GET ADDRESS OF TRACK FORMAT BUFFER
02E6	22FD0A	0286	LD	(ACTBFA),HL	; SET AS ACTIVE BUFFER ADDRESS FOR STIOR
02E9	CD5504	0287	CALL	WRITK	; FORMAT SPECIFIED TRACK
02EC	C20804	0288	JP	NZ,ERRWTK	; GO DISPLAY FORMAT ERROR AND RESTART IF ERROR
02EF	CD7703	0289	CALL	READV	; READ VERIFY ALL SECTORS OF TRACK JUST WRITTEN
02F2	C22004	0290	JP	NZ,ERRRD	; GO DISPLAY READ ERROR AND RESTART IF ERROR
02F5	DB63	0291	IN	A,(SELECT)	; GET DRIVE SELECT STATUS
02F7	EE10	0292	XOR	SIDEM	; CHANGE STATE OF SIDE BIT TO SELECT OTHER SIDE
02F9	D363	0293	OUT	(SELECT),A	; SELECT SIDE 0
02FB	C1	0294	FORNXT: POP	BC	; GET CYLINDER (TRACK) AND HEAD
02FC	0C	0295	FORNXL: INC	C	; INCREMENT TO NEXT TRACK
02FD	79	0296	LD	A,C	; NEXT TRACK # TO FORMAT TO A
02FE	32010B	0297	LD	(TKNUMF),A	; SET CURRENT TRACK # BEING FORMATTED
0301	3A000B	0298	LD	A,(NUMTRK)	; GET NUMBER OF TRACKS TO FORMAT
0304	B9	0299	CP	C	; CHECK IF 2 OR 77 CYLINDERS ARE FORMATTED
0305	CA7103	R 0300	JP	Z,EFMT	; GO RESTORE TO TRACK 0 AND RETURN IF FORMATTED
0308	0600	0301	LD	B,000H	; SET HEAD (SIDE) TO 0
030A	CD7B06	0302	CALL	STIDSD	; STORE ID BYTES IN SINGLE DENSITY FORMAT BUFFR
030D	CD1607	0303	CALL	STIDDD	; STORE ID BYTES IN DOUBLE DENSITY FORMAT BUFFR
0310	C5	0304	PUSH	BC	; SAVE CYLINDER (TRACK) AND HEAD (SIDE)
0311	CDAE01	0305	CALL	CKESC	; CHECK IF ESC ENTERED, GO REDISPLAY SELECTION
		0306			; CUES IF ENTERED, DO NOT RETURN
0314	213E0F	0307	LD	HL,SDBUFF	; GET ADDRESS OF SINGLE DENSITY TRACK FORMAT
		0308			; BUFFER
0317	22FD0A	0309	LD	(ACTBFA),HL	; SET AS ACTIVE BUFFER ADDRESS FOR STIORG
031A	22060B	0310	LD	(TKFBA),HL	; SAVE CURRENT TRACK FORMAT BUFFER ADDRESS
031D	CD4704	0311	CALL	GETDEN	; GET SELECTED FORMAT/DENSITY FLAG
0320	2809	0312	JR	Z,FORDA	; GO DISPLAY '*' AND FORMAT IF SINGLE DENSITY
0322	213E27	0313	LD	HL,DBUFF	; GET ADDRESS OF DOUBLE DENSITY TRACK FORMAT
		0314			; BUFFER
0325	22FD0A	0315	LD	(ACTBFA),HL	; SET AS ACTIVE BUFFER ADDRESS FOR STIORG
0328	22060B	0316	LD	(TKFBA),HL	; SAVE CURRENT TRACK FORMAT BUFFER ADDRESS
032B	21F009	0317	FORDA: LD	HL,ATSKMG	; GET ADDRESS OF '*' MESSAGE
032E	CDC20A	0318	CALL	MSGCON	; DISPLAY '*' TO INDICATE FORMATTING OF NEXT TK

```

0331 CDAE01      0319      CALL      CKESC          ; CHECK IF ESC ENTERED, GO REDISPLAY SELECTION
                                0320          ; CUES IF ENTERED, DO NOT RETURN
0334 CD5504      0321      CALL      WRTTK         ; FORMAT SPECIFIED TRACK
0337 C20804      0322      JP        NZ,ERRWTK    ; GO DISPLAY TRACK ERROR AND RESTART IF ERROR
033A CD7703      0323      CALL      READV        ; READ VERIFY EACH SECTOR OF TRACK FORMATTED
033D C22004      0324      JP        NZ,ERRRD    ; GO DISPLAY READ ERROR AND RESTART IF ERROR
0340 3A050B      0325      LD        A,(SSDSFG)   ; GET SINGLE/DOUBLE SIDED DISKETTE FLAG
0343 B7           0326      OR        A           ; CHECK IF DOUBLE SIDED
0344 2828        0327      JR        Z,FORNT1    ; GO SET UP TO FORMAT NEXT TRACK IF NOT
0346 C1          0328      POP      BC          ; GET HEAD (SIDE) AND CYLINDER (TRACK)
0347 C5          0329      PUSH     BC          ; PUT HEAD AND CYLINDER BACK ON STACK
0348 0601        0330      LD        B,001H      ; SET HEAD (SIDE) TO 1 TO FORMAT OTHER SIDE
034A CD7B06      0331      CALL      STIDSD       ; STORE ID BYTES IN SINGLE DENSITY FORMAT BUFFR
034D CD1607      0332      CALL      STIDDD       ; STORE ID BYTES IN DOUBLE DENSITY FORMAT BUFFR
0350 0603        0333      IN        A,(SELECT)  ; GET DRIVE SELECT STATUS
0352 EE10        0334      XOR      SIDEM        ; TOGGLE SIDE BIT TO OTHER SIDE
0354 D363        0335      OUT      (SELECT),A   ; SELECT SIDE 1 FOR FORMATTING
0356 2A060B      0336      LD        HL,(TKFBA)   ; GET ADDRESS OF TRACK FORMAT BUFFER
0359 22FD0A      0337      LD        (ACTBFA),HL ; SET AS ACTIVE BUFFER ADDRESS FOR STIORG
035C CD5504      0338      CALL      WRTTK         ; FORMAT SPECIFIED TRACK
035F C20804      0339      JP        NZ,ERRWTK    ; GO DISPLAY TRACK ERROR AND RESTART IF ERROR
0362 CD7703      0340      CALL      READV        ; READ VERIFY EACH SECTOR OF TRACK FORMATTED
0365 C22004      0341      JP        NZ,ERRRD    ; GO DISPLAY READ ERROR AND RESTART IF ERROR
0368 DB63        0342      IN        A,(SELECT)  ; GET DRIVE SELECT STATUS
036A EE10        0343      XOR      SIDEM        ; TOGGLE SIDE BIT TO OTHER SIDE
036C D363        0344      OUT      (SELECT),A   ; SELECT SIDE 0 FOR FORMATTING
036E C1          0345      FORNT1: POP      BC          ; GET HEAD (SIDE) AND CYLINDER (TRACK)
036F 188B        0346      JR        FORNX1      ; GO FORMAT NEXT TRACK
0371 3E08        0347      EFMT:   LD        A,RESTOR ; GET RESTORE COMMAND
0373 CD1605      0348      CALL      SCOMME       ; SEND RESTORE COMMAND
0376 C9          0349      RET
                                0350      ;
                                0351      ; THIS SUBROUTINE READS EACH SECTOR OF THE TRACK JUST FORMATTED IN ORDER TO
                                0352      ; INSURE THE TRACK IS PROPERLY FORMATTED.
                                0353      ;
0377 CD4704      0354      READV: CALL      GETDEN    ; GET FORMAT/DENSITY FLAG
037A 284C        0355      JR        Z,RVSD      ; GO READ SINGLE DENSITY IF SINGLE DENSITY
037C B7          0356      OR        A           ; CHECK IF DOUBLE (26 256 BYTE SECTORS) DENSITY
037D 0601        0357      LD        B,001H      ; SET SECTOR TO READ TO 1
037F C2A603      R 0358      JP        NZ,R91024   ; GO READ MAXIMUM (9 1024 BYTE SECTORS) DENSITY
                                0359          ; IF NOT DOUBLE
                                0360      ;
                                0361      ; READ VERIFY 26 256 BYTE SECTORS (DOUBLE DENSITY)
                                0362      ;
0382 061A        0363      LD        B,01AH      ; SET NUMBER OF SECTORS TO READ TO 26
0384 21ED03      0364      LD        HL,SKWETB-1 ; GET ADDRESS OF SECTOR SKEW TABLE
0387 22EC03      0365      LD        (SSKTBA),HL ; INITIALIZE ADDRESS OF SECTOR NUMBER JUST READ
038A 2AEC03      0366      R2561: LD        HL,(SSKTBA) ; GET ADDRESS OF PREVIOUS SECTOR NUMBER
038D 23          0367      INC      HL           ; BUMP TO ADDRESS OF NEXT SECTOR NUMBER TO READ
038E 7E          0368      LD        A,(HL)      ; GET NUMBER OF NEXT SECTOR TO READ
038F 32020B     0369      LD        (CSECT),A   ; SAVE NEXT SECTOR NUMBER TO READ
0392 22EC03      0370      LD        (SSKTBA),HL ; MAKE THIS ADDRESS OF PREVIOUS SECTOR
0395 213E0B     0371      LD        HL,READBF   ; GET ADDRESS OF NEXT SECTOR TO READ
0398 22FD0A      0372      LD        (ACTBFA),HL ; SET AS ACTIVE BUFFER ADDRESS FOR STIORG
039B C5          0373      PUSH     BC          ; SAVE NUMBER OF SECTORS LEFT TO READ
039C CD7304      0374      CALL      RDSEC       ; READ SPECIFIED SECTOR
039F C1          0375      POP      BC          ; GET NUMBER OF SECTORS LEFT TO READ
03A0 C22004      R 0376      JP        NZ,ERRRD    ; GO DISPLAY READ ERROR AND RESTART IF ERROR
03A3 10E5        0377      DJNZ    R2561        ; GO READ NEXT SECTOR ON TRACK IF MORE
03A5 C9          0378      RET
                                0379      ;
                                0380      ; READ VERIFY 9 1024 BYTE SECTORS (MAXIMUM DENSITY)
                                0381      ;
03A6 78          0382      R91024: LD        A,B           ; NUMBER OF NEXT SECTOR TO READ TO A
03A7 32020B     0383      LD        (CSECT),A   ; SAVE CURRENT SECTOR BEING READ
03AA 213E0B     0384      LD        HL,READBF   ; GET ADDRESS OF SECTOR READ BUFFER
03AD 22FD0A      0385      LD        (ACTBFA),HL ; SET AS ACTIVE BUFFER ADDRESS FOR STIORG
03B0 C5          0386      PUSH     BC          ; SAVE CURRENT SECTOR NUMBER ON STACK
03B1 CD7304      0387      CALL      RDSEC       ; READ SPECIFIED SECTOR
03B4 C1          0388      POP      BC          ; GET CURRENT SECTOR NUMBER FROM STACK
03B5 C22004      R 0389      JP        NZ,ERRRD    ; GO DISPLAY READ ERROR AND RESTART IF ERROR
03B8 04          0390      INC      B           ; BUMP SECTOR NUMBER BY 2 SO EVERY OTHER
03B9 04          0391      INC      B           ; SECTOR IS READ ON ONE ROTATION
03BA 78          0392      LD        A,B         ; SECTOR NUMBER JUST READ TO A
03BB FE0B      0393      CP        00BH        ; CHECK IF LAST ODD NUMBERED SECTOR WAS READ
03BD 2805        0394      JR        Z,RD9LE     ; GO SET UP TO READ EVEN NUMBERED SECTORS, YES
03BF FE0A      0395      CP        00AH        ; CHECK IF LAST EVEN NUMBERED SECTOR WAS READ
03C1 C8          0396      RET        Z         ; RETURN IF ALL SECTORS VERIFIED
03C2 18E2        0397      JR        R91024     ; GO READ NEXT SECTOR
03C4 0602        0398      RD9LE: LD        B,002H ; SET UP TO READ FIRST OF EVEN NUMBERED SECTORS

