

Indus GT Firmware ROM Disassembly

```
;?-----\
;| DISASSEMBLED FILE - DONE WITH DISASM 1.0· |
;| Date: 1-06-2008 Time: 17:21 |
;| (c) 1996 Channex aka Lasse S. Tassing |
;| Email: ltassing.ite.dk |
; \-----/
; IN $0,$1 = audio on Sio Buss
; IN $2,$3 = index pulse enable
; IN $4,$5 = Transmit data on Txd sio buss output inverted
; IN $6,$7 = Transmit data on Rxd sio Buss output inverted
; IN $8,$9 = DDEN line on FDC High /Low
; IN $A,$B = Motor OFF/ON
; IN $C,$D = Generate Index pulse
; IN $E,$F = Enable/disable Ramcharger ram from $0000 to $7FFF
              Disable all hardware when ram is enabled

STAT1R = $1000                ;Read only. Clear latches
                                ;Bit0 = drive select switch 1
                                ;Bit1 = drive select switch 2
                                ;Bit2 = density select switch. set at boot up
                                ;Bit3 = n/c
                                ;Bit4 = Track button
                                ;Bit5 = ID button
                                ;Bit6 = Error button
                                ;Bit7 = Write protect from floppy mech
STAT1 = $1001                ;As above, doesn't clear latches
STAT2 = $2000                ;Read only
                                ;Bit0 = Clock out on sio buss
                                ;Bit1 = Clock in on sio buss
                                ;Bit2 = Data out on sio buss
                                ;Bit3 = Data in on sio buss
                                ;Bit4 = +5V on sio buss
                                ;bit5 = command on sio buss
                                ;Bit6 = DRQ from FDC
                                ;Bit7 = IRQ from FDC
CONTROL = $3000              ;First 4 bits. control for stepper motor. writ
LED1 = $4FFF                 ;Bits0-6 Led display
                                ;Bit7 = Busy Led
LED2 = $5000                 ;Bits0-6 Led display
                                ;Bit7 = ENPRE on FDC
COMMANDFDC = $6000          ;Command register FDC write
STATUSFDC = $6000           ;Status register FDC read
TRACKFDC = $6001            ;Track register FDC read/write
SECTORFDC = $6002          ;Sector register FDC read/write
DATAFDC = $6003             ;Data register FDC read/write
                                ;7000 7fff ram 7800-7fff used
RINVEC = $7800              ;Ram interupt vector 3 bytes
Lb93 = $7803                ;read sector 3 bytes
Lb22 = $7806                ;external main wait loop tie in 3bytes
Lb30 = $7809                ;external command tie in 3 bytes
RESTORE = $780C             ;Restore command FDC
READSECT = $780D            ;Read sector command FDC
WRITESECT = $780E           ;Write sector command FDC
READADDR = $780F            ;Read address command FDC
WRITETRK = $7810            ;Write track command FDC
INTENB = $7811              ;enable major interupt routine
UNUSED1 = $7812             ;contains $FB. not used
```

```

UNUSED2 = $7813           ;Contains $FB. not used
LSPEEDF = $7814           ;= FF if high bit is set on command
HSFLAG = $7815           ;= FF once lowspeed sio is complete then does highspee
FLAG?? = $7816           ;?? some thing to do with density flag
DENFLG = $7817           ;density flag 0=Single,1=Double,2=enhanced
MOTORF = $7818           ;Motor on off flag
DCHNGFLG = $7819         ;disk change flag?
WPMOFLG = $781A         ;Motor on when disk inserted flag
DENCST = $781B          ;Check density of disk flag TRACK and ID butto
CPML0D = $781C          ;CP/M load flag. Buttons ERROR and ID
FMTTOT = $781D          ;Format time out value. used for status comman
STATUS0 = $781E         ;Byte zero of status returned to consol
FDCSTA = $781F          ;FDC status register copy
CAUX12 = $7820          ;Store for command frame aux bytes 1 & 2
TRACKNUM = $7822         ;track number on disk to read/write
SECTORN = $7823         ;Sector number to read on track
SECTORBUF = $7824       ;Pointer to sector buffer 7842 2 bytes
FDCMSR = $7826          ;FDC master status register?
CDLOOP1 = $7827         ;Countdown loop 1 byte
CDLOOP2 = $7828         ; Countdownloop 1 byte
FMTstacksave = $7829    ;format stack save?? 2 bytes
FMTSTR = $782B          ; format store. 00= single, 7F= enhanced,FF=do
SECTAB = $782C          ;pointer to sector table. format time 2bytes
BASECD = $782E          ;base store for a count up timer. 2 bytes
BASECDSCRATCH = $7830   ;Scratch used for 782E 2 bytes
MSPINDEL = $7832        ;Counter for motor spin delay? 2 bytes
STAT1S = $7836         ;Store for stat1 current state
LEDERROR = $7838        ;Error store for front leds 2 bytes
TLEDNUM1 = $783A        ;Track led number store 1
TLEDNUM2 = $783B        ;track lebn number store 2
DENLET = $783C          ;led display for density A,b,C
DNMLED = $783D          ;Drive number led pattern store
BUSYLD = $783F          ;Busy led status
FREERAM = $7840         ;pointer to first byte of free ram
SECTBUF = $7842         ;256 byte sector buffer 256 bytes
BUFFER = $7942          ;General purpos Buffer 256 bytes
STACKSAVE = $7A42       ;Stack pointer Save word
STACK = $7A84           ;CPU stack for main program loop 16 bytes. cou
STACKI = $7B84          ;CPU stack for interupt (commands) 16 bytes
Lb38 = $7F00            ;Where external commands are stored and run
; ?[CODE]????????????????????????????????????????????????????????????
; LABEL INSTR.  PARAMETER(s)          ADR/OPCODE  ASCII
      ORG      0000H
      DI              ; 0000 F3      reset vector
      JP      Lb0     ; 0001 C33B0D   jump to test routine
Lba:   JP      Lb1     ; 0004 C3A10C   legal vector
      NOP              ; 0007 00      .
; -----
; high speed recieve
Lb2:   LD      A, (HL) ; 0008 7E      7  restart 08
      RRCA          ; 0009 0F      4
      JR      NC, Lb2 ; 000A 30FC   7/12
Lb3:   LD      A, (HL) ; 000C 7E      7
      RRCA          ; 000D 0F      4
      JR      C, Lb3  ; 000E 38FC   7/12
      AND     D       ; 0010 A2      4
      RET          ; 0011 C9      10 50 t states with no branching
; -----
;delay

```

```

Lb4:   DEC      BC           ; 0012 0B      .
       LD      A,B         ; 0013 78      x
       OR      C           ; 0014 B1      .
       JR      NZ,Lb4      ; 0015 20FB    .
       RET                      ; 0017 C9      .
;-----
; high speed send
Lb5:   LD      A,(HL)      ; 0018 7E      7  load
       AND     E           ; 0019 A3      4  mask out all but clock out
       JR      NC,Lb5      ; 001A 30FC    7/12
Lb6:   LD      A,(HL)      ; 001C 7E      7  get sio signals
       AND     E           ; 001D A3      4  mask out all but clock out
       JR      C,Lb6       ; 001E 38FC    7/12
       IN     A,(C)        ; 0020 ED78    12
       RET                      ; 0022 C9      10  58 t states with no branchin

LDTBL                                ;Led display table bit patterns. inverted.
                                       ;Bit 0 = top bar, bit 1= top right
                                       ;bit 2 = bottom right. bit 3 = bottom
                                       ;bit 4 = bottom left. bit 5 = top left
                                       ;bit 6 = middle bar
DB $C0                                ; 0023 C0      0
DB $F9                                ; 0024 F9      1
DB $A4                                ; 0025 A4      2
DB $B0                                ; 0026 B0      3
DB $99                                ; 0027 99      4
DB $92                                ; 0028 92      5
DB $82                                ; 0029 82      6
DB $F8                                ; 002A F8      7
DB $80                                ; 002B 80      8
DB $98                                ; 002C 98      9
DB $88                                ; 002D 88      A
DB $83                                ; 002E 83      b
DB $C6                                ; 002F C6      C
DB $A1                                ; 0030 A1      d
DB $86                                ; 0031 86      E
DB $8E                                ; 0032 8E      F
                                       ; Commands for 1770 FDC
DB $ff                                ; 0033 FF  Restore?? wrong number
DB $80                                ; 0034 80  read sector, dis spin up seq
DB $A2                                ; 0035 A2  write sector, dis write precomp
DB $C0                                ; 0036 C0  read address
DB $F2                                ; 0037 f2  write track,dis write precomp

;processor interupt starts here
INTCONT JP      RINVEC      ; 0038 C30078  Jump to ram interupt vector
        PUSH   AF          ; 003B F5      continues here. save A and F
        LD     A,(INTENB)  ; 003C 3A1178 ready to do interupt?
        OR     A           ; 003F B7      set flags
        JR     NZ,Lb8      ; 0040 203E    jump relative if not
        PUSH  BC          ; 0042 C5      save registers
        PUSH  DE          ; 0043 D5      onto stack
        PUSH  HL          ; 0044 E5      .
        EX   AF,A'F'      ; 0045 08      .
        PUSH  AF          ; 0046 F5      .
        EXX                     ; 0047 D9      and 2ndery registers
        PUSH  BC          ; 0048 C5      .
        PUSH  DE          ; 0049 D5      .
        PUSH  HL          ; 004A E5      .

```

```

PUSH IX ; 004B DDE5 ..
PUSH IY ; 004D FDE5 ..
LD HL, (STACKSAVE) ; 004F 2A427A temp stack pointer?
LD A,L ; 0052 7D }
OR H ; 0053 B4 .
JR NZ,Lb9 ; 0054 2004 .
ADD HL,SP ; 0056 39 9
LD (STACKSAVE),HL ; 0057 22427A save stack to 7a42
Lb9: LD SP,STACKI ; 005A 31847B set stack to 7b84
EI ; 005D FB .
CALL Lb10 ; 005E CD6D01 get command frame and excaute
XOR A ; 0061 AF .
LD (BUSYLD),A ; 0062 323F78 2?x
CALL UPLED-1-2-B-E ; 0065 CD120B clear busy led?
DI ; 0068 F3 .
LD HL, (STACKSAVE) ; 0069 2A427A recover stack pointer
LD SP,HL ; 006C F9 .put back in to SP
LD HL,$00 ; 006D 210000 zero out
LD (STACKSAVE),HL ; 0070 22427A temp stack pointer
POP IY ; 0073 FDE1 ..
POP IX ; 0075 DDE1 ..
POP HL ; 0077 E1 .
POP DE ; 0078 D1 .
POP BC ; 0079 C1 .
EXX ; 007A D9 .
POP AF ; 007B F1 .
EX AF,A'F' ; 007C 08 .
POP HL ; 007D E1 .
POP DE ; 007E D1 .
POP BC ; 007F C1 .
Lb8: POP AF ; 0080 F1 .
EI ; 0081 FB .
RETI ; 0082 ED4D .M
;Init
Lb236: IM 1 ; 0084 ED56 interupt mode 1
IN A,($03) ; 0086 DB03 ..index
IN A,($04) ; 0088 DB04 ..TXD high
IN A,($06) ; 008A DB06 ..RXD high
IN A,($0A) ; 008C DB0A ..motor off
IN A,($0C) ; 008E DB0C ..ip
LD SP,STACK ; 0090 31847A 1.z stack pointer
LD A,$C3 ; 0093 3EC3 >. set
LD (RINVEC),A ; 0095 320078 2.x $7800
LD HL,INTCONT ; 0098 213B00 !;. to
LD ($7801),HL ; 009B 220178 ".x JP ($003B)
LD ($7803),A ; 009E 320378 2.x set 7803
LD HL,Lb169 ; 00A1 21A00C !.. JP ($0CA0) RET
LD ($7804),HL ; 00A4 220478 ".x set 7806
LD ($7806),A ; 00A7 320678 2.x to JP($0CA0) RET
LD ($7807),HL ; 00AA 220778 ".x set 7809
LD ($7809),A ; 00AD 320978 2.x to JP($0CA0) RET
LD ($780A),HL ; 00B0 220A78 ".x
LD A,$00 ; 00B3 3E00 >.
LD ($7837),A ; 00B5 323778 27x
LD A,(STAT1R) ; 00B8 3A0010 :..
LD A,(STAT1) ; 00BB 3A0110 :..
LD (STAT1S),A ; 00BE 323678 26x
AND $04 ; 00C1 E604 .. density switch
ADD A,$FF ; 00C3 C6FF .. -1