```

```

03C6 18DE      0399      JR      R91024      ; GO READ NEXT SECTOR
                0400      ;
                0401      ; READ VERIFY 26 128 BYTE SECTORS (SINGLE DENSITY)
                0402      ;
03C8 061A      0403  RVSD:   LD      B,01AH      ; SET NUMBER OF SECTORS TO READ TO 26
03CA 21ED03     0404      LD      HL,SKWTB-1    ; GET ADDRESS OF SECTOR SKEW TABLE
03CD 22EC03     0405      LD      (SSKTBA),HL   ; INITIALIZE ADDRESS OF SECTOR NUMBER JUST READ
03D0 2AEC03     0406  RVSD1: LD      HL,(SSKTBA)  ; GET ADDRESS OF PREVIOUS SECTOR NUMBER
03D3 23         0407      INC     HL            ; BUMP TO ADDRESS OF NEXT SECTOR TO READ
03D4 7E         0408      LD      A,(HL)       ; GET NUMBER OF NEXT SECTOR TO READ
03D5 22EC03     0409      LD      (SSKTBA),HL   ; MAKE THIS ADDRESS OF PREVIOUS SECTOR
03D8 32020B     0410      LD      (CSECT),A    ; SAVE NUMBER OF NEXT SECTOR TO READ
03DB 213E0B     0411      LD      HL,READBF    ; GET ADDRESS OF SECTOR READ BUFFER
03DE 22FD0A     0412      LD      (ACTBFA),HL   ; SAVE SECTOR BUFFER ADDRESS
03E1 C5         0413      PUSH   BC            ; SAVE NUMBER OF SECTORS LEFT TO READ
03E2 CD7304     0414      CALL   RDSEC         ; READ SPECIFIED SECTOR
03E5 C1         0415      POP    BC            ; GET NUMBER OF SECTORS LEFT TO READ
03E6 C22004     R 0416     JP      NZ,ERRRD     ; GO DISPLAY READ ERROR AND RESTART IF ERROR
03E9 10E5       0417     DJNZ  RVSD1         ; GO READ NEXT SECTOR IN TRACK IF MORE
03EB C9         0418     RET
03EC ED03       0419  SSKTBA: DW      SKEWTB-1    ; ADDRESS OF SECTOR NUMBER JUST READ
                0420      ;
                0421      ; SECTOR SKEW TABLE FOR TRACK READ VERIFY
                0422      ;
03EE 0105090D  0423  SKEWTB: DB      001H,005H,009H,00DH,011H,015H,019H,003H,007H,00BH,00FH,013H
03FA 1702060A  0424      DB      017H,002H,006H,00AH,00EH,012H,016H,01AH,004H,008H,00CH,010H
0406 1418       0425      DB      014H,018H
                0426      ;
                0427      ; ERROR DURING FORMAT WRITE OF SPECIFIED TRACK
                0428      ;
0408 3A010B     0429  ERRWTK: LD      A,(TKNUMF)    ; GET TRACK NUMBER WHICH HAD ERROR
040B 217B08     0430      LD      HL,TKNMSG     ; GET ADDRESS OF TRACK # IN 'ERROR ON TRACK
                0431      ; NO. XX' MESSAGE
040E CDA80A     0432      CALL   BINDEC        ; CONVERT TRACK # TO DECIMAL ASCII AND STORE
                0433      ; IN ERROR MESSAGE
0411 218D08     0434      LD      HL,FEMSG     ; GET ADDRESS OF FORMAT ERROR MESSAGE
0414 CDC20A     0435      CALL   MSGCON        ; DISPLAY ' - FORMAT ERROR ON DISK'
0417 216408     0436      LD      HL,TKEMSG    ; GET ADDRESS OF "ERROR ON TRACK NO XX" MSG
041A CDC20A     0437      CALL   MSGCON        ; DISPLAY "ERROR ON TRACK NO. XX" ON CONSOLE
041D C39001     0438      JP      REDRRN       ; GO RESET STACK POINTER AND RERUN
                0439      ;
                0440      ; ERROR DURING READ VERIFY OF SPECIFIED TRACK
                0441      ;
0420 3A010B     0442  ERRRD:  LD      A,(TKNUMF)    ; GET NUMBER OF TRACK WHICH HAD ERROR
0423 217B08     0443      LD      HL,TKNMSG     ; GET ADDRESS OF TRACK # IN 'ERROR ON TRACK
                0444      ; NO. XX' MESSAGE
0426 CDA80A     0445      CALL   BINDEC        ; CONVERT TRACK # TO DECIMAL ASCII AND STORE
                0446      ; IN ERROR MESSAGE
0429 21A808     0447      LD      HL,REMSG     ; GET ADDRESS OF READ ERROR ERROR MESSAGE
042C CDC20A     0448      CALL   MSGCON        ; DISPLAY ' - READ ERROR ON DISK'
042F 3A020B     0449      LD      A,(CSECT)    ; GET SECTOR NUMBER OF READ ERROR
0432 218A08     0450      LD      HL,SCNMSG    ; GET ADDRESS OF SECTOR # IN 'SECTOR NO.XX' MSG
0435 CDA80A     0451      CALL   BINDEC        ; CONVERT SECTOR NUMBER TO DECIMAL AND STORE
                0452      ; IN 'SECTOR NO. XX' MESSAGE
0438 216408     0453      LD      HL,TKEMSG    ; GET ADDRESS OF 'ERROR ON TRACK NO. XX' MSG
043B CDC20A     0454      CALL   MSGCON        ; DISPLAY 'ERROR ON TRACK NO. XX' ON CONSOLE
043E 217E08     0455      LD      HL,SCEMSG    ; GET ADDRESS OF 'SECTOR NO. XX' MESSAGE
0441 CDC20A     0456      CALL   MSGCON        ; DISPLAY 'SECTOR NO. XX' ON CONSOLE
0444 C39001     0457      JP      REDRRN       ; GO RESET STACK POINTER AND RERUN
                0458      ;
                0459      ; THIS SUBROUTINE GETS THE SELECTED DISKETTE FORMAT/DENSITY FLAG
                0460      ;
0447 3A010B     0461  GETDEN: LD      A,(TKNUMF)    ; GET TRACK NUMBER BEING FORMATTED
044A B7         0462      OR     A             ; CHECK IF TRACK 0 (MUST FORMAT SINGLE DENSITY)
044B 3EFF       0463      LD      A,0FFH       ; SET SINGLE DENSITY INDICATOR IN A
044D 2803       0464      JR      Z,GETDNR    ; GO RETURN WITH SINGLE DENSITY INDICATOR IN A
                0465      ; AND ZERO FLAG SET IF TRACK 0
044F 3A080B     0466      LD      A,(DENSFL)   ; GET FORMAT/DENSITY FLAG
0452 FEFF       0467  GETDNR: CP      0FFH       ; CHECK IF SINGLE DENSITY
0454 C9         0468      RET                ; RETURN WITH ZERO FLAG SET IF SINGLE DENSITY
                0469      ;
                0470      ; THIS SUBROUTINE CALLS THE FORMAT WRITE SUBROUTINE TO WRITE THE SPECIFIED
                0471      ; TRACK AND PERFORMS ERROR RECOVERY/RETRY IN CASE OF AN ERROR
                0472      ;
0455 0105FF     0473  WRTTK: LD      BC,RETRYC    ; INITIALIZE RESEEK AND WRITE ERROR
                0474      ; RETRY COUNTS
0458 C5         0475  WRTTKR: PUSH   BC            ; SAVE CURRENT RESEEK AND WRITE ERROR
                0476      ; RETRY COUNTS
0459 3E00       0477      LD      A,000H       ; ZERO CONTROLLER
045B 320C0B     0478      LD      (ERRFG),A    ; ERROR STATUS FLAG

```

```

045E F3          0479          DI          ; DISABLE INTERRUPTS DURING TRACK FORMAT WRITE
045F CDF704     0480          CALL        FWTK          ; FORMAT WRITE NEXT TRACK
0462 FB         0481          EI          ; ENABLE INTERRUPTS AFTER TRACK FORMAT WRITE
0463 C1         0482          POP        BC          ; RESTORE ERROR RETRY COUNTS
0464 C8         0483          RET        Z          ; RETURN IF TRACK FORMAT WRITE OK
0465 7B         0484          LD        A,E          ; GET ERROR STATUS
0466 E62F       0485          AND        WCLDBM       ; CHECK IF WRITE FAULT, CRC ERROR, LOST DATA,
                                0486          ; DRQ, OR BUSY
0468 2829       0487          JR        Z,URECER     ; GO RETURN WITH ERROR STATUS IF NOT READY, OR
                                0488          ; WRITE PROTECT (ERROR IS UNRECOVERABLE)
046A 10EC       0489          DJNZ      WRITTKR     ; GO RETRY FORMAT IF ANOTHER WRITE IS OK
046C CD9904     0490          CALL      RESEK       ; REPOSITION HEAD WITH STEP IN/OUT COMMANDS
046F 20E7       0491          JR        NZ,WRITTKR  ; GO RETRY FORMAT WRITE IF ANOTHER SERIES OF
                                0492          ; WRITE ATTEMPTS IS OK
0471 1820       0493          JR        URECER      ; GO RETURN WITH ERROR STATUS IF ERROR APPEARS
                                0494          ; UNRECOVERABLE
                                0495          ;
0496          ; THIS SUBROUTINE CALLS THE SECTOR READ SUBROUTINE TO READ THE SPECIFIED
0497          ; SECTOR AND PERFORMS ERROR RECOVERY/RETRY IN CASE OF AN ERROR
                                0498          ;
0473 0105FF     0499 RDSEC: LD        BC,RETRYC ; INITIALIZE READ AND RESEK ERROR RETRY COUNTS
0476 C5         0500 RDSECR: PUSH    BC          ; SAVE RETRY COUNTS ON THE STACK
0477 3E00       0501          LD        A,000H      ; ZERO CONTROLLER
0479 320C0B     0502          LD        (ERRFG),A  ; ERROR STATUS FLAG
047C F3         0503          DI          ; DISABLE INTERRUPTS DURING SECTOR READ
047D CDC004     0504          CALL      RDSC        ; READ SPECIFIED SECTOR
0480 FB         0505          EI          ; ENABLE INTERRUPTS AFTER SECTOR READ
0481 C1         0506          POP        BC          ; GET RETRY COUNTS
0482 C8         0507          RET        Z          ; RETURN IF READ OK
0483 7B         0508          LD        A,E          ; GET READ OPERATION STATUS
0484 FE10       0509          CP        RNFM        ; CHECK IF RECORD NOT FOUND
0486 280B       0510          JR        Z,URECER     ; GO SET UNRECOVERABLE ERROR STATUS AND RETURN
                                0511          ; IF RECORD NOT FOUND
0488 E67F       0512          AND        NRDYM       ; CHECK IF NOT READY
048A 2807       0513          JR        Z,URECER     ; GO SET UNRECOVERABLE ERROR STATUS AND RETURN
                                0514          ; IF NOT READY
048C 10E8       0515          DJNZ      RDSECR      ; GO RETRY READ IF RETRY OK
048E CD9904     0516          CALL      RESEK       ; RESEK VIA STEP IN/OUT IF SEEK RETRY OK
0491 20E3       0517          JR        NZ,RDSECR   ; GO RETRY READ IF SEEK RETRY COUNT STILL OK
                                0518          ;
0519          ; SET UNRECOVERABLE ERROR STATUS AND RETURN
                                0520          ;
0493 AF         0521 URECER: XOR     A          ; SET
0494 3C         0522          INC     A          ; UNRECOVERABLE
0495 320C0B     0523          LD        (ERRFG),A  ; ERROR STATUS
0498 C9         0524          RET          ;
                                0525          ;
0526          ; THIS SUBROUTINE REPOSITIONS THE HEAD VIA STEP IN/OUT OR OUT/IN AND ALSO
0527          ; RESETS THE WRITE/READ RETRY COUNT
                                0528          ;
0499 3A010B     0529 RESEK: LD        A,(TKNUMF) ; GET CURRENT TRACK NUMBER
049C B7         0530          OR        A          ; CHECK IF TRACK 0
049D 3E5C       0531          LD        A,STPEI     ; SET STEP IN AS LAST COMMAND
049F 280C       0532          JR        Z,RSEEK1    ; GO SET UP TO SEND STEP IN IF 0
04A1 3A010B     0533          LD        A,(TKNUMF)  ; GET CURRENT TRACK NUMBER
04A4 FE4C       0534          CP        76          ; CHECK IF INNERMOST TRACK
04A6 3E7C       0535          LD        A,STPEO     ; SET STEP OUT AS LAST COMMAND
04A8 2803       0536          JR        Z,RSEEK1    ; GO SET UP TO SEND STEP OUT IF 76
04AA 3A0B0B     0537          LD        A,(LSSTEP)  ; GET LAST STEP COMMAND
04AD C5         0538 RSEEK1 PUSH    BC          ; SAVE CURRENT RETRY COUNTS
04AE CD1605     0539          CALL      SCOMME      ; SEND STEP COMMAND
04B1 7A         0540          LD        A,D          ; LAST STEP COMMAND BACK TO A
04B2 EE20       0541          XOR        STEPDM     ; CHANGE STEP COMMAND TO OPPOSITE STEP
                                0542          ; I. E. OUT TO IN OR IN TO OUT
04B4 CD1605     0543          CALL      SCOMME      ; SEND OPPOSITE STEP COMMAND
04B7 7A         0544          LD        A,D          ; LAST STEP COMMAND BACK TO A
04B8 320B0B     0545          LD        (LSSTEP),A  ; SAVE LAST STEP COMMAND SENT
04BB C1         0546          POP        BC          ; RESTORE RETRY COUNTS
04BC 0D         0547          DEC        C          ; DECREMENT RESEK RETRY COUNT
04BD 06FF       0548          LD        B,RWRTRY    ; RESET WRITE RETRY COUNT TO 10
04BF C9         0549          RET          ;
                                0550          ;
0551          ; READ SPECIFIED SECTOR
                                0552          ;
04C0 CDC605     0553 RDSC:  CALL      STIORG   ; SET UP BUFFER SIZE AND NUMBER OF BUFFERS
                                0554          ; BASED ON THE FORMAT/DENSITY AND SET WAIT
                                0555          ; ENABLE DRIVE SELECT STATUS
04C3 C0         0556          RET        NZ          ; RETURN IF SELECT UNSUCCESSFUL
04C4 3A020B     0557          LD        A,(CSECT)   ; GET SECTOR NUMBER TO READ
04C7 D366       0558          OUT      (SECTOR),A  ; OUTPUT SECTOR NUMBER TO READ TO SECTOR PORT