```



```

LD      A, (DENCST)          ; 015B 3A1B78   :.x
OR      A                    ; 015E B7      . set flags
JR      Z, Lb25              ; 015F 2807   (.
CALL    Lb19                 ; 0161 CD6D05 go check density of disk
XOR     A                    ; 0164 AF      . zero A
LD      (DENCST), A         ; 0165 321B78 2.x
Lb25:   CALL    Lb26         ; 0168 CDD401 ...
JR      Lb27                 ; 016B 18E5   .. end wait loop
;-----
Lb10:   IN      A, ($04)     ; 016D DB04   ..TXD high
XOR     A                    ; 016F AF      zero A
LD      ($7815), A         ; 0170 321578 2.x
CALL    Lb28                 ; 0173 CD1802 ... check for command signal
RET     NZ                   ; 0176 C0      .
CALL    Lb29                 ; 0177 CDFD01 ... call get command frame
RET     NZ                   ; 017A C0      . get out if drive num don't m
; jump to command routine-----
LD      (CAUX12), DE        ; 017B ED532078 de=command frame aux bytes
LD      A, $FF              ; 017F 3EFF    >.b= device num c=command
LD      (BUSYLD), A        ; 0181 323F78 2?x
CALL    UPLED-1-2-B-E      ; 0184 CD120B ...? set busy led?
CALL    Lb30                 ; 0187 CD0978 call external command routine
LD      A, C                 ; 018A 79      y
RLCA                    ; 018B 07      .
SBC     A, A                 ; 018C 9F      .
LD      ($7814), A         ; 018D 321478 2.x
LD      A, C                 ; 0190 79      y
AND     $7F                 ; 0191 E67F    remove high bit
LD      C, A                 ; 0193 4F      0
LD      B, $09              ; 0194 0609    number of commands
LD      HL, $1B9            ; 0196 21B901 pointer to start of cmd table
Lb32:   LD      A, (HL)      ; 0199 7E      load command
INC     HL                   ; 019A 23      inc pointer
CP      C                    ; 019B B9      command match?
JR      Z, Lb31              ; 019C 2806    get out if equal
INC     HL                   ; 019E 23      inc pointer
INC     HL                   ; 019F 23      inc pointer
DJNZ   Lb32                 ; 01A0 10F7    Dec B and loop if not zero
JR      Lb33                 ; 01A2 1808    no commands matched. go nend N
Lb31:   LD      A, (HL)      ; 01A4 7E      load HL with jump address
INC     HL                   ; 01A5 23      .
LD      H, (HL)             ; 01A6 66      .
LD      L, A                 ; 01A7 6F      .
CALL    Lb34                 ; 01A8 CDB801 call command
RET     Z                    ; 01AB C8      .
Lb33:   LD      BC, $82      ; 01AC 018200 delay number
CALL    Lb4                  ; 01AF CD1200 jump to delay
LD      A, $4E              ; 01B2 3E4E    load 'N'AK
CALL    SSIOWBYTE           ; 01B4 CD230A go send byte sio
RET     ; 01B7 C9      .

Lb34:   JP      (HL)         ; 01B8 E9      jump command

DB $4E                    ; 01B9 4E      command N
WORD GETBLOCK              ; 01BA 0103
DB $4F                    ; 01BA 4f      command o
WORD PUTBLOCK              ; 01BD 4A03
DB $50                    ; 01BF 50      Put
WORD PUT                   ; 01C0 8D02

```

```

DB $52 ; 01C2 52 read
WORD READ ; 01C3 7902
DB $53 ; 01C5 53 S status
WORD STATUS ; 01C6 2502
DB $57 ; 01C8 57 W write
WORD WRITE ; 01C9 9602
DB $58 ; 01CB 58 command X
WORD EXTERNALCMD ; 01CC D802
DB $21 ; 01CE 21 format
WORD FORMAT ; 01CF 8B03
DB $22 ; 01D1 22 format enhanced
WORD FORMATENH ; 01D2 7E03
;-----
;CP/M LOAD button status check
Lb26: LD A,(CPML0D) ; 01D4 3A1C78 :.x
OR A ; 01D7 B7 set flags
RET Z ; 01D8 C8 .
;load CP/M-----
CALL Lb36 ; 01D9 CD6005 check if disk changed and test
LD HL,(SECTORBUF) ; 01DC 2A2478 save buffer
PUSH HL ; 01DF E5 .
LD HL,$7F00 ; 01E0 21007F place to put data
LD (SECTORBUF),HL ; 01E3 222478 "$x
LD HL,$01 ; 01E6 210100 sector to load
LD (CAUX12),HL ; 01E9 222078 " x
CALL Lb37 ; 01EC CD9504 go get sector 1
POP HL ; 01EF E1 .
LD (SECTORBUF),HL ; 01F0 222478 "$x
CP $43 ; 01F3 FE43 .C complete
CALL Z,Lb38 ; 01F5 CC007F ... call $7f00
XOR A ; 01F8 AF .
LD (CPML0D),A ; 01F9 321C78 2.x
RET ; 01FC C9 .
;-----
; Command frame recieve, test and run
Lb29: LD HL,BUFFER ; 01FD 214279 place to put command frame
LD B,$04 ; 0200 0604 num of bytes to get?
CALL Lb39 ; 0202 CD7C09 get command frame
CALL NZ,Lb40 ; 0205 C49A0A ... ??
CALL DNUMSWITCH ; 0208 CDCE0B ... get drive number switch
LD BC,BUFFER ; 020B ED4B4279 device/command
LD DE,($7944) ; 020F ED5B4479 aux bytes 1/2
LD L,B ; 0213 68 h
LD B,C ; 0214 41 A b= device number
LD C,L ; 0215 4D M c= command
CP B ; 0216 B8 . compare command with device
RET ; 0217 C9 .

Lb28: LD B,$0A ; 0218 060A ..
Lb41: LD A,(STAT2) ; 021A 3A0020 :. valid
AND $30 ; 021D E630 .0 command signal?
CP $10 ; 021F FE10 ..
RET NZ ; 0221 C0 .

DJNZ Lb41 ; 0222 10F6 ..
RET ; 0224 C9 .

STATUS
CALL Lb42 ; 0225 CD0F0A Send Ack
CALL Lb36 ; 0228 CD6005 .`.

```

	CALL	Lb14	; 022B CD3509	.5.
	LD	A, (STATUSFDC)	; 022E 3A0060	:.`
	AND	\$40	; 0231 E640	keep bit 6 of fdc status regis
	ADD	A, \$FF	; 0233 C6FF	3F plus carry
	SBC	A, A	; 0235 9F	either 0 or FF depending on ca
	AND	\$08	; 0236 E608	keep bit 3
	LD	C, A	; 0238 4F	0
	LD	A, (DENFLG)	; 0239 3A1778	:.x
	OR	A	; 023C B7	set flags
	JR	Z, Lb43	; 023D 2805	(.
	ADD	A, \$FE	; 023F C6FE	..
	RRA		; 0241 1F	.
	AND	\$A0	; 0242 E6A0	only get bits 7 and 5
Lb43:	LD	B, A	; 0244 47	G
	LD	A, (STATUS0)	; 0245 3A1E78	:.x
	AND	\$57	; 0248 E657	Bits 6,4,2,1,0
	OR	B	; 024A B0	.
	OR	C	; 024B B1	.
	LD	HL, BUFFER	; 024C 214279	!By
	LD	(HL), A	; 024F 77	w
	INC	HL	; 0250 23	#
	CALL	SETDENFLG	; 0251 CDF20B	...
	LD	A, (\$FDCSTA)	; 0254 3A1F78	load FDC status ram register
	CPL		; 0257 2F	invert
	LD	(HL), A	; 0258 77	store in buffer
	INC	HL	; 0259 23	point to next byte
	LD	A, (\$FMTTOT)	; 025A 3A1D78	load format time out value
	LD	(HL), A	; 025D 77	store in buffer
	INC	HL	; 025E 23	point to next byte
	LD	A, (\$7826)	; 025F 3A2678	:&x
	SRL	A	; 0262 CB3F	.?
	LD	(HL), A	; 0264 77	w
	LD	A, \$43	; 0265 3E43	>C complete
	LD	HL, BUFFER	; 0267 214279	buffer in HL
	LD	B, \$04	; 026A 0604	number of bytes
	CALL	Lb44	; 026C CDAC09	go send
Lb12a:	LD	A, (STATUS0)	; 026F 3A1E78	:.x
	AND	\$18	; 0272 E618	..
	LD	(STATUS0), A	; 0274 321E78	2.x
	XOR	A	; 0277 AF	.
	RET		; 0278 C9	.
READ				
	CALL	CHKSECTNUM	; 0279 CD3C0C	Check sector will be on disk
	JP	NZ, Lb40	; 027C C29A0A	...
	CALL	Lb42	; 027F CD0F0A	send Ack
	CALL	Lb36	; 0282 CD6005	.
	CALL	Lb37	; 0285 CD9504	go read sector. A=C or E
	CALL	Lb46	; 0288 CDA109	go send A then data + chksum
	XOR	A	; 028B AF	.
	RET		; 028C C9	.
PUT				
	CALL	Lb47	; 028D CDA504	...
	RET	NZ	; 0290 C0	Nak out
	CALL	SSIOBYTE	; 0291 CD230A	send byte sio
	XOR	A	; 0294 AF	.
	RET		; 0295 C9	.
WRITE				
	CALL	Lb47	; 0296 CDA504	get data and write sector
	RET	NZ	; 0299 C0	.


```

CP      $45                ; 029A FE45      did it error out?
JR      Z,Lb48             ; 029C 2835      yes
CALL    DATAINVERT       ; 029E CD890A    no. invert data in buffer
CALL    Lb50               ; 02A1 CDA60A    check is sector is 4 or greater
LD      C,B               ; 02A4 48        H
LD      B,$00             ; 02A5 0600      ..
DEC     C                 ; 02A7 0D        .
INC     BC                ; 02A8 03        .
LD      HL,(SECTORBUF)    ; 02A9 2A2478    *$x
LD      DE,BUFFER        ; 02AC 114279    .By
LDIR                               ; 02AF EDB0      ..
CALL    Lb37              ; 02B1 CD9504    go read sector
CP      $45                ; 02B4 FE45      .E
JR      Z,Lb51             ; 02B6 2813      (.
CALL    Lb50               ; 02B8 CDA60A    check if sector is 4 or greater
LD      HL,(SECTORBUF)    ; 02BB 2A2478    *$x
LD      DE,BUFFER        ; 02BE 114279    .By
Lb52:  LD      A,(DE)      ; 02C1 1A        this cannot work properly. nei
CP      (HL)              ; 02C2 BE        are incremented. only 1st byte
JR      NZ,Lb51           ; 02C3 2006      checked 128/256 times for writ
DJNZ   Lb52               ; 02C5 10FA      dec B register and jump if not
LD      A,$43             ; 02C7 3E43      >C complete
JR      Lb48              ; 02C9 1808      ..
Lb51:  LD      HL,$8C98    ; 02CB 21988C    p9
CALL    UPLED-1-2-B-E     ; 02CE CD120B    Send P9 to front led
LD      A,$45             ; 02D1 3E45      >E error
Lb48:  CALL    SSIobyte    ; 02D3 CD230A    send byte sio
XOR    A                  ; 02D6 AF        .
RET                               ; 02D7 C9        .

EXTERNALCMD:                ;          command X
CALL    Lb42               ; 02D8 CD0F0A    send Ack
LD      A,($7821)         ; 02DB 3A2178    daux2
AND    $01                ; 02DE E601      aux 2 = 0?
JR      Z,Lb53             ; 02E0 2817      go straight to external routin
LD      HL,$7F00          ; 02E2 21007F    buffer to put data
LD      A,(CAUX12)        ; 02E5 3A2078    number of bytes to load Daux1
LD      B,A               ; 02E8 47        .
CALL    Lb39              ; 02E9 CD7C09    get bytes
JP      NZ,Lb54           ; 02EC C2A00A    .
CALL    Lb55               ; 02EF CD1B0A    send Ack
LD      A,$43             ; 02F2 3E43      'C'omplete
CALL    SSIobyte          ; 02F4 CD230A    send byte sio
JR      Lb56              ; 02F7 1806      jump relative always
Lb53:  CALL    Lb38         ; 02F9 CD007F    call external command
CALL    C,SSIobyte        ; 02FC DC230A    send byte sio
Lb56:  XOR    A            ; 02FF AF        zero A
RET                               ; 0300 C9        .

GETBLOCK
CALL    Lb42               ; 0301 CD0F0A    send Ack
CALL    Lb36               ; 0304 CD6005    .`.
LD      HL,PCB            ; 0307 213E03    pointer to rom control block.
LD      DE,BUFFER        ; 030A 114279    pointer to ram buffer
LD      BC,$0C           ; 030D 010C00    number of bytes to move
LDIR                               ; 0310 EDB0      move bytes rom to ram
LD      HL,$7945          ; 0312 214579    start with sect per track low
LD      A,(DENFLG)       ; 0315 3A1778    default to 18
CP      $02               ; 0318 FE02      check for double..