```

```

04C9 3E88      0559      LD      A,RDSECC      ; GET SECTOR READ COMMAND
04CB CDE004    0560      CALL   STFLS2        ; SET BIT 1 (F1) IN COMMAND IF SIDE 2 SELECTED
04CE CDE505    0561      CALL   SEHDNL        ; SET 15 MSEC DELAY BIT (2) IF HEAD NOT LOADED
04D1 D364      0562      OUT    (CMD),A       ; SEND SECTOR READ COMMAND TO CONTROLLER
04D3 0E67      0563      LD      C,DATA       ; GET READ DATA PORT ADDRESS
04D5 50        0564      LD      D,B          ; SET UP NUMBER OF BYTES IN SECTOR BUFFER
04D6 EDB2      0565      RDSCLP: INIR        ; READ ONE SECTOR BUFFER OR PARTIAL SECTOR
04D8 42        0566      LD      B,D          ; RESET NUMBER OF BYTES IN SECTOR BUFFER
04D9 1D        0567      DEC    E             ; CHECK IF MORE BUFFER READS MUST BE PERFORMED
                                0568      ; TO READ ENTIRE SECTOR E. G. 1024 BYTE SECTORS
                                0569      ; REQUIRE 4 BUFFER INPUTS TO READ ENTIRE SECTOR
04DA 20FA     0570      JR      NZ,RDSCLP    ; GO INPUT ANOTHER BUFFER IF MORE IN SECTOR
04DC 1EFE     0571      LD      E,RDEM       ; SET UP ERROR MASK TO CHECK NOT READY, RECORD
                                0572      ; NOT FOUND, CRC ERROR, LOST DATA, AND DRQ
04DE 184C     0573      JR      CKSTAT       ; GO GET AND CHECK READ OPERATION STATUS
                                0574      ;
                                0575      ; THIS SUBROUTINE CHECKS IF SIDE 1 IS SELECTED AND SETS BIT 1 (F1) IN THE
                                0576      ; COMMAND IN A IF SIDE 1 IS SELECTED
                                0577      ;
04E0 320D0B   0578      STFLS2: LD      (SAVCOM),A ; SAVE CURRENT COMMAND
04E3 DB63      0579      IN      A,(SELECT)    ; GET DRIVE SELECT STATUS
04E5 E610      0580      AND    SIDEM         ; CHECK IF SIDE 2 IS SELECTED
04E7 CAEE04    0581      R      JP      Z,SETE  ; GO SET BIT 1 IN COMMAND IF IT IS
04EA 3A0D0B   0582      LD      A,(SAVCOM)    ; RELOAD NEXT COMMAND IF SIDE 1 SELECTED
04ED C9        0583      RET
04EE 3A0D0B   0584      SETE:  LD      A,(SAVCOM) ; GET CURRENT COMMAND
04F1 CBCF      0585      SET    F1B,A         ; SET BIT 1 IN COMMAND (F1)
04F3 320D0B   0586      LD      (SAVCOM),A    ; SAVE CURRENT COMMAND
04F6 C9        0587      RET
                                0588      ;
                                0589      ; THIS SUBROUTINE PERFORMS THE FORMAT WRITE OF THE SPECIFIED TRACK
                                0590      ;
04F7 CD8F05   0591      FWTK:  CALL   WTINIT    ; POSITION HEAD TO NEXT TRACK TO FORMAT AND
                                0592      ; SET UP BUFFER SIZE, NUMBER OF BUFFERS, AND
                                0593      ; BUFFER ADDRESS BASED ON FORMAT AND DENSITY
04FA C0        0594      RET    NZ           ; RETURN IF UNSUCCESSFUL SEEK
04FB 3EF4      0595      LD      A,FORTRK     ; GET FORMAT TRACK COMMAND
04FD CDE004    0596      CALL   STFLS2        ; SET BIT 1 (F1) IN COMMAND IF SIDE 2 SELECTED
0500 CDE505    0597      CALL   SEHDNL        ; SET 15 MSEC DELAY BIT (E) IF HEAD NOT LOADED
0503 1628      0598      LD      D,028H       ; SET NUMBER OF BUFFERS REQUIRED TO WRITE TRACK
                                0599      ; TO 40
0505 D364      0600      OUT    (CMD),A       ; SEND WRITE TRACK COMMAND TO DISK CONTROLLER
0507 0E67      0601      LD      C,DATA       ; SET UP WRITE DATA PORT ADDRESS
0509 58        0602      LD      E,B          ; SAVE BUFFER SIZE IN E
050A EDB3      0603      FWTKLP: OTIR        ; SEND NEXT TRACK FORMAT BUFFER TO CONTROLLER
050C 15        0604      DEC    D             ; CHECK IF MORE BUFFERS LEFT TO SEND
050D 43        0605      LD      B,E          ; RESET NUMBER OF WORDS IN BUFFER
050E 20FA     0606      JR      NZ,FWTKLP    ; GO OUTPUT NEXT BUFFER IF MORE
0510 1EF0     0607      LD      E,FORTKM     ; SET UP STATUS MASK TO CHECK NOT READY, WRITE
                                0608      ; PROTECT, WRITE FAULT, AND RECORD NOT FOUND
0512 1818     0609      JR      CKSTAT       ; GO WAIT FOR COMPLETION, DISABLE WAIT, AND
                                0610      ; CHECK WRITE OPERATION STATUS
                                0611      ;
                                0612      ; SUBROUTINE ENTRY TO SEND COMMAND IN A AND USE A ZERO ERROR MASK TO CHECK
                                0613      ; OPERATION STATUS (IGNORE ERRORS)
                                0614      ;
0514 1E00     0615      SCOMIE: LD      E,000H ; SET OPERATION STATUS ERROR MASK TO 0
                                0616      ;
                                0617      ; SUBROUTINE ENTRY TO SEND COMMAND IN A AND USE PRESET ERROR MASK IN E TO
                                0618      ; CHECK OPERATION STATUS
                                0619      ;
0516 57        0620      SCOMME: LD      D,A   ; SAVE COMMAND TO SEND IN D
0517 DB64      0621      CMBSYL: IN      A,(STATUS) ; GET CONTROLLER STATUS
0519 E601      0622      AND    BUSYM        ; CHECK IF BUSY (COMMAND IN EXECUTION)
051B 20FA     0623      JR      NZ,CMBSYL    ; GO CHECK STATUS AGAIN IF BUSY
051D DB63      0624      IN      A,(SELECT)    ; GET DRIVE SELECT STATUS
051F F680      0625      OR     DWAITM       ; DISABLE WAIT
0521 D363      0626      OUT    (SELECT),A    ; SEND DISABLE WAIT DRIVE SELECT
0523 7A        0627      LD      A,D          ; GET COMMAND TO SEND TO CONTROLLER
0524 D364      0628      OUT    (CMD),A       ; SEND COMMAND TO CONTROLLER
0526 F5        0629      PUSH   AF            ; DELAY BEFORE CHECKING
0527 F1        0630      POP    AF            ;
                                0631      ; OPERATION STATUS
0528 F5        0631      PUSH   AF            ;
0529 F1        0632      POP    AF            ;
052A F5        0633      PUSH   AF            ;
052B F1        0634      POP    AF            ;
                                0635      ;
                                0636      ; THIS ENTRY POINT IS USED TO CHECK STATUS AT THE COMPLETION OF A CONTROLLER
                                0637      ; COMMAND.
                                0638      ;

```



```

052C DB64      0639 CKSTAT: IN      A,(STATUS)      ; GET CONTROLLER OPERATION STATUS
052E E601      0640 AND          BUSYM          ; CHECK IF BUSY (COMMAND IN EXECUTION)
0530 20FA      0641 JR           NZ,CKSTAT      ; GO CHECK STATUS AGAIN IF BUSY
0532 DB63      0642 IN          A,(SELECT)     ; GET DRIVE SELECT STATUS
0534 F680      0643 OR          DWAITM        ; DISABLE WAIT
0536 D363      0644 OUT         (SELECT),A     ; SEND DISABLE WAIT DRIVE SELECT
0538 DB64      0645 IN          A,(STATUS)     ; GET CONTROLLER OPERATION STATUS
053A A3        0646 AND          E             ; CHECK FOR ERRORS USING ERROR MASK
053B C8        0647 RET          Z             ; RETURN IF NO ERRORS
0648 ;
0649 ; THIS ENTRY POINT SETS CONTROLLER ERROR STATUS AND RETURNS WITH THE NON-ZERO
0650 ; FLAG SET
0651 ;
053C 5F        0652 OPERR: LD     E,A           ; SAVE OPERATION STATUS IN E
053D 3EFF      0653 LD     A,0FFH          ; GET OPERATION ERROR FLAG
053F 320C0B    0654 LD     (ERRFG),A       ; SET CONTROLLER ERROR STATUS
0542 B7        0655 OR     A             ; SET NON-ZERO FLAG (ERROR)
0543 C9        0656 RET
0657 ;
0658 ; THIS SUBROUTINE BUILDS AND SENDS THE DRIVE SELECT/DENSITY COMMAND.
0659 ; NOTE: SIDE 2 IS ALSO SELECTED.
0660 ;
0661 ;
0544 3AFF0A    0662 GTDSC2: LD   A,(DRIVFT)    ; GET DRIVE SELECTED BY THE OPERATOR
0547 CDF505    0663 CALL  DNTODS          ; CONVERT TO UNIT SELECT IN COMMAND
054A CBA7      0664 RES   SIDSLB,A        ; ZERO SIDE BIT TO SELECT SIDE 2 (INVERSE
                                ; LOGIC)
0665 ;
0666 ; THIS SUBROUTINE ENTRY POINT SELECTS THE DRIVE SPECIFIED IN A AT THE CORRECT
0667 ; DENSITY.
0668 ;
0669 ;
054C CBB7      0670 GTDSCM: RES  DENSEB,A     ; ZERO DENSITY BIT (DOUBLE DENSITY)
054E 5F        0671 LD     E,A           ; SAVE UNIT SELECT COMMAND IN E
054F CD4704    0672 CALL  GETDEN          ; GET THE FORMAT/DENSITY AND CHECK IF SINGLE
0552 7B        0673 LD     A,E           ; RESTORE UNIT SELECT COMMAND TO A
0553 2002      0674 JR     NZ,GTDDTK      ; GO CHECK TRACK NUMBER IF DOUBLE DENSITY
0555 CBF7      0675 SET   DENSEB,A        ; SET DENSITY BIT (SINGLE DENSITY)
0557 5F        0676 GTDDTK: LD   E,A           ; SAVE UNIT SELECT COMMAND IN E
0558 3A010B    0677 LD     A,(TKNUMF)     ; GET CURRENT TRACK NUMBER TO FORMAT
055B B7        0678 OR     A             ; CHECK IF TRACK 0 (SINGLE DENSITY)
055C 7B        0679 LD     A,E           ; RESTORE UNIT SELECT COMMAND TO A
055D 2002      0680 JR     NZ,GSDSEL      ; GO SEND DOUBLE DENSITY DRIVE SELECT IF NOT 0
055F CBF7      0681 SET   DENSEB,A        ; SET DENSITY BIT (SINGLE DENSITY)
0561 D363      0682 GSDSEL: OUT  (SELECT),A   ; SEND DRIVE SELECT/DENSITY COMMAND
0563 C9        0683 RET
0684 ;
0685 ; THIS SUBROUTINE CHECKS IF THE DRIVE ENTERED BY THE OPERATOR IS SELECTED. IF
0686 ; NOT, IT SELECTS THE DRIVE WITH THE SIDE BIT 0 TO SEE IF IT IS DOUBLE SIDED.
0687 ; IF IT IS, IT RETURNS WITH THE SIDE BIT 0. OTHERWISE, IT SETS THE SIDE BIT
0688 ; TO 1 (SINGLE SIDED)
0689 ;
0564 DB63      0690 CKDS:  IN     A,(SELECT)   ; GET DRIVE SELECT STATUS
0566 E60F      0691 AND     DRIVM          ; ISOLATE DRIVE # SELECT BITS
0568 0EFF      0692 LD     C,0FFH          ; INITIALIZE DRIVE # SO FIRST INC ZEROES REG
056A 0C        0693 CKDSL: INC  C           ; INCREMENT SELECTED DRIVE NUMBER
056B 1F        0694 RRA          ; CHECK IF THIS IS THE SELECTED DRIVE
056C 38FC      0695 JR     C,CKDSL        ; GO INCREMENT DRIVE # AND CHECK AGAIN IF NOT
056E 3AFF0A    0696 LD     A,(DRIVFT)     ; GET DRIVE NUMBER SELECTED BY THE OPERATOR
0571 B9        0697 CP     C             ; CHECK IF SAME AS CURRENTLY SELECTED DRIVE
0572 C8        0698 RET     Z             ; RETURN IF SAME
0573 CD4405    0699 CALL  GTDSC2          ; GET DRIVE ENTERED BY OPERATOR AND BUILD UNIT
                                ; PORTION OF SELECT COMMAND
0700 ;
0576 CD0206    0701 ISELS: CALL  SDRSEL      ; SEND SELECT COMMAND
0579 C8        0702 RET     Z             ; RETURN IF SELECT O. K., DISKETTE IS DOUBLE
                                ; SIDED IF SIDE BIT IS 0
0703 ;
057A DB63      0704 IN     A,(SELECT)     ; GET CURRENT DRIVE SELECT STATUS
057C 5F        0705 LD     E,A           ; SAVE SELECT STATUS IN E
057D E610      0706 AND     SIDEM          ; CHECK IF SIDE 1 HAS BEEN RESELECTED
057F 2005      0707 JR     NZ,ISELDS      ; GO DESELECT IF SIDE 1, NOT READY
0581 7B        0708 LD     A,E           ; CURRENT DRIVE SELECT STATUS BACK TO A
0582 F610      0709 OR     SIDEM          ; SET BIT TO SELECT SIDE 1, NOT DOUBLE SIDED
0584 18F0      0710 JR     ISELS          ; GO SEND SELECT COMMAND FOR SIDE 1
0586 1E80      0711 ISELDS: LD   E,DNRDYM   ; SET DRIVE NOT READY STATUS
0588 3EFF      0712 LD     A,DESEL        ; SET UP DESELECT COMMAND
058A D363      0713 OUT   (SELECT),A     ; SEND DESELECT COMMAND
058C 7B        0714 LD     A,E           ; DRIVE NOT READY STATUS TO A
058D 18AD      0715 JR     OPERR          ; GO SET CONTROLLER ERROR STATUS AND RETURN
                                ; WITH NON-ZERO FLAG SET IF NOT READY
0716 ;
0717 ;
0718 ; THIS SUBROUTINE POSITIONS THE HEAD TO THE NEXT CYLINDER (TRACK) TO FORMAT

```

```

0719 ;
058F DB63 0720 WTINIT: IN A,(SELECT) ; GET DRIVE SELECT STATUS
0591 CD4C05 0721 CALL GTDSCM ; SEND DRIVE SELECT/DENSITY COMMAND
0594 1E91 0722 LD E,STERM ; GET STATUS ERROR MASK
0596 3A010B 0723 LD A,(TKNUMF) ; GET # OF NEXT TRACK TO FORMAT
0599 4F 0724 LD C,A ; TRACK NUMBER TO FORMAT TO C
059A DB65 0725 IN A,(TRACK) ; GET TRACK # OF CURRENT HEAD POSITION
059C B9 0726 CP C ; CHECK IF POSITIONED AT NEXT TRACK TO FORMAT
059D 2827 0727 JR Z,STIORG ; GO SET UP BUFFER SIZE AND NUMBER OF BUFFERS
0728 ; IF ALREADY POSITIONED
059F 3E7C 0729 LD A,STEPO ; GET STEP OUT COMMAND
05A1 3002 0730 JR NC,WTSTEP ; GO STORE STEP OUT OF POSITIONED GREATER THAN
0731 ; NEXT TRACK TO FORMAT
05A3 3E5C 0732 LD A,STEPI ; GET STEP IN COMMAND
05A5 320B0B 0733 WTSTEP: LD (LSSTEP),A ; SAVE LAST STEP COMMAND SENT
05A8 0603 0734 LD B,003H ; SET SEEK RETRY COUNT TO 3
05AA 79 0735 WTRTRY: LD A,C ; GET TRACK NUMBER OF NEXT TRACK TO FORMAT
05AB D367 0736 OUT (DATA),A ; SET DATA REGISTER TO NEXT TRACK
05AD 3E18 0737 LD A,SKHDL ; SET UP SEEK, HEAD LOAD
05AF C5 0738 PUSH BC ; SAVE TRACK AND RETRY COUNT ON THE STACK
05B0 CD1605 0739 CALL SCOMME ; SEND COMMAND, USE MASK IN E TO CHECK FOR
0740 ; ERRORS
05B3 C1 0741 POP BC ; GET TRACK AND RETRY COUNT
05B4 2810 0742 JR Z,STIORG ; GO SET UP BUFFER SIZE AND NUMBER OF BUFFERS
0743 ; IF NO ERRORS ON SEEK
05B6 3E08 0744 LD A,RESTOR ; GET RESTORE COMMAND
05B8 C5 0745 PUSH BC ; SAVE TRACK NUMBER AND RETRY COUNT
05B9 CD1605 0746 CALL SCOMME ; SEND RESTORE COMMAND WITH SAME MASK
05BC C1 0747 POP BC ; RESTORE TRACK # AND RETRY COUNT
05BD 1E91 0748 LD E,STERM ; RESET ERROR MASK IN CASE OF ERROR ON RESTORE
05BF 10E9 0749 DJNZ WTRTRY ; GO SEND SEEK COMMAND IF ANOTHER RETRY O. K.
05C1 3E10 0750 LD A,SEEKEM ; SET TRACK NOT FOUND STATUS
05C3 C33C05 0751 JP OPERR ; GO SET CONTROLLER ERROR STATUS AND RETURN
0752 ; WITH NON-ZERO (ERROR) FLAG SET
0753 ;
0754 ; THIS ENTRY POINT SETS UP BUFFER SIZE AND NUMBER OF BUFFERS BASED ON DENSITY
0755 ; AND THEN SETS WAIT ENABLE DRIVE STATUS
0756 ;
05C6 2AFD0A 0757 STIORG: LD HL,(ACTBFA) ; SET INPUT/OUTPUT ADDRESS TO START OF ACTIVE
0758 ; BUFFER
05C9 0600 0759 LD B,000H ; SET BLOCK SIZE TO 256 BYTES
05CB CD4704 0760 CALL GETDEN ; GET FORMAT/DENSITY FLAG
05CE 2809 0761 JR Z,STIOSD ; GO CHANGE TO 128 BYTE BLOCKS IF SINGLE
0762 ; DENSITY
05D0 B7 0763 OR A ; CHECK IF DOUBLE DENSITY (26 256 BYTE SECTORS)
05D1 1E01 0764 LD E,001H ; SET BLOCKS PER SECTOR TO 1 (DOUBLE DENSITY)
05D3 2808 0765 JR Z,STIOD ; GO SET WAIT ENABLE IF DOUBLE DENSITY
05D5 1E04 0766 LD E,004H ; SET BLOCKS PER SECTOR TO 4 (MAXIMUM DENSITY)
05D7 1804 0767 JR STIOD ; GO SET WAIT ENABLE IF MAXIMUM DENSITY
05D9 0680 0768 STIOSD: LD B,080H ; SET BLOCK SIZE TO 128 BYTES
05DB 1E01 0769 LD E,001H ; SET BLOCKS PER SECTOR TO 1
05DD DB63 0770 STIOD: IN A,(SELECT) ; GET DRIVE SELECT STATUS
05DF E67F 0771 AND EWAITM ; SET UP WAIT ENABLE
05E1 D363 0772 OUT (SELECT),A ; SEND DRIVE SELECT COMMAND WITH WAIT ENABLE
05E3 AF 0773 XOR A ; SET OPERATION SUCCESSFUL STATUS
05E4 C9 0774 RET
0775 ;
0776 ; THIS SUBROUTINE SETS THE 15 MSEC DELAY BIT IN THE COMMAND IN A IF HEAD NOT
0777 ; LOADED
0778 ;
05E5 57 0779 SEHDNL: LD D,A ; SAVE COMMAND IN D
05E6 3ED0 0780 LD A,FORINT ; SET UP FORCE INTERRUPT COMMAND (TERMINATE
0781 ; ACTIVE COMMAND AND SET UP STATUS)
05E8 D364 0782 OUT (CMD),A ; SEND FORCE INTERRUPT COMMAND
05EA F5 0783 PUSH AF ; DELAY BEFORE CHECKING
05EB F1 0784 POP AF ; OPERATION STATUS
05EC DB64 0785 IN A,(STATUS) ; GET CONTROLLER STATUS
05EE E620 0786 AND HDNTLM ; CHECK IF HEAD LOADED
05F0 7A 0787 LD A,D ; GET COMMAND
05F1 C0 0788 RET NZ ; RETURN IF HEAD IS LOADED
05F2 F604 0789 OR EBITM ; SET E BIT IN COMMAND FOR 15 MSEC DELAY
05F4 C9 0790 RET
0791 ;
0792 ; THIS SUBROUTINE CONVERTS THE DRIVE NUMBER TO THE DRIVE BIT IN THE DRIVE
0793 ; SELECT COMMAND
0794 ;
05F5 E603 0795 DNTODS: AND 003H ; ISOLATE DRIVE NUMBER ENTERED BY OPERATOR
05F7 B7 0796 OR A ; CHECK IF DRIVE A
05F8 47 0797 LD B,A ; DRIVE NUMBER TO B FOR LOOP COUNT TO SHIFT
0798 ; DRIVE BIT