```

```

Lb57: JR      NZ,Lb57          ; 031A 2002    no .
      LD      (HL), $1A      ; 031C 361A    put 26 into sect per track
      INC     HL              ; 031E 23      set pointer to
      INC     HL              ; 031F 23      record method
      ADD     A, $FF         ; 0320 C6FF    ..
      SBC     A, A           ; 0322 9F      .
      AND     $04            ; 0323 E604    ..
      LD      (HL), A        ; 0325 77      w
      INC     HL              ; 0326 23      #
      CALL    Lb58           ; 0327 CDB40A  ...call num bytes per sector
      DEC     B              ; 032A 05      -1
      LD      C, B           ; 032B 48      H
      LD      B, $00         ; 032C 0600    set high byte to zero
      INC     BC              ; 032E 03      add 1
      LD      (HL), B        ; 032F 70      put sect num high in
      INC     HL              ; 0330 23      set pointer +1
      LD      (HL), C        ; 0331 71      put in low byte
      LD      A, $43         ; 0332 3E43    >C complete
      LD      HL, BUFFER     ; 0334 214279  buffer start to send
      LD      B, $0C         ; 0337 060C    num of bytes to send..
      CALL    Lb44           ; 0339 CDAC09  send bytes
      XOR     A              ; 033C AF      .
      RET                        ; 033D C9      .

;percom control block
PCB
      DB $28                ; 033E 28      number of tracks
      DB $01                ; 033f 01      step rate
      DB $00                ; 0340 00      sectors per track high
      DB $12                ; 0341 12      sectors per track low
      DB $00                ; 0342 00      number of sides -1.
      DB $00                ; 0344 00      record method. 0=mf. 4=mfm
      DB $FF                ; 0346 FF      bytes per sector high.
      DB $00                ; 0347 00      bytes per sector low.
      DB $00                ; 0348 00      .
      DP $00                ; 0349 00      .

PUTBLOCK
      CALL    Lb42           ; 034A CD0F0A  send Ack
      LD      B, $0C         ; 034D 060C    number of bytes to get
      LD      HL, BUFFER     ; 034F 214279  place to put them
      CALL    Lb39           ; 0352 CD7C09  get and caculate checksum
      JP      NZ, Lb54       ; 0355 C2A00A  get out if chksum don't match
      CALL    Lb55           ; 0358 CD1B0A  send Ack?
      CALL    Lb36           ; 035B CD6005  .`.
      LD      A, ($7947)     ; 035E 3A4779  :Gy record method 0=fm 4=mfm
      OR      A              ; 0361 B7      . set flags
      JR      Z, Lb60        ; 0362 280A    (.
      LD      A, ($7948)     ; 0364 3A4879  bytes per sect high
      OR      A              ; 0367 B7      set flags
      LD      A, $02         ; 0368 3E02    2=enhanced
      JR      Z, Lb60        ; 036A 2802    (.
      LD      A, $01         ; 036C 3E01    1= double 0=single 2=enhanced
Lb60: LD      (DENFLG), A     ; 036E 321778  2.x
      LD      ($7816), A     ; 0371 321678  2.x
      CALL    SETDENFLG     ; 0374 CDF20B  set dden + front led store
      LD      A, $43         ; 0377 3E43    >C complete
      CALL    SSIobyte      ; 0379 CD230A  send byte sio
      XOR     A              ; 037C AF      set flags??
      RET                        ; 037D C9      .

```

FORMATENH

```

LD      A,$02                ; 037E 3E02    format enhanced
LD      (DENFLG),A          ; 0380 321778  2.x
LD      ($7816),A          ; 0383 321678  2.x
CALL    SETDENFLG          ; 0386 CDF20B  set den type for front leds
JR      Lb61                ; 0389 180F    ..

```

FORMAT

```

LD      HL,DENFLG          ; 038B 211778  format
LD      A,(HL)             ; 038E 7E      ~
SUB     $02                ; 038F D602    ..
JR      NZ,Lb61            ; 0391 2007    .
LD      (HL),A             ; 0393 77      w
LD      ($7816),A          ; 0394 321678  2.x
CALL    SETDENFLG          ; 0397 CDF20B  ...

```

Lb61:

```

CALL    Lb42                ; 039A CD0F0A  send Ack
LD      A,(DENFLG)         ; 039D 3A1778  :.x
PUSH    AF                  ; 03A0 F5      .
CALL    Lb36                ; 03A1 CD6005  .`.
POP     AF                  ; 03A4 F1      .
LD      (DENFLG),A         ; 03A5 321778  2.x
LD      ($7816),A          ; 03A8 321678  2.x
CALL    SETDENFLG          ; 03AB CDF20B  ...
CALL    Lb15                ; 03AE CDBE08  turn motor on
LD      HL,$FFFF           ; 03B1 21FFFF  !..
LD      (CAUX12),HL        ; 03B4 222078  " x
CALL    Lb58                ; 03B7 CDB40A  set B to bytes per sector
XOR     A                  ; 03BA AF      .
LD      (TRACKNUM),A       ; 03BB 322278  2"x
DEC     A                  ; 03BE 3D      =
LD      HL,BUFFER          ; 03BF 214279  !By

```

Lb62:

```

LD      (HL),A             ; 03C2 77      put FF in buffer
INC     HL                  ; 03C3 23      #
DJNZ   Lb62                ; 03C4 10FC    fill buffer with FF

```

Lb72:

```

CALL    Lb63                ; 03C6 CDF407  step to track
LD      A,$05               ; 03C9 3E05    times to try
LD      (CDLOOP2),A        ; 03CB 322878  2(x

```

Lb67:

```

CALL    Lb23                ; 03CE CD440B  buttons
CALL    Lb64                ; 03D1 CD1306  format
LD      ($FDCSTA),A        ; 03D4 321F78  2.x
AND     $44                 ; 03D7 E644    .D
JR      Z,Lb65              ; 03D9 281B    (.
LD      B,A                 ; 03DB 47      G
AND     $40                 ; 03DC E640    .@
LD      A,B                 ; 03DE 78      x
JR      NZ,Lb66             ; 03DF 2006    .
LD      HL,CDLOOP2         ; 03E1 212878  !(x
DEC     (HL)                ; 03E4 35      5
JR      NZ,Lb67            ; 03E5 20E7    .

```

Lb66:

```

CALL    Lb68                ; 03E7 CDF60A  ...
LD      H,$8E               ; 03EA 268E    &.
CALL    Lb69                ; 03EC CD050B  ...
LD      HL,STATUS0         ; 03EF 211E78  !.x
SET     2,(HL)              ; 03F2 CBD6    ..
JR      Lb70                ; 03F4 1818    ..

```

Lb65:

```

LD      HL,TRACKNUM        ; 03F6 212278  !"x
LD      A,(HL)             ; 03F9 7E      ~
CP      $4E                 ; 03FA FE4E    78 exceded FDC track counter?

```

```

JR      Z,Lb71          ; 03FC 2804  (
INC     (HL)           ; 03FE 34    4
INC     (HL)           ; 03FF 34    4
Lb71:  JR      Lb72          ; 0400 18C4  ..
CALL   Lb23           ; 0402 CD440B buttons
CALL   Lb73           ; 0405 CD1504 go check format
JR      NZ,Lb70        ; 0408 2004  .
LD     A,$43          ; 040A 3E43  Complete
JR      Lb74          ; 040C 1802  ..
Lb70:  LD     A,$45          ; 040E 3E45  Error
Lb74:  CALL   Lb75           ; 0410 CD9709 ...
XOR    A              ; 0413 AF    .
RET                               ; 0414 C9    .
;-----
Lb73:  CALL   Lb58           ; 0415 CDB40A load B with sector size
LD     A,B            ; 0418 78    x
SRL    A              ; 0419 CB3F  .?
DEC    A              ; 041B 3D    =
LD     ($7829),A      ; 041C 322978 2)x
XOR    A              ; 041F AF    zero A
LD     (TRACKNUM),A  ; 0420 322278 2"x
LD     IY,BUFFER     ; 0423 FD214279 .!By
Lb86:  CALL   Lb63           ; 0427 CDF407 step to track 0
CALL   Lb76           ; 042A CDB807 write track number to FDC
LD     IX,$736        ; 042D DD213607 sector order single/double
LD     A,(DENFLG)    ; 0431 3A1778 :.x
CP     $02            ; 0434 FE02  ..
JR      NZ,Lb77        ; 0436 2004  .
LD     IX,$79D        ; 0438 DD219D07 sector order enhanced
;format track
Lb77:  LD     A,(IX+$00)   ; 043C DD7E00 get sector order
LD     (SECTORFDC),A ; 043F 320260 write sector order to FDC
LD     A,$05          ; 0442 3E05  >.
LD     (CDLOOP2),A   ; 0444 322878 2(x
Lb83:  LD     HL,(SECTORBUF) ; 0447 2A2478 *$x
CALL   Lb78           ; 044A CDD505 ...
JR      Z,Lb79        ; 044D 2805  (
CALL   Lb80           ; 044F CD6809 .h.
JR      Lb81          ; 0452 1805  ..
Lb79:  CALL   Lb14          ; 0454 CD3509 .5.
OR     $10            ; 0457 F610  ..
Lb81:  AND    $1C          ; 0459 E61C  ..
JR      Z,Lb82        ; 045B 281F  (
LD     HL,CDLOOP2    ; 045D 212878 !(x
DEC    (HL)           ; 0460 35    5
JR      NZ,Lb83        ; 0461 20E4  .
CALL   Lb84           ; 0463 CD7F0C ...
LD     (IY+$00),L    ; 0466 FD7500 .u.
INC    IY             ; 0469 FD23  .#
LD     (IY+$00),H    ; 046B FD7400 .t.
INC    IY             ; 046E FD23  .#
LD     HL,$8E98      ; 0470 21988E !..
CALL   Lb69           ; 0473 CD050B ...
LD     HL,$7829      ; 0476 212978 !)x
DEC    (HL)           ; 0479 35    5
JR      Z,Lb85        ; 047A 2812  (
Lb82:  INC    IX           ; 047C DD23  .#
LD     A,(IX+$00)    ; 047E DD7E00 .~.
RLCA                               ; 0481 07    .

```

```

JR      NC,Lb77      ; 0482 30B8      0.

LD      HL,TRACKNUM ; 0484 212278    !"x
INC     (HL)        ; 0487 34         4
INC     (HL)        ; 0488 34         4
LD      A,(HL)      ; 0489 7E         ~
CP      $4F         ; 048A FE4F       .0
JR      C,Lb86      ; 048C 3899       8.

Lb85:   LD      HL,(BUFFER) ; 048E 2A4279    *By
        INC     HL        ; 0491 23         #
        LD      A,H       ; 0492 7C         |
        OR      L         ; 0493 B5         .
        RET     ; 0494 C9         .

;-----
;this routine loads sector num stored in 7820 and stores to pointer located in 7824
Lb37:   CALL    TRKSECT    ; 0495 CD570C    .W.
        XOR     A         ; 0498 AF         zero A
        CALL   Lb88      ; 0499 CDDA04    ...
        LD      B,$43    ; 049C 0643     .C complete
        JR      Z,Lb89   ; 049E 2802     (.
        LD      B,$45    ; 04A0 0645     .E error
Lb89:   XOR     A         ; 04A2 AF         .
        LD      A,B       ; 04A3 78         x
        RET     ; 04A4 C9         .

Lb47:   CALL    CHKSECTNUM ; 04A5 CD3C0C    check sector will fit on disk
        JP      NZ,Lb40  ; 04A8 C29A0A    set status bit
        CALL   Lb42      ; 04AB CD0F0A    send Ack, set hs flag
        CALL   Lb90      ; 04AE CD7309    get data from sio
        JP      NZ,Lb54  ; 04B1 C2A00A    ...
        CALL   Lb55      ; 04B4 CD1B0A    send Ack
Lb62a:  CALL    Lb36      ; 04B7 CD6005    check den/disk spinning
        CALL    TRKSECT  ; 04BA CD570C    get track and sector for write
        LD      A,$FF    ; 04BD 3EFF     >.
        CALL   Lb88      ; 04BF CDDA04    setup and write sector
        LD      B,$43    ; 04C2 0643     .C
        JR      Z,Lb91   ; 04C4 2802     (.
        LD      B,$45    ; 04C6 0645     .E
Lb91:   XOR     A         ; 04C8 AF         .
        LD      A,B       ; 04C9 78         x
        RET     ; 04CA C9         .

XCMD2:           ;write sector
        LD      A,$FF    ; 04CB 3EFF     >.
        JR      Lb92     ; 04CD 1801     ..jump always

XCMD1:  XOR     A         ; 04CF AF         zero A
Lb92:   LD      C,A       ; 04D0 4F         0
        LD      A,D       ; 04D1 7A         z
        LD      (TRACKNUM),A ; 04D2 322278    2"x
        LD      A,E       ; 04D5 7B         {
        LD      (SECTORN),A ; 04D6 322378    2#x
        LD      A,C       ; 04D9 79         y

Lb88:   CALL    Lb93      ; 04DA CD0378    ..x jump to Ram vector
        OR      A         ; 04DD B7         set flags
        LD      A,$C2    ; 04DE 3EC2     >.
        JR      Z,Lb94   ; 04E0 2802     (.

```

```

Lb94: LD A,$8C ; 04E2 3E8C >.
      PUSH AF ; 04E4 F5 .
      CALL Lb15 ; 04E5 CDBE08 turn motor on
      LD A,$02 ; 04E8 3E02 >.
      LD (CDLOOP1),A ; 04EA 322778 do this loop twice
Lb103: CALL Lb63 ; 04ED CDF407 step to track NN
      LD A,$05 ; 04F0 3E05 >.
      LD (CDLOOP2),A ; 04F2 322878 2(x
Lb102: CALL Lb76 ; 04F5 CDB807 set FDC to track num
      LD A,(SECTORN) ; 04F8 3A2378 set FDC to Sectornum
      LD (SECTORFDC),A ; 04FB 320260 2.`
      LD HL,(SECTORBUF) ; 04FE 2A2478 *$x
      POP AF ; 0501 F1 .
      PUSH AF ; 0502 F5 .
      JR NZ,Lb95 ; 0503 200A .
      CALL Lb78 ; 0505 CDD505 read sector data
      CALL DATAINVERT ; 0508 CD890A invert data in buffer
      JR Z,Lb96 ; 050B 280C (.
      JR Lb97 ; 050D 1811 ..

Lb95: CALL DATAINVERT ; 050F CD890A invert data in buffer
      DI ; 0512 F3 .
      CALL Lb98 ; 0513 CDF405 write sector data
      EI ; 0516 FB .
      JR NZ,Lb97 ; 0517 2007 .
Lb96: CALL Lb14 ; 0519 CD3509 reset FDC?
      OR $10 ; 051C F610 ..
      JR Lb99 ; 051E 1803 ..
Lb97: CALL Lb80 ; 0520 CD6809 wait for FDC to be free?
Lb99: LD ($FDCSTA),A ; 0523 321F78 2.x
      AND $7C ; 0526 E67C fdc status
      JR Z,Lb100 ; 0528 282D (-
      LD B,A ; 052A 47 G
      AND $40 ; 052B E640 .@
      LD A,B ; 052D 78 x
      JR NZ,Lb101 ; 052E 2017 .
      LD HL,CDLOOP2 ; 0530 212878 !(x
      DEC (HL) ; 0533 35 5
      JR NZ,Lb102 ; 0534 20BF .
      LD B,A ; 0536 47 G
      AND $10 ; 0537 E610 ..
      LD A,B ; 0539 78 x
      JR Z,Lb101 ; 053A 280B (.
      LD HL,CDLOOP1 ; 053C 212778 !'x
      DEC (HL) ; 053F 35 5
      JR Z,Lb101 ; 0540 2805 (.
      CALL Lb16 ; 0542 CDC107 ...
      JR Lb103 ; 0545 18A6 ..

Lb101: CALL Lb68 ; 0547 CDF60A get error led number
      POP AF ; 054A F1 .
      LD H,A ; 054B 67 g
      CALL Lb69 ; 054C CD050B beep and display error
      LD HL,STATUS0 ; 054F 211E78 !.x
      SET 2,(HL) ; 0552 CBD6 ..
      OR $FF ; 0554 F6FF ..
      RET ; 0556 C9 .

Lb100: POP AF ; 0557 F1 .