```

```

05F9 3E01      0799      LD      A,001H      ; START WITH DRIVE A BIT
05FB 2803      0800      JR      Z,DSCPL     ; IF DRIVE A, GO COMPLEMENT FOR SELECT COMMAND
05FD 07         0801      DNCVL:  RLCA        ; MOVE DRIVE BIT TO NEXT DRIVE POSITION
05FE 10FD      0802      DJNZ   DNCVL       ; GO SHIFT AGAIN IF NOT CORRECT DRIVE POSITION
0600 2F         0803      DSCPL:  CPL         ; COMPLEMENT FOR SELECT COMMAND REVERSE LOGIC
0601 C9         0804      RET
0805 ;
0806 ; THIS SUBROUTINE SENDS THE DRIVE/DENSITY/SIDE SELECT COMMAND IN A
0807 ;
0602 D363      0808      SDRSEL: OUT      (SELECT),A      ; SEND DRIVE/DENSITY/SIDE SELECT
0604 0623      0809      LD      B,023H     ; SET OUTER LOOP DELAY LOOP COUNT
0606 3EB2      0810      SDSLL1: LD      A,0B2H     ; SET INNER LOOP DELAY LOOP COUNT
0608 3D         0811      SDSLL2: DEC     A         ; DECREMENT INNER LOOP COUNT
0609 C20806    R 0812      JP      NZ,SDSLL2   ; GO DECREMENT AGAIN IF NOT EXPIRED
060C 05         0813      DEC     B         ; DECREMENT OUTER LOOP COUNT
060D 20F7      0814      JR      NZ,SDSLL1   ; GO RUN INNER LOOP IF NOT EXPIRED NOTE: THIS
0815 ; DELAYS 22.078 MSEC. VERSAFLOPPY II BOOK SAYS
0816 ; TO DELAY AT LEAST 18 MSEC BEFORE CHECKING
0817 ; STATUS
060F DB64      0818      IN      A,(STATUS)  ; GET CONTROLLER STATUS
0611 E680      0819      AND    DNRDYM      ; ISOLATE DRIVE NOT READY BIT
0613 C9         0820      RET
0821 ;
0822 ; THIS SUBROUTINE BUILDS THE SINGLE DENSITY TRACK FORMAT BUFFER TO FORMAT
0823 ; SINGLE DENSITY TRACKS
0824 ;
0614 113F0F    0825      BSDTKB: LD      DE,SDBUFF+1 ; GET ADDRESS OF SD TRACK BUFFER+1
0617 213E0F    0826      LD      HL,SDBUFF   ; GET ADDRESS OF SD TRACK BUFFER
061A 36FF      0827      LD      (HL),0FFH   ; SET FIRST BYTE TO TRACK PREAMBLE BYTE
061C 012800    0828      LD      BC,00028H   ; PREPARE TO SET 40 BYTE PREAMBLE
061F EDB0      0829      LDIR   ; STORE 40 MORE BYTES OF 0FFH
0621 3600      0830      LD      (HL),000H   ; STORE FIRST SYNC FIELD BYTE NOTE: THIS BYTE
0831 ; OVERWRITES THE LAST BYTE OF 0FFH PREVIOUSLY
0832 ; STORED SO ONLY 40 BYTES OF PREAMBLE REMAIN
0623 010600    0833      LD      BC,00006H   ; PREPARE TO SET SYNC FIELD
0626 EDB0      0834      LDIR   ; STORE 6 MORE BYTES OF 0
0628 36FC      0835      LD      (HL),0FCH   ; STORE INDEX MARK
062A 23        0836      INC    HL           ; INCREMENT BUFFER ADDRESSES
062B 13        0837      INC    DE           ; TO POST INDEX GAP
062C 36FF      0838      LD      (HL),0FFH   ; STORE FIRST POST INDEX GAP BYTE
062E 011A00    0839      LD      BC,0001AH   ; STORE 26 MORE BYTES OF 0FF
0631 EDB0      0840      LDIR   ; SET UP POST INDEX GAP
0633 3E1A      0841      LD      A,01AH     ; SET SECTORS AREAS LEFT TO INITIALIZE TO 26
0635 3600      0842      BSDTKL: LD      (HL),000H ; STORE FIRST BYTE OF SYNC FIELD
0637 010600    0843      LD      BC,00006H   ; STORE 6 BYTES OF ID SYNC FIELD
063A EDB0      0844      LDIR   ; SET UP ID SYNC FIELD
063C 36FE      0845      LD      (HL),0FEH   ; STORE ID ADDRESS MARK BYTE (ALSO IS A FLAG
0846 ; FOR THE CONTROLLER TO START CRC CALCULATION)
063E 23        0847      INC    HL           ; INCREMENT BUFFER ADDRESSES
063F 13        0848      INC    DE           ; TO CYLINDER #
0640 3600      0849      LD      (HL),000H   ; SET FIRST ID BYTE TO 0
0642 010400    0850      LD      BC,00004H   ; STORE 0 IN 4 ID BYTES
0645 EDB0      0851      LDIR   ; INITIALIZE ID BYTES TO 0
0647 36F7      0852      LD      (HL),0F7H   ; STORE FLAG TO WRITE CRC CHARACTERS
0649 23        0853      INC    HL           ; INCREMENT BUFFER ADDRESSES
064A 13        0854      INC    DE           ; TO POST ID GAP
064B 36FF      0855      LD      (HL),0FFH   ; STORE FIRST BYTE OF POST ID GAP
064D 010B00    0856      LD      BC,0000BH   ; STORE 11 BYTES OF 0FFH IN POST ID GAP
0650 EDB0      0857      LDIR   ; SET UP POST ID GAP
0652 3600      0858      LD      (HL),000H   ; STORE FIRST BYTE OF DATA SYNC FIELD
0654 010600    0859      LD      BC,00006H   ; STORE 0 IN 6 DATA SYNC FIELD BYTES
0657 EDB0      0860      LDIR   ; SET UP SECTOR DATA SYNC FIELD
0659 36FB      0861      LD      (HL),0FBH   ; STORE DATA ADDRESS MARK (ALSO IS A FLAG FOR
0862 ; THE CONTROLLER TO START DATA CRC CALCULATION)
065B 23        0863      INC    HL           ; INCREMENT BUFFER ADDRESSES
065C 13        0864      INC    DE           ; TO SECTOR DATA
065D 36E5      0865      LD      (HL),0E5H   ; SET FIRST SECTOR DATA BYTE TO 0E5H
065F 018000    0866      LD      BC,00080H   ; STORE 128 BYTES OF 0E5H IN SECTOR DATA FIELD
0662 EDB0      0867      LDIR   ; INITIALIZE SECTOR DATA BYTES TO 0E5H
0664 36F7      0868      LD      (HL),0F7H   ; STORE CONTROLLER FLAG TO WRITE CRC BYTES
0666 23        0869      INC    HL           ; INCREMENT BUFFER ADDRESSES
0667 13        0870      INC    DE           ; TO POST DATA GAP
0668 36FF      0871      LD      (HL),0FFH   ; STORE FIRST BYTE OF POST DATA GAP
066A 011B00    0872      LD      BC,0001BH   ; STORE 27 BYTES OF 0FFH IN POST DATA GAP
066D EDB0      0873      LDIR   ; SET UP POST DATA GAP
066F 3D         0874      DEC     A         ; CHECK IF ALL 26 SECTORS OF FORMAT DATA STORED
0670 C23506    R 0875      JP      NZ,BSDTKL   ; GO STORE ANOTHER SECTOR OF FORMAT DATA IF NOT
0673 36FF      0876      LD      (HL),0FFH   ; STORE 0FFH IN FIRST BYTE OF FINAL PRE-INDEX
0877 ; GAP
0675 01FA00    0878      LD      BC,000FAH   ; STORE 250 MORE BYTES OF FINAL PRE-INDEX GAP