```

```

RET      Z      ; 0558 C8      .

LD      HL,STATUS0 ; 0559 211E78    !.x
RES     3,(HL)    ; 055C CB9E      ..
XOR     A        ; 055E AF        .
RET     ; 055F C9      .

;-----
;Check if disk changed
Lb36:   LD      A,($7819) ; 0560 3A1978    :.x
        OR      A        ; 0563 B7        set flags
        RET     Z        ; 0564 C8        .
        CALL   Lb19      ; 0565 CD6D05    go set density
        XOR     A        ; 0568 AF        .
        LD      ($7819),A ; 0569 321978    2.x
        RET     ; 056C C9      .

;-----
;go read sector 1 and set density
Lb19:   CALL   Lb15      ; 056D CDBE08    turn on motor?
        CALL   Lb16      ; 0570 CDC107    step to track zero
        CALL   Lb104     ; 0573 CDAA05    ...
        JR     Z,Lb105   ; 0576 281B      (.
        LD     HL,DENFLG ; 0578 211778    !.x
        LD     A,(HL)    ; 057B 7E        ~
        ADD   A,$FF      ; 057C C6FF      .. -1
        CCF      ; 057E 3F        ?
        LD     A,$00     ; 057F 3E00     >.
        RLA      ; 0581 17        .
        LD     (HL),A    ; 0582 77        w
        CALL   Lb104     ; 0583 CDAA05    ...
        JR     Z,Lb105   ; 0586 280B      (.
        LD     A,($7816) ; 0588 3A1678    :.x
        LD     (DENFLG),A ; 058B 321778    2.x
        CALL   SETDENFLG ; 058E CDF20B    ...
        JR     Lb106     ; 0591 1816      ..
Lb105:  LD     HL,DENFLG ; 0593 211778    !.x
        LD     A,(HL)    ; 0596 7E        ~
        OR     A        ; 0597 B7        . set flags
        JR     Z,Lb106   ; 0598 280F      (.
        LD     A,($7945) ; 059A 3A4579    :Ey
        CP     $01      ; 059D FE01      ..
        JR     Z,Lb107   ; 059F 2802      (.
        LD     A,$02     ; 05A1 3E02     >. enhanced density
Lb107:  LD     (DENFLG),A ; 05A3 321778    2.x
        LD     ($7816),A ; 05A6 321678    2.x
Lb106:  RET     ; 05A9 C9      .

;-----
Lb104:  LD     A,$02     ; 05AA 3E02     >.
        LD     (CDLOOP2),A ; 05AC 322878    2(x
Lb110:  CALL   SETDENFLG ; 05AF CDF20B    ...
        CALL   Lb108     ; 05B2 CDCD05    read sector
        JR     Z,Lb109   ; 05B5 2810      (.
        CALL   Lb80      ; 05B7 CD6809    .h.
        AND   $18        ; 05BA E618      ..
        RET     Z        ; 05BC C8        .

        AND   $10        ; 05BD E610      ..
        JR     NZ,Lb109  ; 05BF 2006      .
        LD     HL,CDLOOP2 ; 05C1 212878    !(x
        DEC   (HL)      ; 05C4 35        5

```

```

Lb109: JR      NZ,Lb110          ; 05C5 20E8      .
        CALL   Lb14             ; 05C7 CD3509    .5.
        OR     $FF              ; 05CA F6FF     ..
        RET    ; 05CC C9       .
;-----
; Read sector data from fdc and store in buffer
Lb108: LD     HL,BUFFER         ; 05CD 214279    !By
        LD     A,(READADDR)     ; 05D0 3A0F78    :.x
        JR     Lb111            ; 05D3 1803     ..

Lb78:  LD     A,(READSECT)      ; 05D5 3A0D78    :.x
Lb111: LD     DE,STAT2          ; 05D8 110020    ..
        LD     (COMMANDFDC),A   ; 05DB 320060    2.`
Lb113: LD     BC,$6E73         ; 05DE 01736E    .sn
Lb112: DEC    BC                ; 05E1 0B        .
        LD     A,B              ; 05E2 78        x
        OR     C                 ; 05E3 B1        .
        RET    Z                 ; 05E4 C8        .

        LD     A,(DE)           ; 05E5 1A        .
        AND    $C0              ; 05E6 E6C0     ..
        JR     Z,Lb112          ; 05E8 28F7     (.
        AND    $80              ; 05EA E680     ..
        RET    NZ               ; 05EC C0        .

        LD     A,(DATAFDC)      ; 05ED 3A0360    :.`read byte data
        LD     (HL),A           ; 05F0 77        w  put in buffer
        INC    HL                ; 05F1 23        #
        JR     Lb113            ; 05F2 18EA     ..

; write data to FDC data register from buffer
Lb98:  LD     DE,STAT2          ; 05F4 110020    ..
        LD     A,(WRITESECT)    ; 05F7 3A0E78    :.x
        LD     (COMMANDFDC),A   ; 05FA 320060    2.`
Lb115: LD     BC,$6E73         ; 05FD 01736E    load time out value
Lb114: DEC    BC                ; 0600 0B        .
        LD     A,B              ; 0601 78        x
        OR     C                 ; 0602 B1        .
        RET    Z                 ; 0603 C8        .

        LD     A,(DE)           ; 0604 1A        .
        AND    $C0              ; 0605 E6C0     ..
        JR     Z,Lb114          ; 0607 28F7     (.
        AND    $80              ; 0609 E680     ..
        RET    NZ               ; 060B C0        .

        LD     A,(HL)           ; 060C 7E        ~  get data from buffer
        LD     (DATAFDC),A      ; 060D 320360    2.`write data to sector
        INC    HL                ; 0610 23        # next byte
        JR     Lb115            ; 0611 18EA     ..

;
;-----
;CP/M INIT Command enters here to Format track
;-----
;
Lb64:  DI     ; 0613 F3         .
        LD     HL,$00           ; 0614 210000    !..
        ADD    HL,SP            ; 0617 39        9
        LD     ($7829),HL       ; 0618 222978    save stack to 7829
        IN    A,($03)           ; 061B DB03     ip enable

```



```

IN      A,($0C)           ; 061D DB0C      ip off
LD      A,(DENFLG)       ; 061F 3A1778  density flag
OR      A                ; 0622 B7      set flags
JR      NZ,Lb116         ; 0623 2011    .
LD      DE,$254          ; 0625 115402  single
LD      C,$FF           ; 0628 0EFF    ..
LD      SP,$722         ; 062A 312207  set stack to 722
EXX     ; 062D D9      .
LD      HL,$736         ; 062E 213607  pointer to sector order
LD      DE,$6EE         ; 0631 11EE06  ...
JR      Lb117           ; 0634 181C    jump always
;-----MFM
Lb116: LD      DE,$50D   ; 0636 110D05  enhanced and double
LD      C,$4E          ; 0639 0E4E    .N
LD      SP,$785         ; 063B 318507  set stack to 785
EXX     ; 063E D9      .
LD      DE,$749         ; 063F 114907  .I.
LD      HL,$736         ; 0642 213607  sector order single/double
DEC     A              ; 0645 3D      =
LD      A,$FF          ; 0646 3EFF    bytes per sector.
JR      Z,Lb118         ; 0648 2805    Double? jump if yes
LD      HL,$79D         ; 064A 219D07  sector order enhanced
LD      A,$7F          ; 064D 3E7F    bytes per sector
Lb118: LD      (FMTSTR),A ; 064F 322B78  ff=double 7f=enhanced/single

Lb117: LD      BC,$782C  ; 0652 ED4B2C78 custom sector table
LD      A,B            ; 0656 78      x
OR      C              ; 0657 B1      .
JR      Z,Lb119         ; 0658 2802    (.
LD      L,C            ; 065A 69      load custom
LD      H,B            ; 065B 60      sector order table pointer
Lb119: EXX     ; 065C D9      .
LD      HL,DATAFDC     ; 065D 210360  load HL 6003, data register fd
LD      A,(WRITETRK)   ; 0660 3A1078  load write track command to fd
LD      (COMMANDFDC),A ; 0663 320060  2.`
Lb120: LD      A,(STAT2) ; 0666 3A0020  :. fdc
AND     $C0            ; 0669 E6C0    .. wait for drq/irq to go low
JR      Z,Lb120         ; 066B 28F9    (.
LD      (HL),C         ; 066D 71      q
LD      B,$4B          ; 066E 064B
IN      A,($0D)        ; 0672 DB0D    ip pulse?
Lb122: LD      A,(STAT2) ; 0674 3A0020  fdc
AND     $C0            ; 0677 E6C0    wait for drq/irq to go low
JR      Z,Lb122         ; 0679 28F9    (.

;HL = 6003. FDC data register
;DE = number of bytes to write after index pulse
;C = byte to write
;HL' = pointer to sector order
;DE' = 2nd stack pointer after 1st sector after Index pulse
;SP = routine to RET to or BC data

LD      (HL),C         ; 067B 71      q
IN      A,($0C)        ; 067C DB0C    ..ip not pulse?
Lb123: LD      A,(STAT2) ; 067E 3A0020  :. fdc
AND     $C0            ; 0681 E6C0    ..
JR      Z,Lb123         ; 0683 28F9    (.
LD      (HL),C         ; 0685 71      q
DEC     DE             ; 0686 1B      .
LD      A,D           ; 0687 7A      z

```

```

OR      E      ; 0688 B3      .
JR      NZ,Lb123 ; 0689 20F3      .
LD      DE,STAT2 ; 068B 110020     ..
JP      Lb124   ; 068E C39206     ...
;DE = 2000 (STAT2) for FDC IRQ/DRQ
; other registers as above

Lb125:  PUSH    BC      ; 0691 C5      inc stack pointer
Lb124:  LD      A,(DE)   ; 0692 1A      .
        AND     $C0     ; 0693 E6C0     wait for fdc drq/irq
        JR      Z,Lb124 ; 0695 28FB     (.
        LD     (HL),C   ; 0697 71      q
        RET     ; 0698 C9      .

FMTRT1:  DJNZ   Lb125   ; 0699 10F6     ..

FMTRT2:  POP     BC      ; 069B C1      .
        JP     Lb124   ; 069C C39206   ...

FMTRT3:  LD      A,($7826) ; 069F 3A2678   :&x
        RRCA   ; 06A2 0F      .
        LD     C,A     ; 06A3 4F      O
        JP     Lb124   ; 06A4 C39206   ...

FMTRT4:  LD      A,(DENFLG) ; 06A7 3A1778   :.x
        AND     $01     ; 06AA E601     ..
        LD     C,A     ; 06AC 4F      O
        JP     Lb124   ; 06AD C39206   ...

FMTRT5:  ;write sector number id field
        EXX    ; 06B0 D9      .
        LD     A,(HL)   ; 06B1 7E      sector number
        INC    HL      ; 06B2 23      #
        EXX    ; 06B3 D9      .
        LD     C,A     ; 06B4 4F      O
        JP     Lb124   ; 06B5 C39206   ...

FMTRT6:  LD      A,(FMTSTR) ; 06B8 3A2B78   :+x
        LD     B,A     ; 06BB 47      G
        JP     Lb124   ; 06BC C39206   ...

FMTRT7:  EXX    ; 06BF D9      .
        LD     C,(HL)   ; 06C0 4E      N
        EX     DE,HL   ; 06C1 EB      .
        EXX    ; 06C2 D9      .
        JP     Lb124   ; 06C3 C39206   ...

FMTRT8:  EXX    ; 06C6 D9      .
        LD     SP,HL   ; 06C7 F9      .
        EX     DE,HL   ; 06C8 EB      .
        LD     A,C     ; 06C9 79      y
        EXX    ; 06CA D9      .
        RLCA   ; 06CB 07      .
        JP     NC,Lb124 ; 06CC D29206   ...

Lb126:  LD      A,(DE)   ; 06CF 1A      .
        AND     $C0     ; 06D0 E6C0     wait for FDC drq/irq
        JR      Z,Lb126 ; 06D2 28FB     (.
        LD     (HL),C   ; 06D4 71      q
        IN     A,($0D)  ; 06D5 DB0D     .. ip pulse
Lb127:  LD      A,(DE)   ; 06D7 1A      .