```

```

0678 EDB0      0879      LDIR          ; SET UP FINAL PRE-INDEX GAP
067A C9        0880      RET
0881      ;
0882      ; THIS SUBROUTINE STORES THE SECTOR ID BYTES (CYLINDER, HEAD, AND SECTOR) IN
0883      ; THE SINGLE DENSITY TRACK FORMAT BUFFER
0884      ;
067B 218E0F   0885 STIDSD: LD      HL,SDBUFF+050H ; GET ADDRESS OF CYLINDER # IN FIRST SECTOR ID
067E 3E01      0886      LD          A,001H           ; INITIALIZE SECTOR NUMBER TO 1
0680 71        0887 STIDSL: LD      (HL),C        ; STORE CYLINDER # IN NEXT SECTOR ID
0681 23        0888      INC          HL           ; BUMP ID ADDRESS TO HEAD #
0682 70        0889      LD          (HL),B        ; STORE HEAD # (SIDE) IN NEXT SECTOR ID
0683 23        0890      INC          HL           ; BUMP ID ADDRESS TO SECTOR #
0684 77        0891      LD          (HL),A        ; STORE SECTOR # IN NEXT SECTOR ID
0685 3C        0892      INC          A           ; INCREMENT TO NEXT SECTOR #
0686 FE1B      0893      CP          01BH          ; CHECK IF ALL 26 SECTOR ID'S STORED
0688 C8        0894      RET          Z           ; RETURN IF ALL SECTOR ID'S STORED IN TRACK
0689 11B800    0895      LD          DE,000B8H      ; GET OFFSET TO NEXT SECTOR ID (CYLINDER #)
068C 19        0896      ADD         HL,DE          ; COMPUTE ADDRESS OF CYLINDER # ID IN NEXT
068D 18F1      0897      JR          STIDSL       ; GO STORE ID'S IN NEXT SECTOR
0898      ;
0899      ; THIS SUBROUTINE SETS UP A DOUBLE DENSITY (26 256 BYTE SECTORS) OR MAXIMUM
0900      ; DENSITY (9 1024 BYTE SECTORS) BASED ON THE FORMAT/DENSITY FLAG
0901      ;
068F CD4704    0902 BDDTKB: CALL   GETDEN        ; GET THE FORMAT/DENSITY FLAG
0692 B7        0903      OR          A           ; CHECK IF DOUBLE DENSITY WAS SELECTED
0693 C23107     0904      JP          NZ,BMDTKB     ; GO FORMAT MAXIMUM DENSITY IF NOT
0696 113F27    0905      LD          DE,DBUFF+1     ; GET ADDRESS OF DOUBLE DENSITY TRACK BUFFER+1
0699 213E27    0906      LD          HL,DBUFF       ; GET ADDRESS OF DOUBLE DENSITY TRACK BUFFER
069C 364E      0907      LD          (HL),04EH      ; SET FIRST PREAMBLE BYTE TO 04EH
069E 015000    0908      LD          BC,00050H      ; STORE 80 BYTES OF 04EH IN PREAMBLE
06A1 EDB0      0909      LDIR        ; SET UP TRACK PREAMBLE
06A3 3600      0910      LD          (HL),000H      ; STORE FIRST SYNC FIELD BYTE
06A5 010C00   0911      LD          BC,0000CH      ; STORE 12 BYTES OF 0 IN SYNC FIELD
06A8 EDB0      0912      LDIR        ; SET UP SYNC FIELD
06AA 36F6      0913      LD          (HL),0F6H      ; SET FIRST TRACK SYNC BYTE TO 0F6H (WRITE 0C2H
0914      ; IN MFM)
06AC 010300   0915      LD          BC,00003H      ; STORE 3 BYTES OF 0F6H
06AF EDB0      0916      LDIR        ; SET UP TRACK SYNC BYTES
06B1 36FC      0917      LD          (HL),0FCH      ; SET TRACK INDEX MARK TO 0FC
06B3 23        0918      INC          HL           ; INCREMENT BUFFER ADDRESSES
06B4 13        0919      INC          DE           ;
06B5 364E      0920      LD          (HL),04EH      ; SET FIRST POST INDEX GAP BYTE TO 04EH
06B7 013200   0921      LD          BC,00032H      ; STORE 50 BYTES OF 04EH
06BA EDB0      0922      LDIR        ; SET UP POST INDEX GAP
06BC 3E1A      0923      LD          A,01AH         ; SET SECTOR AREAS LEFT TO INITIALIZE TO 26
06BE 3600      0924 BDDTKL: LD      (HL),000H      ; STORE 0 IN FIRST SECTOR SYNC FIELD BYTE
06C0 010C00   0925      LD          BC,0000CH      ; STORE 12 BYTES OF 0 IN SECTOR SYNC FIELD
06C3 EDB0      0926      LDIR        ; SET UP SECTOR SYNC FIELD FOR NEXT SECTOR
06C5 36F5      0927      LD          (HL),0F5H      ; SET FIRST SECTOR SYNC BYTE TO 0F5H (WRITE
0928      ; 0A1H IN MFM)
06C7 010300   0929      LD          BC,00003H      ; STORE 3 SECTOR SYNC BYTES OF 0F5H
06CA EDB0      0930      LDIR        ; SET UP SECTOR SYNC BYTES
06CC 36FE      0931      LD          (HL),0FEH      ; STORE ID ADDRESS MARK (ALSO A FLAG FOR THE
0932      ; CONTROLLER TO START ID FIELD CRC CALCULATION)
06CE 23        0933      INC          HL           ; INCREMENT BUFFER ADDRESSES
06CF 13        0934      INC          DE           ;
06D0 3600      0935      LD          (HL),000H      ; SET FIRST ID BYTE (CYLINDER #) TO 0
06D2 010300   0936      LD          BC,00003H      ; STORE 0 IN ALL 3 SECTOR ID BYTES
06D5 EDB0      0937      LDIR        ; SET SECTOR ID BYTES TO 0
06D7 3601      0938      LD          (HL),001H      ; SET SIZE BYTE TO 1 (256 BYTE SECTORS)
06D9 23        0939      INC          HL           ; INCREMENT BUFFER ADDRESSES
06DA 13        0940      INC          DE           ;
06DB 36F7      0941      LD          (HL),0F7H      ; STORE CONTROLLER FLAG BYTE TO WRITE 2 CRC
0942      ; BYTES
06DD 23        0943      INC          HL           ; INCREMENT BUFFER ADDRESSES
06DE 13        0944      INC          DE           ;
06DF 364E      0945      LD          (HL),04EH      ; SET FIRST POST ID GAP BYTE TO 04EH
06E1 011600   0946      LD          BC,00016H      ; STORE 26 BYTES OF 04EH IN POST ID GAP
06E4 EDB0      0947      LDIR        ; SET UP POST ID GAP
06E6 3600      0948      LD          (HL),000H      ; SET FIRST SECTOR SYNC FIELD BYTE TO 0
06E8 010C00   0949      LD          BC,0000CH      ; STORE 12 BYTES OF 0 IN SYNC FIELD
06EB EDB0      0950      LDIR        ; SET UP SYNC FIELD
06ED 36F5      0951      LD          (HL),0F5H      ; STORE 0F5H IN FIRST SECTOR SYNC BYTE (WRITE
0952      ; 0A1H IN MFM)
06EF 010300   0953      LD          BC,00003H      ; STORE 3 BYTES OF 0F5H IN SYNC BYTES
06F2 EDB0      0954      LDIR        ; SET UP SECTOR SYNC BYTES
06F4 36FB      0955      LD          (HL),0FBH      ; STORE 0FBH IN DATA ADDRESS MARK (FLAGS
0956      ; CONTROLLER TO START CRC CALCULATION)
06F6 23        0957      INC          HL           ; INCREMENT BUFFER ADDRESSES
06F7 13        0958      INC          DE           ;

```

```

06F8 36E5      0959      LD      (HL),0E5H      ; SET FIRST SECTOR DATA BYTE TO 0E5H
06FA 010001    0960      LD      BC,00100H      ; STORE 256 BYTES OF 0E5H IN SECTOR DATA
06FD EDB0       0961      LDIR                       ; SET UP SECTOR DATA
06FF 36F7      0962      LD      (HL),0F7H      ; STORE CONTROLLER FLAG TO WRITE 2 CRC BYTES
0701 23        0963      INC     HL              ; INCREMENT BUFFER ADDRESSES
0702 13        0964      INC     DE              ; TO POST DATA GAP
0703 364E      0965      LD      (HL),04EH      ; SET FIRST POST DATA GAP BYTE TO 04EH
0705 013600    0966      LD      BC,00036H      ; STORE 54 BYTES OF 04EH IN POST DATA GAP
0708 EDB0       0967      LDIR                       ; SET UP POST DATA GAP
070A 3D        0968      DEC     A              ; CHECK IF ALL SECTOR FORMAT AREAS INITIALIZED
070B C2BE06     R 0969      JP      NZ,BDDTKL      ; GO STORE NEXT SECTOR FORMAT AREA IF NOT
070E 364E      0970      LD      (HL),04EH      ; SET FIRST FINAL PRE-INDEX BYTE TO 04EH
0710 015802    0971      LD      BC,00258H      ; STORE 600 MORE BYTES OF PRE-INDEX GAP
0713 EDB0       0972      LDIR                       ; SET UP PRE-INDEX GAP
0715 C9        0973      RET

0974 ;
0975 ; THIS SUBROUTINE STORES THE SECTOR ID'S IN EITHER DOUBLE DENSITY (26 256 BYTE
0976 ; SECTORS) OR MAXIMUM DENSITY (9 1024 BYTE SECTORS) TRACK FORMAT BUFFER BASED
0977 ; ON THE FORMAT/DENSITY FLAG
0978 ;
0716 CD4704    0979 STIDDD: CALL  GETDEN      ; GET THE FORMAT/DENSITY FLAG
0719 B7         0980      OR      A              ; CHECK IF DOUBLE DENSITY BUFFER
071A C2B107    0981      JP      NZ,STIDMD      ; GO STORE MAXIMUM DENSITY ID'S IF NOT DOUBLE
071D 21E027    0982      LD      HL,DBUFF+0A2H  ; GET ADDRESS OF FIRST DOUBLE DENSITY BUFFER
0983 ; ID BYTE (CYLINDER #)
0720 3E01      0984      LD      A,001H        ; INITIALIZE SECTOR NUMBER ID TO 1
0722 71        0985      STIDDL: LD      (HL),C  ; STORE CYLINDER # IN NEXT SECTOR ID BYTE
0723 23        0986      INC     HL            ; BUMP ID ADDRESS TO HEAD #
0724 70        0987      LD      (HL),B        ; STORE HEAD # IN NEXT SECTOR ID BYTE
0725 23        0988      INC     HL            ; BUMP ID ADDRESS TO SECTOR #
0726 77        0989      LD      (HL),A        ; STORE SECTOR # IN NEXT SECTOR ID BYTE
0727 3C        0990      INC     A              ; INCREMENT TO NEXT SECTOR NUMBER
0728 FE1B      0991      CP      01BH         ; CHECK IF ALL SECTOR ID'S ARE STORED
072A C8        0992      RET      Z            ; RETURN IF ALL SECTOR ID'S STORED
072B 117001    0993      LD      DE,00170H     ; GET OFFSET OF NEXT SECTOR ID BYTES
072E 19        0994      ADD     HL,DE         ; COMPUTE ADDRESS OF NEXT SECTOR ID BYTES
072F 18F1      0995      JR      STIDDL        ; GO STORE ID BYTES FOR NEXT SECTOR
0996 ;
0997 ; THIS ENTRY POINT SETS UP A TRACK FORMAT BUFFER TO FORMAT MAXIMIM DENSITY
0998 ; (9 1024 BYTE SECTORS)
0999 ;
0731 113F27    1000 BMDTKB: LD      DE,DBUFF+1 ; GET ADDRESS OF MAXIMUM DENSITY BUFFER+1
0734 213E27    1001      LD      HL,DBUFF      ; GET ADDRESS OF MAXIMUM DENSITY BUFFER
0737 364E      1002      LD      (HL),04EH     ; SET FIRST BYTE OF TRACK PREAMBLE TO 04EH
0739 011000    1003      LD      BC,00010H     ; STORE 16 BYTES OF 04E (IBM STANDARD IS 80)
073C EDB0       1004      LDIR                       ; SET UP TRACK PREAMBLE
073E 3600      1005      LD      (HL),000H     ; SET FIRST SYNC FIELD BYTE TO 0
0740 010C00    1006      LD      BC,0000CH     ; STORE 12 SYNC FIELD BYTES OF 0
0743 EDB0       1007      LDIR                       ; SET UP TRACK SYNC FIELD
0745 36F6      1008      LD      (HL),0F6H     ; SET FIRST TRACK SYNC BYTE TO 0F6H
1009 ; (CONTROLLER WILL WRITE 0C2H MFM)
0747 010300    1010      LD      BC,00003H     ; STORE 3 BYTES OF 0F6H IN TRACK SYNC
074A EDB0       1011      LDIR                       ; SET UP TRACK SYNC BYTES
074C 36FC      1012      LD      (HL),0FCH     ; SET TRACK INDEX MARK TO 0FCH
074E 23        1013      INC     HL            ; INCREMENT BUFFER ADDRESSES
074F 13        1014      INC     DE              ; TO POST INDEX GAP
0750 364E      1015      LD      (HL),04EH     ; SET FIRST POST INDEX GAP BYTE TO 04EH
0752 011000    1016      LD      BC,00010H     ; STORE 16 BYTES OF 04E (IBM STANDARD IS 50)
0755 EDB0       1017      LDIR                       ; SET UP POST INDEX GAP
0757 3E09      1018      LD      A,009H        ; SET SECTOR AREAS LEFT TO INITIALIZE TO 9
0759 3600      1019 BMDTKL: LD      (HL),000H ; SET FIRST SECTOR ID SYNC FIELD BYTE TO 0
075B 010C00    1020      LD      BC,0000CH     ; STORE 12 SECTOR ID SYNC FIELD BYTES OF 0
075E EDB0       1021      LDIR                       ; SET UP SECTOR ID SYNC FIELD
0760 36F5      1022      LD      (HL),0F5H     ; SET FIRST SECTOR ID SYNC BYTE TO 0F5H
1023 ; (CONTROLLER WILL WRITE 0A1H)
0762 010300    1024      LD      BC,00003H     ; STORE 3 SECTOR ID SYNC BYTES OF 0F5H
0765 EDB0       1025      LDIR                       ; SET UP SECTOR ID SYNC BYTES
0767 36FE      1026      LD      (HL),0FEH     ; SET ID ADDRESS MARK TO 0FEH (ALSO FLAGS
1027 ; CONTROLLER TO START CRC CALCULATION FOR ID)
0769 23        1028      INC     HL            ; INCREMENT BUFFER ADDRESSES
076A 13        1029      INC     DE              ; TO FIRST ID BYTE
076B 3600      1030      LD      (HL),000H     ; SET FIRST ID BYTE (CYLINDER #) TO 0
076D 010300    1031      LD      BC,00003H     ; STORE 0 IN 3 ID BYTES
0770 EDB0       1032      LDIR                       ; SET FIRST 3 ID'S TO 0
0772 3603      1033      LD      (HL),003H     ; SET SECTOR SIZE TO 3 (1024 BYTES)
0774 23        1034      INC     HL            ; INCREMENT BUFFER ADDRESSES
0775 13        1035      INC     DE              ; TO CRC BYTES
0776 36F7      1036      LD      (HL),0F7H     ; STORE CONTROLLER FLAG BYTE TO WRITE 2 CRC
1037 ; BYTES
0778 23        1038      INC     HL            ; INCREMENT BUFFER ADDRESSES