```

```

AND      $C0                ; 06D8 E6C0      wait for fdc drq/irq
JR       Z,Lb127            ; 06DA 28FB      (
AND      $80                ; 06DC E680      ..
JR       NZ,Lb128          ; 06DE 2003      .
LD       (HL),C             ; 06E0 71        q
JR       Lb127              ; 06E1 18F4      ..
Lb128:  LD      SP,($7829)   ; 06E3 ED7B2978 restore stack pointer
IN       A,($0C)           ; 06E7 DB0C      ..ip pulse
CALL    Lb80                ; 06E9 CD6809    .h.
EI       ; 06EC FB        .
RET      ; 06ED C9        .

```

;data*****?;

```

WORD FMTRT2                ; 06EE 9B06      set de here FM
DB $00,$06                 ; 06F0 0006      ..
WORD FMTRT1                ; 06F2 9906      ..
WORD $01FE                 ; 06F4 FE01      ..
WORD FMTRT3                ; 06F6 9F06      ...
WORD FMTRT2                ; 06F8 9B06      .
WORD $0100                 ; 06FA 0001      ..
WORD FMTRT5                ; 06FC B006      ...write sector number
WORD FMTRT2                ; 06FE 9B06      .
WORD $0100                 ; 0700 0001      ..
WORD FMTRT2                ; 0702 9B06      ...
WORD $01F7                 ; 0704 F701      .
WORD FMTRT2                ; 0706 9B06      ...
WORD $1100                 ; 0708 0011      .
WORD FMTRT1                ; 070A 9906      ...
WORD $01FB                 ; 070C FB01      .
WORD FMTRT2                ; 070E 9B06      ...
WORD $80FF                 ; 0710 FF80      .
WORD FMTRT1                ; 0712 9906      .
WORD $01F7                 ; 0714 F701      ..
WORD FMTRT2                ; 0716 9B06      ...
WORD $0900                 ; 0718 0009      .
WORD FMTRT1                ; 071A 9906      .
WORD $03FF                 ; 071C FF03      ..
WORD FMTRT7                ; 071E BF06      .
WORD FMTRT8                ; 0720 C606      ..
WORD FMTRT2                ; 0722 9B06      stack set to here FM
WORD $0600                 ; 0724 0006      .data and counter
WORD FMTRT1                ; 0726 9906      ..
WORD $01FC                 ; 0728 FC01      ..
WORD FMTRT2                ; 072A 9B06      ...
WORD $0800                 ; 072C 0008      .
WORD FMTRT1                ; 072E 9906      .
WORD $0300                 ; 0730 0003      ..
WORD FMTRT7                ; 0732 BF06      .
WORD FMTRT8                ; 0733 C606      ..
DB $01,$03,$05            ; 0736 010305    sector order
DB $07,$09,$0B            ; 0739 07090B    .
DB $0D,$0F,$11            ; 073C 0D0F11    .
DB $02,$04,$06            ; 073F 020406    ...
DB $08,$0A,$0C            ; 0742 080A0C    ..
DB $0E,$10,$12            ; 0745 0E1012    ..
DB $80                     ; 0748 80        end data

WORD FMTRT2                ; 0749 9B06      de set to here MFM
WORD $0C00                 ; 074B 000C      ..

```

```

WORD FMTRT1           ; 074D 9906      .
WORD $03F5           ; 074F F503      ..
WORD FMTRT1           ; 0751 9906      .
WORD $01FE           ; 0753 FE01      ..
WORD FMTRT3           ; 0755 9F06      ...
WORD FMTRT2           ; 0757 9B06      .
WORD $0100           ; 0759 0001      ..
WORD FMTRT5           ; 075B B006      ...write sector number
WORD FMTRT4           ; 075D A706      .
WORD FMTRT2           ; 075F 9B06      ..
WORD $01F7           ; 0761 F701      ..
WORD FMTRT2           ; 0763 9B06      ...
WORD $224E           ; 0765 4E22      N
WORD FMTRT1           ; 0767 9906      "...
WORD $03F5           ; 0769 F503      .
WORD FMTRT1           ; 076B 9906      .
WORD $01FB           ; 076D FB01      ..
WORD FMTRT2           ; 076F 9B06      ...
WORD $01FF           ; 0771 FF01      .
WORD FMTRT6           ; 0773 B806      ...
WORD FMTRT1           ; 0775 9906      .
WORD $01F7           ; 0777 F701      ..
WORD FMTRT2           ; 0779 9B06      ...
WORD $154E           ; 077B 4E15      N
WORD FMTRT1           ; 077D 9906      .
WORD $034E           ; 077F 4E03      .N
WORD FMTRT7           ; 0781 BF06      .
WORD FMTRT8           ; 0783 C606      ..
WORD FMTRT2           ; 0785 9B06      stack set to here MFM
WORD $0C00           ; 0787 000C      write 0 B times
WORD FMTRT1           ; 0789 9906      .
WORD $03F6           ; 078B F603      write F6 3 times
WORD FMTRT1           ; 078D 9906      .
WORD $01FC           ; 078F FC01      write FC 1 times
WORD FMTRT2           ; 0791 9B06      ...
WORD $134E           ; 0793 4E13      write 4E $13 times
WORD FMTRT1           ; 0795 9906      .
WORD $034E           ; 0797 4E03      write 4E 3 times
WORD FMTRT7           ; 0799 BF06      load first sectornumber and wr
WORD FMTRT8           ; 079B C606      set stack 749
DB $01,$03,$05      ; 079D 010305  sector order enhanced
DB $07,$09,$0B      ; 07A0 07090B      .
DB $0D,$0F,$11      ; 07A3 0D0F11      .
DB $13,$15,$17      ; 07A6 131517      ...
DB $19,$02,$04      ; 07A9 190204      .
DB $06,$08,$0A      ; 07AC 06080A      ..
DB $0C,$0E,$10      ; 07AF 0C0E10      .
DB $12,$14,$16      ; 07B2 121416      .
DB $18,$1A          ; 07B4 181A        ..
DB $80              ; 07B7 80          . end data

```

```

;-----
Lb76:  LD      A, (TRACKNUM)      ; 07B8 3A2278      : "x
      SRL     A                  ; 07BB CB3F        X 2
      LD      (TRACKFDC), A      ; 07BD 320160     2. track register
      RET                               ; 07C0 C9         .
;-----

```

```

TRK0:                               ; step to track zero
Lb16:  LD      A, $58             ; 07C1 3E58        >X
      LD      ($7826), A         ; 07C3 322678     2&x

```

```

LD      A,$08                ; 07C6 3E08      >.
LD      (CONTROL),A         ; 07C8 320030    2.0 stepper motor
CALL    Lb129                ; 07CB CD5508    .U.
LD      A,$01                ; 07CE 3E01      >.
LD      (CONTROL),A         ; 07D0 320030    2.0 stepper motor
CALL    Lb129                ; 07D3 CD5508    .U.
Lb133:  CALL    Lb130         ; 07D6 CD6F08    .o.
LD      A,($7826)           ; 07D9 3A2678    :&x
OR      A                    ; 07DC B7        .
JR      Z,Lb131              ; 07DD 2805      (.
CALL    Lb132                ; 07DF CD4208    .B.
JR      Lb133                ; 07E2 18F2      ..

Lb131:  XOR     A             ; 07E4 AF        .
CALL    Lb134                ; 07E5 CDC30A    ...
LD      (TLEDNUM1),HL       ; 07E8 223A78    " :x
CALL    UPLED-1-2-B-E       ; 07EB CD120B    ...
JR      Lb135                ; 07EE 1814      ..

STTRACK                                ;step to track-----
Lb17:   LD      A,D           ; 07F0 7A        z
LD      (TRACKNUM),A        ; 07F1 322278    2"x

; CP/M INIT command enters here for step to track
Lb63:   LD      A,(TRACKNUM) ; 07F4 3A2278    : "x
LD      HL,$7826           ; 07F7 212678    7826= track currenty at?
CP      (HL)                ; 07FA BE        .
JR      Z,Lb136             ; 07FB 2810      (.
PUSH    AF                  ; 07FD F5        .
CALL    Lb137               ; 07FE CD1C08    ...
POP     AF                  ; 0801 F1        .
JR      NC,Lb138            ; 0802 3006      0.
Lb135:  CALL    Lb132         ; 0804 CD4208    .B.
CALL    Lb139               ; 0807 CD3C08    .<.
Lb138:  CALL    Lb129         ; 080A CD5508    .U.
Lb136:  XOR     A             ; 080D AF        .
LD      (CONTROL),A         ; 080E 320030    2.0 stepper motor
LD      HL,$FFFF           ; 0811 21FFFF    !..
LD      ($7830),HL         ; 0814 223078    "0x
EI      ; 0817 FB        .
CALL    Lb140               ; 0818 CD8B08    ...
RET     ; 081B C9        .

; -----
Lb137:  LD      A,(TRACKNUM) ; 081C 3A2278    : "x
LD      HL,$7826           ; 081F 212678    !&x
CP      (HL)                ; 0822 BE        .
RET     Z                   ; 0823 C8        .

PUSH    AF                  ; 0824 F5        .
CALL    C,Lb132             ; 0825 DC4208    .B.
POP     AF                  ; 0828 F1        .
CALL    NC,Lb139            ; 0829 D43C08    .<.
LD      A,($7826)           ; 082C 3A2678    :&x
SRL     A                   ; 082F CB3F      .?
CALL    Lb134               ; 0831 CDC30A    ...
LD      (TLEDNUM1),HL       ; 0834 223A78    " :x
CALL    UPLED-1-2-B-E       ; 0837 CD120B    ...
JR      Lb137               ; 083A 18E0      ..

Lb139:  CALL    Lb129         ; 083C CD5508    .U.

```

```

INC      (HL)           ; 083F 34      4
JR       Lb141          ; 0840 1804  ..
;-----
Lb132:   CALL          Lb129      ; 0842 CD5508  .U.
DEC      (HL)           ; 0845 35      5
Lb141:   LD            A, (HL)    ; 0846 7E      ~
AND      $03            ; 0847 E603    ..
LD       B, A           ; 0849 47      G
XOR     A               ; 084A AF      .
SCF     ;               ; 084B 37      7
INC     B               ; 084C 04      .
Lb142:   RLA           ; 084D 17      .
DJNZ    Lb142          ; 084E 10FD    ..
LD      (CONTROL), A    ; 0850 320030  2.0 stepper motor
EI      ;               ; 0853 FB      .
RET     ;               ; 0854 C9      .
;-----
Lb129:   EI            ; 0855 FB      .
CALL    Lb23           ; 0856 CD440B  buttons
DI      ;               ; 0859 F3      .
LD      HL, $7830      ; 085A 213078  !0x
INC     (HL)           ; 085D 34      4
JR      NZ, Lb129      ; 085E 20F5    .
INC     HL              ; 0860 23      #
INC     (HL)           ; 0861 34      4
DEC     HL              ; 0862 2B      +
JR      NZ, Lb129      ; 0863 20F0    .
LD      HL, ($782E)    ; 0865 2A2E78  *.x
LD      ($7830), HL    ; 0868 223078  "0x
LD      HL, $7826      ; 086B 212678  !&x
RET     ;               ; 086E C9      .
;-----
Lb130:   LD            A, ($7826) ; 086F 3A2678  :&x
AND     $03            ; 0872 E603    ..
RET     Z              ; 0874 C8      .

LD      A, (STATUSFDC) ; 0875 3A0060  :.`status register
AND     $04            ; 0878 E604    .. track zero?
RET     Z              ; 087A C8      .

LD      A, $0A         ; 087B 3E0A    >.
Lb143:   DEC          A           ; 087D 3D      =
JR      NZ, Lb143      ; 087E 20FD    .
LD      A, (STATUSFDC) ; 0880 3A0060  :.`status register
AND     $04            ; 0883 E604    ..
RET     Z              ; 0885 C8      .

XOR     A              ; 0886 AF      .
LD      ($7826), A     ; 0887 322678  2&x
RET     ;               ; 088A C9      .
;-----
Lb140:   LD            A, ($7826) ; 088B 3A2678  :&x
CP      $28            ; 088E FE28    .( track 40?
LD      A, (RESTORE)   ; 0890 3A0C78  restore = 0
JR      NC, Lb144      ; 0893 3007    0.
OR      A               ; 0895 B7      .
JR      Z, Lb145       ; 0896 281D    (.
LD      A, $02         ; 0898 3E02    >.
JR      Lb146          ; 089A 1804    ..

```

```

Lb144:  OR      A           ; 089C B7      .
        JR      Z,Lb147   ; 089D 2815   (.
        XOR     A           ; 089F AF      .
Lb146:  LD      B,A         ; 08A0 47      G
        LD      A,($00A8) ; 08A1 3AA800  what?????????? 06
        AND     $FD        ; 08A4 E6FD    ..
        OR      B           ; 08A6 B0      .
        LD      ($00A8),A ; 08A7 32A800  what??????????
        LD      A,($00F0) ; 08AA 3AF000  What??????????
        AND     $FD        ; 08AD E6FD    ..
        OR      B           ; 08AF B0      .
        LD      ($00F0),A ; 08B0 32F000  store to rom?????
        RET              ; 08B3 C9      .
;-----
Lb147:  DEC     A           ; 08B4 3D      =
Lb145:  LD      ($783E),A ; 08B5 323E78  2>x
        CALL   UPLED-1-2-B-E ; 08B8 CD120B  ...
        RET              ; 08BB C9      .
MONOFF: ; motor on or off-----
        JR      NC,Lb148   ; 08BC 3058    0X
;CP/M init enters here for motor on
Lb15:   CALL   Lb14         ; 08BE CD3509  .5.reset FDC?
        LD      A,(MOTORF) ; 08C1 3A1878  :.x
        OR      A           ; 08C4 B7      set flags
        JR      NZ,Lb149   ; 08C5 2017    .
        IN      A,($0B)    ; 08C7 DB0B    ..motor on
        LD      L,$01      ; 08C9 2E01    ..
Lb150:  LD      BC,$2C7A    ; 08CB 017A2C  .z,
        CALL   Lb4         ; 08CE CD1200  ... delay.time =BC
        DEC     L           ; 08D1 2D      -wait for motor to spin up
        JR      NZ,Lb150   ; 08D2 20F7    .
        LD      A,$FF      ; 08D4 3EFF    >.
        LD      (MOTORF),A ; 08D6 321878  2.x
        LD      HL,STATUS0 ; 08D9 211E78  !.x
        SET    4,(HL)      ; 08DC CBE6    ..
Lb149:  LD      HL,$7FFF    ; 08DE 21FF7F  !..
        LD      ($7832),HL ; 08E1 223278  "2x
        RET              ; 08E4 C9      .
;-----
Lb24:   DI              ; 08E5 F3      .
        LD      HL,($7832) ; 08E6 2A3278  *2x motor run delay?
        LD      A,H        ; 08E9 7C      |
        OR      L           ; 08EA B5      .
        JR      Z,Lb151   ; 08EB 2846    (F
        DEC     HL         ; 08ED 2B      +
        LD      ($7832),HL ; 08EE 223278  "2x
        EI              ; 08F1 FB      .
Lb153:  LD      HL,($7832) ; 08F2 2A3278  *2x
        LD      A,H        ; 08F5 7C      |
        OR      L           ; 08F6 B5      .
        JR      NZ,Lb151   ; 08F7 203A    :
        DI              ; 08F9 F3      .
        LD      A,($7826) ; 08FA 3A2678  FDC status register shadow?
        CP      $4E        ; 08FD FE4E    .N
        JR      NC,Lb152   ; 08FF 301C    0.
        CALL   Lb139       ; 0901 CD3C08  .<.
        DI              ; 0904 F3      .
        LD      A,($7826) ; 0905 3A2678  :&x
        SRL     A           ; 0908 CB3F    .?

```

```

CALL      Lb134                ; 090A CDC30A
LD        (TLEDNUM1),HL        ; 090D 223A78    ":x
CALL      UPLED-1-2-B-E        ; 0910 CD120B    ...
EI        ; 0913 FB            .
JR        Lb153                ; 0914 18DC        ..
Lb148:   DI                    ; 0916 F3            .
LD        HL,$0000             ; 0917 210000    !..
LD        ($7832),HL          ; 091A 223278    "2x
Lb152:   CALL      Lb129        ; 091D CD5508    .U.
DI        ; 0920 F3            .
LD        HL,($7832)          ; 0921 2A3278    *2x
LD        A,H                  ; 0924 7C        |
OR        L                    ; 0925 B5            .
JR        NZ,Lb151             ; 0926 200B        .
IN        A,($0A)              ; 0928 DB0A        ..motor off
LD        HL,STATUS0          ; 092A 211E78    !.x
RES       4,(HL)               ; 092D CBA6        ..
XOR      A                     ; 092F AF            .
LD        (MOTORF),A          ; 0930 321878    2.x
Lb151:   EI                    ; 0933 FB            .
RET       ; 0934 C9            .
                ; reset the fdc with d8 then d0 -----
Lb14:   LD        A,$D8         ; 0935 3ED8        >. force interupt require rese
LD        (COMMANDFDC),A      ; 0937 320060    2.`command register
CALL      Lb80                 ; 093A CD6809    .h.
PUSH     AF                     ; 093D F5            .
LD        A,$D0                ; 093E 3ED0        >.terminate with no interupt
LD        (COMMANDFDC),A      ; 0940 320060    2.`command register
Lb154:   LD        A,(STATUSFDC) ; 0943 3A0060    :.`status register
LD        A,(STAT2)           ; 0946 3A0020    :. fdc
AND      $80                   ; 0949 E680        .. wait for irq to go low
JR        NZ,Lb154             ; 094B 20F6        .
LD        A,(TRACKFDC)        ; 094D 3A0160    :.`
LD        (DATAFDC),A         ; 0950 320360    2.`
LD        B,$0F                ; 0953 060F        ..
Lb155:   DJNZ     Lb155         ; 0955 10FE        ..
LD        A,$10                ; 0957 3E10        >.seek with no verify
LD        (COMMANDFDC),A      ; 0959 320060    2.`
Lb156:   LD        A,(STAT2)    ; 095C 3A0020    :. fdc
AND      $80                   ; 095F E680        ..wait for irq to go high
JR        Z,Lb156              ; 0961 28F9        (.
LD        A,(STATUSFDC)       ; 0963 3A0060    :.`
POP      AF                     ; 0966 F1            .
RET       ; 0967 C9            .
;-----
Lb80:   LD        A,(STAT2)     ; 0968 3A0020    :. fdc
AND      $80                   ; 096B E680        .. wait for irq to go high
JR        Z,Lb80               ; 096D 28F9        (.
LD        A,(STATUSFDC)       ; 096F 3A0060    :.`
RET       ; 0972 C9            .
;-----
Lb90:   CALL      Lb50          ; 0973 CDA60A    check sector is 4 or greater
LD        HL,(SECTORBUF)      ; 0976 2A2478    *$x
JR        Lb39                ; 0979 1801        Jump always
GSIOWLOCK
EX       DE,HL                 ; 097B EB            .
Lb39:   PUSH     BC             ; 097C C5            .
PUSH     HL                     ; 097D E5            .
Lb158:   PUSH     BC             ; 097E C5            .

```



```

PUSH    HL                ; 097F E5      .
CALL    GSIOBYTE         ; 0980 CDD109  ... get byte
POP     HL                ; 0983 E1      .
LD      (HL),C           ; 0984 71     q
INC     HL                ; 0985 23     #
POP     BC                ; 0986 C1     .
DJNZ   Lb158             ; 0987 10F5   ..
CALL    GSIOBYTE         ; 0989 CDD109  ... get byte
LD      A,C              ; 098C 79     y
POP     HL                ; 098D E1     .
POP     BC                ; 098E C1     .
PUSH   AF                ; 098F F5     .
CALL    caculate-chksum  ; 0990 CD810A  ... caculate checksum
POP     BC                ; 0993 C1     .
LD      C,B              ; 0994 48     H
CP      C                ; 0995 B9     .
RET     ; 0996 C9     .

Lb75:   PUSH   AF                ; 0997 F5     .
        CALL   Lb50              ; 0998 CDA60A  ...
        POP    AF                ; 099B F1     .
        LD     HL,BUFFER         ; 099C 214279  set HL to buffer
        JR     Lb44              ; 099F 180B   jump always..

Lb46:   PUSH   AF                ; 09A1 F5     save A
        CALL   Lb50              ; 09A2 CDA60A  check sector is 4 or greater
        ;B = LSB of sector size to send
        POP    AF                ; 09A5 F1     retrieve A
        LD     HL,(SECTORBUF)    ; 09A6 2A2478
        JR     Lb44              ; 09A9 1801   jump always

SSIOBLOCK:
EX      DE,HL              ; 09AB EB     .

Lb44:   PUSH   BC                ; 09AC C5     .
        PUSH  HL                ; 09AD E5     .
        PUSH  AF                ; 09AE F5     save AF
        CALL  caculate-chksum    ; 09AF CD810A  go add up checksum
        POP   BC                ; 09B2 C1     get back was AF to now BC
        PUSH  AF                ; 09B3 F5     store checksum
        LD    A,B              ; 09B4 78     what was in A now back in A
        LD    B,$5A            ; 09B5 065A   .Z
        CALL  CDOWNDELAY        ; 09B7 CD9E0C  ...
        CALL  SSIobyte         ; 09BA CD230A  send byte sio (C for read)
        POP   AF                ; 09BD F1     .
        POP   HL                ; 09BE E1     .
        POP   BC                ; 09BF C1     .
        PUSH  AF                ; 09C0 F5     .

Lb161:  PUSH   BC                ; 09C1 C5     11 .
        PUSH  HL                ; 09C2 E5     11 .
        LD    A,(HL)           ; 09C3 7E     7  load buffer to A
        CALL  SSIobyte         ; 09C4 CD230A  17 send byte sio
        POP   HL                ; 09C7 E1     10 .
        INC   HL                ; 09C8 23     6  #
        POP   BC                ; 09C9 C1     10 .
        DJNZ  Lb161            ; 09CA 10F5   13/8 ..
        POP   AF                ; 09CC F1     10 get checksum
        CALL  SSIobyte         ; 09CD CD230A  17 send byte sio
        RET     ; 09D0 C9     .
;-----

```

;this section of code is send and recieve sio. high speed is included but doesn't work

GSIOBYTE:

```
LD      A,($7815)           ; 09D1 3A1578   :.x
OR      A                   ; 09D4 B7       .
JR      NZ,Lb162           ; 09D5 2024    $
LD      E,$07              ; 09D7 1E07    ..8 bits
Lb163:  LD      A,(STAT2)   ; 09D9 3A0020  :. sio data in
AND     $04                 ; 09DC E604    wait for start bit
JR      NZ,Lb163           ; 09DE 20F9    .
LD      B,$12              ; 09E0 0612
CALL   CDOWNDELAY         ; 09E2 CD9E0C 17 lets go delay 239 t states
INC     DE                  ; 09E5 13      .
Lb164:  LD      A,(STAT2)   ; 09E6 3A0020 13 recive data bit
AND     $04                 ; 09E9 E604    7 get data bit
ADD     A,$FF              ; 09EB C6FF    7 move data bit to carry
RR      C                   ; 09ED CB19    8 rotate carry to c
LD      B,$09              ; 09EF 0609    7 ..
CALL   CDOWNDELAY         ; 09F1 CD9E0C 17 delay 122 t states
OR      $00                 ; 09F4 F600    7 ..
NOP     ; 09F6 00          4 .
DEC     E                   ; 09F7 1D          4 .
JR      NZ,Lb164           ; 09F8 20EC    7/12 .208 t states per bit
RET     ; 09FA C9          .recieved byte in C
;high speed sio-----
Lb162:  PUSH   HL           ; 09FB E5      .
LD      B,$08              ; 09FC 0608    8 bits
LD      HL,STAT2          ; 09FE 210020  !.
LD      D,$02              ; 0A01 1602    mask byte data in /2
Lb165:  RST    08H          ; 0A03 CF      get start bit
;
JR      NZ,Lb165           ; 0A04 20FD    .
Lb166:  RST    08H          ; 0A06 CF      11 .
ADD     A,$FF              ; 0A07 C6FF    7 ..
RR      C                   ; 0A09 CB19    8 ..
DJNZ   Lb166              ; 0A0B 10F9    8/13 39 t states 89 t states be
POP     HL                  ; 0A0D E1      .
RET     ; 0A0E C9          .

Lb42:   LD      A,$41       ; 0A0F 3E41    >A ack?
CALL   SSIIOBYTE         ; 0A11 CD230A  send byte sio
LD      A,($7814)        ; 0A14 3A1478  :.x
LD      ($7815),A        ; 0A17 321578  2.x
RET     ; 0A1A C9          .

Lb55:   LD      BC,$82     ; 0A1B 018200  ...
CALL   Lb4                ; 0A1E CD1200  ... delay?
LD      A,$41             ; 0A21 3E41    >A ack?

SSIIOBYTE:
CPL     ; 0A23 2F          4 invert accumulator
LD      C,A               ; 0A24 4F          4 save to C
Lb167:  LD      A,(STAT2)   ; 0A25 3A0020 13
AND     $20                ; 0A28 E620    7 command
JR      Z,Lb167           ; 0A2A 28F9    12/7 wait for command to go hig
LD      A,($7815)        ; 0A2C 3A1578 13 :.x
OR      A                   ; 0A2F B7          4 .
JR      NZ,Lb168         ; 0A30 2035    12/7 go send highspeed
IN      A,($05)          ; 0A32 DB05    11 start bit
LD      E,$08            ; 0A34 1E08    7 .. 8 bits
CALL   Lb169             ; 0A36 CDA00C 17 delay 10 t states
```

```

LD      A, (CAUX12)          ; 0A39 3A2078 13 : x
NEG
Lb170: LD      A,R           ; 0A3E ED5F 9  ._
LD      B,$06              ; 0A40 0606 7  ..
CALL    CDOWNDELAY        ; 0A42 CD9E0C 17 delay 83 t states
LD      B,C               ; 0A45 41 4
LD      A,C               ; 0A46 79 4
AND     $01                ; 0A47 E601 7 ..first bit
OR      $04                ; 0A49 F604 7 .. set txd
LD      C,A               ; 0A4B 4F 4 high or low with 1
IN      A, (C)            ; 0A4C ED78 12 .x send bit
LD      C,B               ; 0A4E 48 4 H
SRL     C                  ; 0A4F CB39 8 .9 shift right
CALL    Lb169             ; 0A51 CDA00C 17 call return delay 10 t state
DEC     E                  ; 0A54 1D 4 .
JR      NZ,Lb170          ; 0A55 20E7 7/12 send 8 bits 209 t states p
LD      B,$08              ; 0A57 0608 7 ..
CALL    CDOWNDELAY        ; 0A59 CD9E0C 17 delay 109 t states
LD      B,$00              ; 0A5C 0600 7 ..
LD      B,$00              ; 0A5E 0600 7 ..
IN      A, ($04)          ; 0A60 DB04 11 send stop bit
LD      B,$0E              ; 0A62 060E 7
JP      CDOWNDELAY        ; 0A64 C39E0C 10 wait 187 t states
                                ; send highspeed -----
Lb168: PUSH    HL          ; 0A67 E5 save HL
LD      D,C               ; 0A68 51 Q
LD      B,$08              ; 0A69 0608 8 bits
LD      HL,STAT2          ; 0A6B 210020 load sio reg to HL
LD      E,$01             ; 0A6E 1E01 clock out mask
LD      C,$05             ; 0A70 0E05 send start bit
RST     18H               ; 0A72 DF get start bit
Lb171: LD      C,$02       ; 0A73 0E02 7 send data base byte /2
SRL     D                  ; 0A75 CB3A 8 Shift right, bit 0 to carry
RL      C                  ; 0A77 CB11 8 rotate left, carry to bit 0
RST     18H               ; 0A79 DF 11 send bit
DJNZ   Lb171              ; 0A7A 10F7 8/13 47 t states 105 t states
                                ;
LD      C,$04             ; 0A7C 0E04 7 .. set txd high?
RST     18H               ; 0A7E DF 11 send stop bit
POP     HL                 ; 0A7F E1 restore HL
RET     ; 0A80 C9 .
caculate-chksum:
XOR     A                  ; -----
Lb172: ADD     A, (HL)      ; 0A81 AF . caculate checksum
ADC     A,$00              ; 0A82 86 .
INC     HL                 ; 0A83 CE00 ..
DJNZ   Lb172              ; 0A85 23 #
RET     ; 0A86 10FA ..
                                ; 0A88 C9 .
DATAINVERT:
Lb49:  PUSH    AF          ; invert data in buffer-----
PUSH    HL                 ; 0A89 F5 .
CALL    Lb50               ; 0A8A E5 .
LD      HL, (SECTORBUF)   ; 0A8B CDA60A ...
Lb173: LD      A, (HL)     ; 0A8E 2A2478 *$x
CPL
LD      (HL),A            ; 0A91 7E ~
INC     HL                 ; 0A92 2F /
DJNZ   Lb173              ; 0A93 77 w
POP     HL                 ; 0A94 23 #
DJNZ   Lb173              ; 0A95 10FA ..
POP     HL                 ; 0A97 E1 .