```

```

0779 13 1039 INC DE ; TO POST ID GAP
077A 364E 1040 LD (HL),04EH ; SET FIRST POST ID GAP BYTE TO 04EH
077C 011600 1041 LD BC,00016H ; STORE 22 BYTES OF 04EH IN POST ID GAP
077F EDB0 1042 LDIR ; SET UP POST ID GAP
0781 3600 1043 LD (HL),000H ; SET FIRST SECTOR SYNC FIELD BYTE TO 0
0783 010C00 1044 LD BC,0000CH ; STORE 0 IN 12 SECTOR SYNC FIELD BYTES
0786 EDB0 1045 LDIR ; SET UP SECTOR SYNC
0788 36F5 1046 LD (HL),0F5H ; SET FIRST SECTOR SYNC BYTE TO 0F5H
1047 ; (CONTROLLER WILL WRITE 0A1H)
078A 010300 1048 LD BC,00003H ; STORE 3 BYTES OF 0F5H IN SECTOR SYNC BYTES
078D EDB0 1049 LDIR ; SET UP SECTOR SYNC BYTES
078F 36FB 1050 LD (HL),0FBH ; SET DATA ADDRESS MARK TO 0FBH (ALSO FLAGS
1051 ; CONTROLLER TO START CRC CALCULATION
0791 23 1052 INC HL ; INCREMENT BUFFER ADDRESSES
0792 13 1053 INC DE ; TO DATA FIELD
0793 36E5 1054 LD (HL),0E5H ; SET FIRST SECTOR DATA BYTE TO 0E5H
0795 010004 1055 LD BC,00400H ; STORE 1024 BYTES OF 0E5H IN SECTOR DATA FIELD
0798 EDB0 1056 LDIR ; SET UP SECTOR DATA FIELD
079A 36F7 1057 LD (HL),0F7H ; SET FIRST CRC BYTE TO 0F7H (FLAGS CONTROLLER
1058 ; TO STORE 2 CRC BYTES
079C 23 1059 INC HL ; INCREMENT BUFFER ADDRESSES
079D 13 1060 INC DE ; TO POST DATA GAP
079E 364E 1061 LD (HL),04EH ; SET FIRST POST DATA GAP BYTE TO 04EH
07A0 011000 1062 LD BC,00010H ; STORE 16 BYTES OF 04E IN POST DATA GAP (IBM
1063 ; STANDARD IS 116)
07A3 EDB0 1064 LDIR ; SET UP POST DATA GAP
07A5 3D 1065 DEC A ; CHECK IF ALL 9 SECTOR AREAS INITIALIZED
07A6 C25907 R 1066 JP NZ,BMDTKL ; GO INITIALIZE NEXT SECTOR AREA IF NOT
07A9 364E 1067 LD (HL),04EH ; SET FIRST FINAL PRE-INDEX GAP BYTE TO 04EH
07AB 015802 1068 LD BC,00258H ; STORE 600 MORE BYTES OF FINAL PRE-INDEX GAP
07AE EDB0 1069 LDIR ; SET UP FINAL PRE-INDEX GAP
07B0 C9 1070 RET
1071 ;
1072 ; THIS ENTRY POINT STORES THE SECTOR ID BYTES IN THE MAXIMUM DENSITY (9 1024
1073 ; BYTE SECTORS) TRACK FORMAT BUFFER
1074 ;
07B1 3A080B 1075 STIDMD: LD A,(DENSFL) ; GET DENSITY FLAG)
07B4 B7 1076 OR A ; TEST FOR DOUBLE DENSITY
07B5 C8 1077 RET Z ; OOPS, GOT HERE BY MISTAKE
07B6 217E27 1078 LD HL,DBUFF+040H ; GET ADDRESS OF FIRST ID BYTE (CYLINDER #)
1079 ; IN THE MAXIMUM DENSITY BUFFER
07B9 3E01 1080 LD A,001H ; SET CURRENT SECTOR NUMBER TO 1
07BB 71 1081 STIDML: LD (HL),C ; STORE CYLINDER # IN NEXT ID BYTE
07BC 23 1082 INC HL ; BUMP BUFFER ADDRESS TO HEAD #
07BD 70 1083 LD (HL),B ; STORE HEAD # IN NEXT ID BYTE
07BE 23 1084 INC HL ; BUMP BUFFER ADDRESS TO SECTOR #
07BF 77 1085 LD (HL),A ; STORE SECTOR # IN NEXT ID BYTE
07C0 3C 1086 INC A ; BUMP SECTOR # TO NEXT SECTOR
07C1 FE0A 1087 CP 00AH ; CHECK IF ID'S STORED FOR ALL 9 SECTORS
07C3 C8 1088 RET Z ; RETURN IF ALL SECTOR ID'S STORED
07C4 114A04 1089 LD DE,0044AH ; GET OFFSET OF NEXT SECTOR ID'S
07C7 19 1090 ADD HL,DE ; COMPUTE ADDRESS OF NEXT SECTOR ID'S IN BUFFER
07C8 18F1 1091 JR STIDML ; GO STORE ID'S IN NEXT SECTOR ID FIELD
1092 ;
1093 ; THE FOLLOWING ARE TEXT MESSAGES USED BY THE FORMAT PROGRAM
1094 ;
07CA 0D0A24 1095 CRLFMG: DB CR,LF,'$'
07CD 0D0A0D0A 1096 TRKCUE: DB CR,LF,CR,LF,'ENTER TRACKS ( s or A ) - $'
07CE 0D0A454E 1097 EDRVCU: DB CR,LF,'ENTER DRIVE ( a B c or d ) - $'
080C 0D0A454E 1098 EDENCU: DB CR,LF,'ENTER DENSITY ( s d or X ) - $'
082C 0D0A0D0A 1099 RERNCU: DB CR,LF,CR,LF,'TO RUN AGAIN TYPE CR, ESC FOR OPTIONS,'
0856 205E4320 1100 DB ' ^C TO ABORT $'
0864 0D0A0D0A 1101 TKEMSG: DB CR,LF,CR,LF,'ERROR ON TRACK NO. '
087B 202024 1102 TKNMSG: DB ' $'
087E 20534543 1103 SCEMSG: DB ' SECTOR NO. '
088A 202024 1104 SCNMSG: DB ' $'
088D 0D0A202D 1105 FEMSG: DB CR,LF,' - FORMAT ERROR ON DISK.$'
08A8 0D0A202D 1106 REMSG: DB CR,LF,' - READ ERROR ON DISK.$'
08C1 0D0A0D0A 1107 TFMTCU: DB CR,LF,CR,LF,'TO FORMAT: $'
08D1 53595354 1108 SYSCU: DB 'SYSTEM TRACKS $'
08E1 414C4C20 1109 FILECU: DB 'ALL TRACKS $'
08EE 53494E47 1110 SINGCU: DB 'SINGLE $'
08F6 444F5542 1111 DOUBCU: DB 'DOUBLE $'
08FE 4D415849 1112 MAXCU: DB 'MAXIMUM $'
0907 44454E53 1113 DONDCU: DB 'DENSITY ON DRIVE '
0919 20202C0D 1114 DDCUS: DB ' ,',CR,LF,'TYPE CR TO CONTINUE, ^C TO END,'
093D 20455343 1115 DB ' ESC FOR NEW OPTIONS. $'
0955 0D0A0D0A 1116 TKNHDR: DB CR,LF,CR,LF,'0 1 2 3 4'
0982 20202020 1117 DB ' 5 6 7',CR,LF,'01234567890123'
09B0 34353637 1118 DB '45678901234567890123456789012345678901234567890123456'

```

```

09E5 37383930 1119 DB '7890123456$'
09F0 2A24 1120 ATSKMG: DB '*$'
1121 ;
1122 ; THIS SUBROUTINE MAKES A DIRECT CONSOLE I/O CALL TO CP/M BDOS
1123 ;
09F2 0E06 1124 DCONIO: LD C,06H ; GET DIRECT CONSOLE I/O CODE FOR BDOS
09F4 C30500 1125 JP BDOS ; CALL BDOS TO PERFORM DIRECT CONSOLE I/O
1126 ;
1127 ; THIS SUBROUTINE MAKES A CONSOLE OUTPUT CALL TO CP/M BDOS TO OUTPUT THE
1128 ; ASCII CHARACTER IN REGISTER E
1129 ;
09F7 0E02 1130 DCONO: LD C,002H ; GET CONSOLE OUTPUT CODE FOR BDOS
09F9 C30500 1131 JP BDOS ; CALL BDOS TO OUTPUT E TO THE CONSOLE
1132 ;
1133 ; THIS SUBROUTINE GETS THE OPERATOR RESPONSE TO A PREVIOUSLY DISPLAYED CUE AND
1134 ; CONVERTS THE RESPONSE FROM LOWER CASE TO UPPER CASE IF NECESSARY
1135 ;
09FC 21280A 1136 GTOPRP: LD HL,CONIBF+1 ; GET ADDRESS OF # OF CHARACTERS IN RESPONSE
1137 ; BUFFER
09FF 3600 1138 LD (HL),000H ; SET # OF CHARACTER IN BUFFER TO 0
0A01 23 1139 INC HL ; BUMP TO ADDRESS OF FIRST CHARACTER
0A02 22250A 1140 LD (CONBFC),HL ; SAVE ADDRESS OF FIRST CHARACTER
0A05 11270A 1141 LD DE,CONIBF ; GET ADDRESS OF CONSOLE BUFFER
0A08 0E0A 1142 LD C,00AH ; GET READ CONSOLE BUFFER REQUEST FOR BDOS
0A0A CD0500 1143 CALL BDOS ; CALL BDOS TO GET CUE RESPONSE
0A0D 21280A 1144 LD HL,CONIBF+1 ; GET # OF CHARACTERS IN THE BUFFER
0A10 35 1145 DEC (HL) ; CHECK IF ANY TYPED IN (ONLY CR ENTERED)
0A11 3E0D 1146 LD A,00DH ; SET RESPONSE TO CR
0A13 F8 1147 RET M ; RETURN WITH CR AS RESPONSE IF NONE
0A14 2A250A 1148 LD HL,(CONBFC) ; GET ADDRESS OF FIRST CHARACTER
0A17 7E 1149 LD A,(HL) ; GET CHARACTER ENTERED
0A18 23 1150 INC HL ; BUMP ADDRESS TO NEXT CHARACTER
0A19 22250A 1151 LD (CONBFC),HL ; SAVE ADDRESS OF NEXT CHARACTER
0A1C FE61 1152 CP 'a' ; CHECK IF CHARACTER < 'a'
0A1E D8 1153 RET C ; RETURN IF < 'a' , NO CONVERSION REQUIRED
0A1F FE7B 1154 CP 'z'+1 ; CHECK IF CHARACTER > 'z'
0A21 D0 1155 RET NC ; RETURN IF > 'z' , NO CONVERSION REQUIRED
0A22 E65F 1156 AND 05FH ; CHANGE RESPONSE FROM LOWER CASE TO UPPER CASE
0A24 C9 1157 RET
1158 ;
0A25 (0002) 1159 CONBFC: DS 2 ; ADDRESS OF FIRST CHARACTER IN THE
1160 ; RESPONSE BUFFER
0A27 7F00 1161 CONIBF: DB 07FH,0 ; CONSOLE INPUT BUFFER, BUFFER SIZE, NUMBER
1162 ; OF CHARACTERS IN THE BUFFER
0A29 (007F) 1163 DS 07FH ; BUFFER FOR CONSOLE CHARACTERS TYPED IN
1164 ;
1165 ; THIS SUBROUTINE CONVERTS THE NUMBER (<100) IN A TO ASCII DECIMAL AND STORES
1166 ; IT IN THE MESSAGE ADDRESS (HL), LEADING ZEROES SUPPRESSED
1167 ;
0AA8 060A 1168 BINDEC: LD B,00AH ; LOAD BASE (10) TO CONVERT NUMBER
0AAA CDB80A 1169 CALL CVTMSG ; CONVERT MOST SIGNIFICANT DIGIT OF NUMBER
1170 ; AND STORE IT AT (HL)
0AAD C630 1171 ADD A,'0' ; CONVERT REMAINDER TO ASCII
0AAF 77 1172 LD (HL),A ; STORE REMAINDER IN 1'S DIGIT
0AB0 2B 1173 DEC HL ; COMPUTE ADDRESS OF 10'S DIGIT OF NUMBER
0AB1 7E 1174 LD A,(HL) ; GET 10'S DIGIT OF NUMBER#
0AB2 FE30 1175 CP '0' ; CHECK IF '0'
0AB4 C0 1176 RET NZ ; RETURN IF NOT '0'
0AB5 3620 1177 LD (HL),' ' ; SUPPRESS LEADING '0'
0AB7 C9 1178 RET
1179 ;
1180 ; THIS SUBROUTINE CONVERTS THE MOST SIGNIFICANT DIGIT OF THE NUMBER IN A AND
1181 ; STORES IT AT (HL). REMAINDER (1'S DIGIT) IS LEFT IN A
1182 ;
0AB8 362F 1183 CVTMSG: LD (HL),'0'-1 ; STORE CONVERSION CONSTANT, '0'-1, IN 10'S
1184 ; DIGIT OF MESSAGE
0ABA 34 1185 CVTMSL: INC (HL) ; BUMP TO NEXT 10'S DIGIT
0ABB 90 1186 SUB B ; CHECK IF MORE 10'S LEFT IN NUMBER
0ABC D2BA0A R 1187 JP NC,CVTMSL ; GO BUMP TO NEXT 10'S DIGIT IF YES
0ABF 80 1188 ADD A,B ; RESTORE REMAINDER FROM 10'S DIGIT CONVERSION
0AC0 23 1189 INC HL ; BUMP TO ADDRESS OF 1'S DIGIT IN MESSAGE
0AC1 C9 1190 RET
1191 ;
1192 ; THIS SUBROUTINE CALLS CP/M BDOS TO DISPLAY THE MESSAGE WHOSE ADDRESS IS IN
1193 ; HL ON THE CONSOLE
1194 ;
0AC2 EB 1195 MSGCON: EX DE,HL ; MESSAGE ADDRESS TO DE FOR BDOS
0AC3 0E09 1196 LD C,009H ; SET FUNCTION CODE TO PRINT STRING TERMINATED
1197 ; BY A '$'
0AC5 C30500 1198 JP BDOS ; CALL BDOS TO DISPLAY STRING ON CONSOLE

```

```

1199 ;
1200 ; THIS SUBROUTINE INITIALIZES VECTORS TO THE CBIOS VECTORS TABLE
1201 ;
0AC8 3A0200 1202 LKBIOS: LD      A,(WBOOTV+2) ; GET ADDRESS OF WBOOT VECTOR IN CBIOS
0ACB 21DB0A 1203 LD      HL,CONST+2 ; GET ADDRESS TO STORE FIRST CBIOS BRANCH
1204 ; TABLE BIAS IN BRANCH TABLE
0ACE 060C 1205 LD      B,00CH ; GET NUMBER OF JUMP INSTRUCTIONS TO BIAS
0AD0 77 1206 LKBISL: LD     (HL),A ; BIAS NEXT JUMP TO CBIOS VECTOR
0AD1 23 1207 INC     HL ; INCREMENT TO
0AD2 23 1208 INC     HL ; NEXT BRANCH
0AD3 23 1209 INC     HL ;
0AD4 05 1210 DEC     B ; TABLE ADDRESS
0AD5 C2D00A R 1211 JP      NZ,LKBISL ; CHECK IF MORE JUMPS TO CBIOS VECTORS TO BIAS
0AD8 C9 1212 RET ; GO BIAS NEXT IF MORE
0AD9 C30600 1213 CONST: JP     00006H ; CONSOLE STATUS
0ADC C30900 1214 CONIN: JP     00009H ; CONSOLE CHARACTER IN
0ADF C30C00 1215 CONOUT: JP    0000CH ; CONSOLE CHARACTER OUT
0AE2 C31800 1216 HOME: JP     00018H ; MOVE DISK HEAD TO HOME POSITION
0AE5 C31B00 1217 SELDSK: JP    0001BH ; SELECT DISK
0AE8 C31E00 1218 SETTRK: JP    0001EH ; SET TRACK NUMBER
0AEB C32100 1219 SETSEC: JP    00021H ; SET SECTOR NUMBER
0AEE C32400 1220 SETDMA: JP    00024H ; SET DMA ADDRESS
0AF1 C32700 1221 READ: JP     00027H ; READ DISK
0AF4 C32A00 1222 WRITE: JP    0002AH ; WRITE DISK
0AF7 C32D00 1223 LISTST: JP   0002DH ; RETURN LIST STATUS
0AFA C33000 1224 SECTRN: JP   00030H ; SECTOR TRANSLATE
1225 ;
1226 ; THE FOLLOWING ARE DATA CELLS USED BY THE FORMAT PROGRAM
1227 ;
0AFD (0002) 1228 ACTBFA: DS     2 ; ADDRESS OF TRACK FORMAT BUFFER
0AFF (0001) 1229 DRIVFT: DS     1 ; DRIVE NUMBER TO FORMAT
0B00 (0001) 1230 NUMTRK: DS     1 ; NUMBER OF TRACKS TO FORMAT
0B01 (0001) 1231 TKNUMF: DS     1 ; TRACK NUMBER BEING FORMATTED
0B02 (0001) 1232 CSECT: DS     1 ; NEXT SECTOR TO READ
0B03 (0002) 1233 CTRACK: DS     2 ; CURRENT TRACK NUMBER
0B05 (0001) 1234 SSDSFG: DS     1 ; SINGLE/DOUBLE SIDED DISKETTE FLAG
0B06 (0002) 1235 TKFBA: DS     2 ; ADDRESS OF TRACK FORMAT BUFFER
0B08 (0001) 1236 DENSFL: DS     1 ; FORMAT/DENSITY FLAG
0B09 (0002) 1237 CPMVER: DS     2 ; CP/M VERSION NUMBER
0B0B (0001) 1238 LSSTEP: DS     1 ; LAST STEP COMMAND (OUT/IN)
0B0C (0001) 1239 ERRFG: DS     1 ; CONTROLLER ERROR STATUS FLAG
0B0D (0001) 1240 SAVCOM: DS     1 ; CONTROLLER CONNAND STORAGE DURING SIDE 1 F1
1241 ; CHECK
0B0E (0030) 1242 DS      030H ; FORMAT1 STACK SPACE
1243 STACK: ; START OF FORMAT1 STACK
0B3E (0400) 1244 READBF: DS    0400H ; RESERVE 1024 BYTES FOR THE LARGEST READ
1245 ; BUFFER REQUIRED
0F3E (1800) 1246 SDBUFF: DS    01800H ; RESERVE 6144 BYTES FOR THE SINGLE DENSITY
1247 ; FORMAT TRACK BUFFER
273E (2C00) 1248 DDBUFF: DS    02C00H ; RESERVE 11264 BYTES FOR THE DOUBLE DENSITY
1249 ; FORMAT TRACK BUFFER
533E (0000) 1250 END

```

```

Errors 0
Range Count 35

```

Symbol	Value	Defn	References
ACTBFA	0AFD	1228	0259 0286 0309 0315 0337 0372 0385 0412 0757
ATSKMG	09F0	1120	0261 0317
BDDTKB	068F	0902	0233
BDDTKL	06BE	0924	0969
BDOS	0005	0029	0085 1125 1131 1143 1198
BINDEC	0AA8	1168	0432 0445 0451
BMDTKB	0731	1000	0904
BMDTKL	0759	1019	1066
BSDTKB	0614	0825	0232
BSDTKL	0635	0842	0875
BUSYM	0001	0059	0622 0640
CKDS	0564	0690	0242
CKDSL	056A	0693	0695
CKESC	01AE	0122	0230 0264 0305 0319
CKSTAT	052C	0639	0573 0609 0641
CMSBYL	0517	0621	0623
CMD	0064	0023	0562 0600 0628 0782
CONBFC	0A25	1159	1140 1148 1151
CONIBF	0A27	1161	1136 1141 1144
CONIN	0ADC	1214	
CONOUT	0ADF	1215	
CONST	0AD9	1213	1203



CPMVER	0B09	1237	0086																	
CR	000D	0014	0081	0100	0112	0139	0157	0183	1095	1096	1096	1097	1098	1099	1099	1101	1101	1105		
	1106	1107	1107	1114	1116															
					1116	1117														
CRLFMG	07CA	1095	0221																	
CSECT	0B02	1232	0369	0383	0410	0449	0557													
CTRACK	0B03	1233																		
CVTMSG	0AB8	1183	1169																	
CVTMSL	0ABA	1185	1187																	
DATA	0067	0022	0563	0601	0736															
DCONIO	09F2	1124	0123																	
DCONO	09F7	1130																		
DDBUFF	273E	1248	0313	0905	0906	0982	1000	1001	1078											
DDCUS	0919	1114	0142																	
DENSB	0006	0072	0670	0675	0681															
DENSFL	0B08	1236	0093	0172	0205	0466	1075													
DESEL	00FF	0041	0712																	
DNCVL	05FD	0801	0802																	
DNRDYM	FF80	0055	0711	0819																
DNTODS	05F5	0795	0663																	
DONDCU	0907	1113	0214																	
DOUBCU	08F6	1111	0212																	
DRIVFT	0AFF	1229	0144	0662	0696															
DRIVM	000F	0068	0691																	
DSCPL	0600	0803	0800																	
DWAITM	FF80	0067	0625	0643																
EBITM	0004	0063	0789																	
EDENCU	080C	1098	0152																	
EDRVCU	07EC	1097	0134																	
EFMT	0371	0347	0300																	
ERRFG	0B0C	1239	0478	0502	0523	0654														
ERRRD	0420	0442	0270	0290	0324	0341	0376	0389	0416											
ERRWTK	0408	0429	0267	0288	0322	0339														
ESC	001B	0016	0110	0124	0137	0155	0181													
EWAITM	007F	0066	0067	0771																
F1B	0001	0074	0585																	
FEMSG	088D	1105	0434																	
FILECU	08E1	1109	0203																	
FORDA	032B	0317	0312																	
FORDSK	0276	0229	0104																	
FORINT	00D0	0038	0780																	
FORMAT	0157	0083	0080																	
FORMAT1	0177	0095	0113																	
FORNT1	036E	0345	0327																	
FORNK1	02FC	0295	0346																	
FORNXT	02FB	0294	0273																	
FORTKM	00F0	0045	0607																	
FORTRK	00F4	0040	0595																	
FWTK	04F7	0591	0480																	
FWTKLP	050A	0603	0606																	
GETDEN	0447	0461	0311	0354	0672	0760	0902	0979												
GETDNR	0452	0467	0464																	
GSDSEL	0561	0682	0680																	
GTDDTK	0557	0676	0674																	
GTDRE	01D1	0142	0140																	
Symbol	Value	Defn	References																	
GTDRIV	01BC	0134	0094	0146																
GTDESC2	0544	0662	0699																	
GTDESCM	054C	0670	0721																	
GTFCMX	01F4	0161	0158																	
GTFMAT	01DF	0152	0091	0170																
GTFSTD	0209	0171	0163	0166																
GTNBRT	020E	0176	0090	0191																
GTNTRK	0228	0186	0184																	
GTOPRP	09FC	1136	0097	0109	0136	0154	0180													
GTRK2	0237	0192	0187																	
HDNTLM	0020	0058	0786																	
HOME	0AE2	1216																		
ISELDS	0586	0711	0707																	
ISELS	0576	0701	0710																	
LF	000A	0015	0081	1095	1096	1096	1097	1098	1099	1099	1101	1101	1105	1106	1107	1107	1114	1116		
	1116	1117																		
LISTST	0AF7	1223																		
LKBIOS	0AC8	1202	0087																	
LKBISL	0AD0	1206	1211																	
LSSTEP	0B0B	1238	0257	0537	0545	0733														
MAXCU	08FE	1112	0210																	
MCL	0103	0081	0088																	
MSGCON	0AC2	1195	0089	0103	0107	0135	0153	0179	0198	0204	0213	0215	0222	0262	0318	0435	0437	0448		