```

```

      POP      AF                ; 0A98 F1      .
      RET                        ; 0A99 C9      .
                                ;-----
Lb40:  LD      HL,STATUS0        ; 0A9A 211E78  !.x
      SET     0,(HL)            ; 0A9D CBC6    .. set bit 0 of
      RET                        ; 0A9F C9      . 781e to 0
                                ;-----
Lb54:  LD      HL,STATUS0        ; 0AA0 211E78  !.x
      SET     1,(HL)            ; 0AA3 CBCE    ..
      RET                        ; 0AA5 C9      .
; check if sector is 4 or greater-----
Lb50:  LD      B,$80             ; 0AA6 0680    ..
      LD      A,(CAUX12)        ; 0AA8 3A2078  : x
      CP      $04               ; 0AAB FE04    check if sector 4
      JR      NC,Lb58           ; 0AAD 3005    0.
      LD      A,($7821)         ; 0AAF 3A2178  2nd byte of CAUX12
      OR      A                 ; 0AB2 B7      .
      RET     Z                 ; 0AB3 C8      .

Lb58:  LD      B,$80             ; 0AB4 0680    ..bytes per sector single
      LD      A,(DENFLG)        ; 0AB6 3A1778  :.x
      DEC     A                 ; 0AB9 3D      = 7817 =1 = double
      RET     NZ                ; 0ABA C0      .
      LD      B,$00             ; 0ABB 0600    .. bytes per sector
      RET                        ; 0ABD C9      .

LEDNUM:
      CALL    Lb134             ; 0ABE CDC30A  ...
      EX     DE,HL             ; 0AC1 EB      .
      RET                        ; 0AC2 C9      .
                                ;-----
Lb134: LD      HL,$22           ; 0AC3 212200  !".
Lb174: INC     HL               ; 0AC6 23      #
      SUB     $0A               ; 0AC7 D60A    ..
      JR      NC,Lb174         ; 0AC9 30FB    0.
      ADD    A,$0A             ; 0ACB C60A    ..
      LD     D,(HL)            ; 0ACD 56      V
      CALL   GETLEDNUM         ; 0ACE CDED0A  ... get led display number
      LD     E,A               ; 0AD1 5F      -
      EX    DE,HL             ; 0AD2 EB      .
      RET                        ; 0AD3 C9      .

G2LEDNUM:
      CALL    Lb176             ; 0AD4 CDD90A  ...
      EX     DE,HL             ; 0AD7 EB      Swap DE to HL
      RET                        ; 0AD8 C9      .

Lb176: LD      E,A             ; 0AD9 5F      -
      RRCA   ; 0ADA 0F          rotate right A
      RRCA   ; 0ADB 0F          put high 4bits
      RRCA   ; 0ADC 0F          to low 4 bits
      RRCA   ; 0ADD 0F
      AND    $0F               ; 0ADE E60F    only want low 4 bits
      CALL   GETLEDNUM         ; 0AE0 CDED0A  get led number
      LD     D,A               ; 0AE3 57      store high in D
      LD     A,E               ; 0AE4 7B      {
      AND    $0F               ; 0AE5 E60F    get low 4 bits
      CALL   GETLEDNUM         ; 0AE7 CDED0A  get led number
      LD     E,A               ; 0AEA 5F      store in E
      EX    DE,HL             ; 0AEB EB      swap to HL.
      RET                        ; 0AEC C9      .

```

```

GETLEDNUM: ;-----
            LD      C,A          ; 0AED 4F      get number pattern
            LD      B,$00        ; 0AEE 0600   .. to display
            LD      HL,LDTBL     ; 0AF0 212300 !#.on front
            ADD     HL,BC        ; 0AF3 09     .leds
            LD      A,(HL)       ; 0AF4 7E     ~number in
            RET                 ; 0AF5 C9     . A register
            ;-----

Lb68:      LD      B,$09        ; 0AF6 0609   ..
            JR      Lb177       ; 0AF8 1803   ..
Lb179:     RLCA                 ; 0AFA 07     .
            JR      C,Lb178     ; 0AFB 3802   8.
Lb177:     DJNZ    Lb179       ; 0AFD 10FB   ..
Lb178:     LD      A,B         ; 0AFF 78     x

            CALL   GETLEDNUM    ; 0B00 CDED0A ...get number to display
            LD      L,A         ; 0B03 6F     o
            RET                 ; 0B04 C9     .
            ;-----

Lb69:      LD      (LEDERROR),HL ; 0B05 223878 "8x
            LD      HL,$7837    ; 0B08 213778 !7x
            LD      A,(HL)      ; 0B0B 7E     ~
            OR      $80         ; 0B0C F680   ..
            LD      (HL),A     ; 0B0E 77     w
            CALL   beep        ; 0B0F CDDF0B ...
            ;-----

UPLED-1-2-B-E:
Lb11:      LD      A,($7837)    ; 0B12 3A3778 :7x
            LD      HL,(LEDERROR) ; 0B15 2A3878 *8x
            ADD     A,A         ; 0B18 87     .
            JR      C,Lb12     ; 0B19 380E   8.
            CP     $04         ; 0B1B FE04   ..
            JR      Z,Lb12     ; 0B1D 280A   (.
            LD      HL,(DENLET) ; 0B1F 2A3C78 *<x
            CP     $02         ; 0B22 FE02   ..
            JR      Z,Lb12     ; 0B24 2803   (.
            LD      HL,(TLEDNUM1) ; 0B26 2A3A78 *:x
Lb12:      LD      A,H         ; 0B29 7C     |
            AND    $7F         ; 0B2A E67F   ..
            LD      H,A        ; 0B2C 67     g
            LD      A,($783E)   ; 0B2D 3A3E78 :>x
            AND    $80         ; 0B30 E680   ..
            OR     H           ; 0B32 B4     .
            LD      H,A        ; 0B33 67     g
            LD      A,L        ; 0B34 7D     }
            AND    $7F         ; 0B35 E67F   ..
            LD      L,A        ; 0B37 6F     o
            LD      A,(BUSYLD)  ; 0B38 3A3F78 :?x
            CPL                 ; 0B3B 2F     /
            AND    $80         ; 0B3C E680   ..
            OR     L           ; 0B3E B5     .
            LD      L,A        ; 0B3F 6F     o
            LD      (LED1),HL   ; 0B40 22FF4F ".O set busy led?
            RET                 ; 0B43 C9     .

BUTTONS:
;CP/M INIT command enters here for buttons
Lb23:      DI                 ; 0B44 F3     .
            LD      A,(STAT1S)  ; 0B45 3A3678 buttons status ram
            LD      B,A         ; 0B48 47     G

```

```

LD      A, (STAT1)      ; 0B49 3A0110    :..
LD      C,A            ; 0B4C 4F      0
AND     $80            ; 0B4D E680    .. disk write protect
JR      Z,Lb180        ; 0B4F 2808    (.
LD      A,$FF         ; 0B51 3EFF    >.
LD      ($7819),A     ; 0B53 321978  2.x
LD      ($781A),A     ; 0B56 321A78  2.x
Lb180: LD      A,C      ; 0B59 79      y
AND     $70           ; 0B5A E670    mask out all but buttons
LD      (STAT1S),A   ; 0B5C 323678  save to ram
LD      C,A          ; 0B5F 4F      0
LD      A,(STAT1R)  ; 0B60 3A0010  :..
LD      A,B          ; 0B63 78      x
XOR     C             ; 0B64 A9      .
AND     C             ; 0B65 A1      .
JR      Z,Lb181      ; 0B66 2853    (S
LD      B,$01        ; 0B68 0601    ..
BIT     5,A          ; 0B6A CB6F    test ID button pressed
JR      NZ,Lb182     ; 0B6C 201D    .
LD      B,$02        ; 0B6E 0602    ..
BIT     6,A          ; 0B70 CB77    test ERROR button being presse
JR      NZ,Lb183     ; 0B72 200C    .
BIT     6,C          ; 0B74 CB71    test error button held
JR      Z,Lb184      ; 0B76 281E    (.
LD      HL,$86BF     ; 0B78 21BF86  E- front leds
LD      (LEDERROR),HL ; 0B7B 223878  set error for front leds
JR      Lb185        ; 0B7E 1837    .7
Lb183: BIT     5,C    ; 0B80 CB69    test ID button held
JR      Z,Lb186      ; 0B82 282F    get out if not
LD      A,$FF         ; 0B84 3EFF    yes. set load
LD      (CPML0D),A   ; 0B86 321C78  cpm flag. hold id, press error
JR      Lb181        ; 0B89 1830    .0
Lb182: BIT     4,C    ; 0B8B CB61    test TRACK button held
JR      Z,Lb186      ; 0B8D 2824    button pressed?
LD      A,$FF         ; 0B8F 3EFF    yes. hold track,pres id
LD      (DENCST),A   ; 0B91 321B78  Track and ID button status
JR      Lb186        ; 0B94 181D    density change status
Lb184: BIT     5,C    ; 0B96 CB69    test ID button being held
JR      Z,Lb187      ; 0B98 2817    (.
XOR     A             ; 0B9A AF      hold id, press track
LD      ($7819),A     ; 0B9B 321978
LD      HL,DENFLG    ; 0B9E 211778  change density
LD      A,(HL)       ; 0BA1 7E      ~
INC     A             ; 0BA2 3C      <
CP      $03          ; 0BA3 FE03    ..
JR      C,Lb188      ; 0BA5 3801    8.
XOR     A             ; 0BA7 AF      .
Lb188: LD      (HL),A ; 0BA8 77      w
LD      ($7816),A     ; 0BA9 321678  2.x
CALL    SETDENFLG    ; 0BAC CDF20B  ...
JR      Lb185        ; 0BAF 1806    ..
Lb187: LD      B,$00  ; 0BB1 0600    ..
Lb186: LD      A,B    ; 0BB3 78      x
LD      ($7837),A     ; 0BB4 323778  1=chck disk density.2=drive ty
Lb185: EI          ; 0BB7 FB      .
CALL    beep         ; 0BB8 CDDF0B  send beep
Lb181: CALL    DNUMSWITCH ; 0BBB CDCE0B  get drive number switch
EI      ; 0BBE FB      .
LD      A,($781A)    ; 0BBF 3A1A78  :.x

```

```

OR      A          ; 0BC2 B7      .
CALL    NZ,Lb15    ; 0BC3 C4BE08  motor on
XOR     A          ; 0BC6 AF       .
LD      ($781A),A  ; 0BC7 321A78  2.x
CALL    UPLED-1-2-B-E ; 0BCA CD120B  ...
RET     ; 0BCD C9      .

;Gets drive number from dip switches
DNUMSWITCH:
LD      A,(STAT1)  ; 0BCE 3A0110  :..
AND     $03        ; 0BD1 E603    .. Drive number switch
INC     A          ; 0BD3 3C       <
PUSH    AF         ; 0BD4 F5       .
CALL    GETLEDNUM  ; 0BD5 CDED0A  ...led display number
LD      (DNMLED),A ; 0BD8 323D78  2=x store led drive number in
POP     AF         ; 0BDB F1       .
OR      $30        ; 0BDC F630    .0 device number drive set to
RET     ; 0BDE C9      .
;-----

;sends audio beep to consol
beep:   LD      C,$00      ; 0BDF 0E00    ..
        LD      H,$20      ; 0BE1 2620    &
Lb190:  LD      L,$90      ; 0BE3 2E90    ..
        LD      A,C        ; 0BE5 79      y
        XOR     $01        ; 0BE6 EE01    ..
        LD      C,A        ; 0BE8 4F      O
        IN     A,(C)       ; 0BE9 ED78    .xaudio beep?
Lb189:  DEC     L          ; 0BEB 2D      -
        JR     NZ,Lb189    ; 0BEC 20FD    .
        DEC     H          ; 0BEE 25      %
        JR     NZ,Lb190    ; 0BEF 20F2    .
        RET     ; 0BF1 C9      .

SETDENFLG:
LD      B,$88       ; 0BF2 0688    A for front led
LD      A,(DENFLG)  ; 0BF4 3A1778  :.x on FDC to single or double
OR      A          ; 0BF7 B7      set flags
JR     Z,Lb191     ; 0BF8 2808    (. depending on contents
LD      B,$83       ; 0BFA 0683    .. of $7817
CP     $01         ; 0BFC FE01    .. double
JR     Z,Lb191     ; 0BFE 2802    (.
LD      B,$C6       ; 0C00 06C6    C for front led display
Lb191:  ADD     A,$FF     ; 0C02 C6FF    if a =0
        CCF     ; 0C04 3F      then this will set a to FF
        SBC     A,A       ; 0C05 9F      else a is set to 0
        AND     $01      ; 0C06 E601    get low bit
        OR      $08      ; 0C08 F608    set to dden line
        LD      C,A        ; 0C0A 4F      O
        IN     A,(C)     ; 0C0B ED78    set dden to 1 or 0
        LD      A,B       ; 0C0D 78      B=88 single 83 double
        LD      (DENLET),A ; 0C0E 323C78  C6 enhanced
        RET     ; 0C11 C9      88=A 83=b c6=C
        ;               ;               for front led display

; this routine determines the FDC type and loads FDC commands to 780C,7810
Lb18:   CALL    Lb14      ; 0C12 CD3509  .5.
        LD      HL,$0033  ; 0C15 213300  !3.
        LD      A,$D8     ; 0C18 3ED8    >.
        LD      (COMMANDFDC),A ; 0C1A 320060  2.`
        LD      B,$0A     ; 0C1D 060A    ..
Lb192:  DJNZ   Lb192     ; 0C1F 10FE    ..
        LD      A,(STAT2) ; 0C21 3A0020  :.

```

```

AND      $80                ; 0C24 E680    ..
JR       Z,Lb193           ; 0C26 2803    (.irq low
LD       HL,$C37          ; 0C28 21370C  !7.
Lb193:  LD       DE,RESTORE ; 0C2B 110C78  ..x
LD       BC,$05          ; 0C2E 010500  ...
LDIR    ; 0C31 EDB0    ..
CALL    Lb14             ; 0C33 CD3509  .5.
RET     ; 0C36 C9      .

;data.....FDC Commands 2797
DB $00 ; 0C37 00    Restore
DB $88 ; 0C38 88    Read sector command, side = 0
DB $A8 ; 0C39 A8    Write sector command, side = 0
DB $C0 ; 0C3A C0    Read address command
DB $F0 ; 0C3B F0    Write track command, side = 0
;.....
; This routine makes sure sector required doesn't excede number of sectors on disk
CHKSECTNUM:
LD      HL,(CAUX12)      ; 0C3C 2A2078  * x
DEC     HL              ; 0C3F 2B      +
LD      DE,$2D0         ; 0C40 11D002  ... total sectors on disk. sin
LD      A,(DENFLG)     ; 0C43 3A1778  :.x
CP      $02            ; 0C46 FE02    ..
JR      NZ,Lb194       ; 0C48 2003    .
LD      DE,$410        ; 0C4A 111004  ... total sectors on disk enha
Lb194:  OR       A       ; 0C4D B7      .
SBC    HL,DE          ; 0C4E ED52    .R
JR      C,Lb195       ; 0C50 3803    8.
OR     $FF           ; 0C52 F6FF    ..
RET    ; 0C54 C9      .

Lb195:  XOR     A       ; 0C55 AF      .
RET    ; 0C56 C9      .

; this routine get track and sector number of disk from total sector number
; In command frame
TRKSECT:
Lb87:  LD      HL,(CAUX12) ; 0C57 2A2078  sector num from cmd frame
DEC     HL          ; 0C5A 2B      subtract 1
LD      BC,$1012   ; 0C5B 011210  18 sectors per track in C
LD      A,(DENFLG) ; 0C5E 3A1778  :.x
CP      $02        ; 0C61 FE02    enhanced density?
JR      NZ,Lb196   ; 0C63 2002    .
LD      C,$1A      ; 0C65 0E1A    Yes. 26 sectors per track in C
Lb196:  LD      D,$00 ; 0C67 1600    ..
XOR    A          ; 0C69 AF      zero A
Lb198:  SLA     D      ; 0C6A CB22    shift left to carry
ADD    HL,HL      ; 0C6C 29      double sector number
RLA    ; 0C6D 17    rotate left accumulator
CP     C          ; 0C6E B9      compare a with C
JR     C,Lb197   ; 0C6F 3802    8.
SUB    C          ; 0C71 91      .
INC    D          ; 0C72 14      .
Lb197:  DJNZ   Lb198 ; 0C73 10F5    ..
INC    A          ; 0C75 3C      <
LD     (SECTORN),A ; 0C76 322378  2#x
LD     A,D        ; 0C79 7A      z
ADD    A,A        ; 0C7A 87      .
LD     (TRACKNUM),A ; 0C7B 322278  2"x
RET    ; 0C7E C9      .

; read track register, multiply out and add sector register. 18/26 sectors allowed for

```



```

WORD BC7824 ; 0CCE DC0C e $CDC Returns sector buffer poi
WORD IX780C ; 0CD0 E00C f $CE0 returns FDC Restore comma
WORD beep ; 0CD2 DF0B 10 $BDF Send Beep to audio in li
WORD UPLED-1-2-B-E ; 0CD4 120B 11 $B12 Update front led display
WORD LBCDE ; 0CD6 E50C 12 $CE5 returns FDC read/write s
WORD SETDENFLG ; 0CD8 F20B 13 $BF2 set density Flags and DD
WORD LDEIX ; 0CDA EC0C 14 $CEC DE=781F,IX=$7826. flags
BC7824:
LD BC,SECTORBUF ; 0CDC 012478 .$x
RET ; 0CDF C9 .
IX780C:
LD IX,RESTORE ; 0CE0 DD210C78 .!.x
RET ; 0CE4 C9 .
LBCDE
LD BC,READSECT ; 0CE5 010D78 ..x
LD DE,WRITESECT ; 0CE8 110E78 ..x
RET ; 0CEB C9 .
LDEIX
LD DE,$FDCSTA ; 0CEC 111F78 ..x
LD IX,$7826 ; 0CEF DD212678 .!&x
RET ; 0CF3 C9 .
VERNUM
LD DE,$120 ; 0CF4 112001 . . version number
RET ; 0CF7 C9 .
;-----
;routines for startup test.
Lb218: DI ; 0CF8 F3 .
LD SP,HL ; 0CF9 F9 .
Lb213: DI ; 0CFA F3 .
LD A,$89 ; 0CFB 3E89 >.
LD (LED2),A ; 0CFD 320050 2. figure H
LD B,$70 ; 0D00 0670 .p
LD C,$00 ; 0D02 0E00 ..
Lb205: LD A,(STAT1R) ; 0D04 3A0010 :.. front buttons
LD A,(STAT1) ; 0D07 3A0110 :..
AND B ; 0D0A A0 .
JR Z,Lb204 ; 0D0B 280D (.
DEC L ; 0D0D 2D -
JR NZ,Lb205 ; 0D0E 20F4 .
LD L,$90 ; 0D10 2E90 ..
LD A,C ; 0D12 79 y
XOR $01 ; 0D13 EE01 ..
LD C,A ; 0D15 4F O
IN A,(C) ; 0D16 ED78 .x audio beep
JR Lb205 ; 0D18 18EA ..
Lb204: LD A,(STAT1R) ; 0D1A 3A0010 :..
LD A,(STAT1) ; 0D1D 3A0110 :..
AND B ; 0D20 A0 .
CP B ; 0D21 B8 .
JR Z,Lb206 ; 0D22 280D (.
DEC L ; 0D24 2D -
JR NZ,Lb204 ; 0D25 20F3 .
LD L,$90 ; 0D27 2E90 ..
LD A,C ; 0D29 79 y
XOR $01 ; 0D2A EE01 .. audio beep
LD C,A ; 0D2C 4F O
IN A,(C) ; 0D2D ED78 .x
JR Lb204 ; 0D2F 18E9 ..
Lb206: LD A,$C6 ; 0D31 3EC6 >. figure C

```

```

LD      (LED2),A          ; 0D33 320050  2.P
SBC     HL,HL             ; 0D36 ED62    .b
ADD     HL,SP             ; 0D38 39    9
JP      (HL)              ; 0D39 E9    . jump somewhere

Lb207:  HALT              ; 0D3A 76    v
;startup test routine-----
Lb0:    DI                ; 0D3B F3    .init routine
IN      A,($03)           ; 0D3C DB03  ip enable?
IN      A,($04)           ; 0D3E DB04  txd high
IN      A,($06)           ; 0D40 DB06  rxd high
IN      A,($0A)           ; 0D42 DB0A  motor off
IN      A,($0C)           ; 0D44 DB0C  ip on?
XOR     A                 ; 0D46 AF    zero A
JR      NZ,Lb207          ; 0D47 20F1  get out if A <>0
JR      C,Lb207           ; 0D49 38EF  8.
JP      P0,Lb207          ; 0D4B E23A0D  .:.
JP      M,Lb207           ; 0D4E FA3A0D  .:.
DEC     A                 ; 0D51 3D    =
ADD     A,A               ; 0D52 87    .
JR      Z,Lb207           ; 0D53 28E5  check zero flag
JR      NC,Lb207          ; 0D55 30E3  check carry flag
JP      PE,Lb207          ; 0D57 EA3A0D  check positive flag
JP      P,Lb207           ; 0D5A F23A0D  check paraity flag
LD      A,$55             ; 0D5D 3E55  >U
CP      $55               ; 0D5F FE55  .U
JR      NZ,Lb207          ; 0D61 20D7  .
RLCA                    ; 0D63 07    .
CP      $AA               ; 0D64 FEAA  ..
JR      NZ,Lb207          ; 0D66 20D2  .
LD      A,$80             ; 0D68 3E80  >. Figure 8
LD      (LED1),A          ; 0D6A 32FF4F  2.O
LD      (LED2),A          ; 0D6D 320050  2.P
XOR     A                 ; 0D70 AF    .
Lb208:  LD      B,A        ; 0D71 47    check registers
LD      C,B               ; 0D72 48    H
LD      D,C               ; 0D73 51    Q
LD      E,D               ; 0D74 5A    Z
LD      H,E               ; 0D75 63    c
LD      L,H               ; 0D76 6C    l
EX      AF,A'F'           ; 0D77 08    .
LD      A,L               ; 0D78 7D    }
LD      I,A               ; 0D79 ED47  .G
LD      A,I               ; 0D7B ED57  .W
EXX                    ; 0D7D D9    .
LD      B,A               ; 0D7E 47    G
LD      C,B               ; 0D7F 48    H
LD      D,C               ; 0D80 51    Q
LD      E,D               ; 0D81 5A    Z
LD      H,E               ; 0D82 63    c
LD      L,H               ; 0D83 6C    l
EX      AF,A'F'           ; 0D84 08    .
CP      L                 ; 0D85 BD    .
JR      NZ,Lb207          ; 0D86 20B2  .
INC     A                 ; 0D88 3C    <
JR      NZ,Lb208          ; 0D89 20E6  .
LD      B,A               ; 0D8B 47    G
LD      C,A               ; 0D8C 4F    O
Lb209:  DEC     C         ; 0D8D 0D    .

```

```

JR      NZ,Lb209           ; 0D8E 20FD      .
DJNZ   Lb209             ; 0D90 10FB      ..
XOR    A                 ; 0D92 AF        .
LD     E,A               ; 0D93 5F        _
LD     D,A               ; 0D94 57        W
Lb210: EX    DE,HL        ; 0D95 EB        .
LD     SP,HL            ; 0D96 F9        .
EX     DE,HL            ; 0D97 EB        .
LD     L,A               ; 0D98 6F        o
LD     H,A               ; 0D99 67        g
ADD    HL,SP             ; 0D9A 39        9
SBC    HL,DE             ; 0D9B ED52      .R
JR     NZ,Lb207         ; 0D9D 209B      .
INC    E                 ; 0D9F 1C        .
INC    D                 ; 0DA0 14        .
JR     NZ,Lb210         ; 0DA1 20F2      end test processor
XOR    A                 ; 0DA3 AF        zero A
LD     I,A               ; 0DA4 ED47      .G
LD     A,(STAT1R)       ; 0DA6 3A0010    :..
LD     A,(STAT1)        ; 0DA9 3A0110    :..
LD     B,A               ; 0DAC 47        G
OR     $70               ; 0DAD F670      .p
SUB    B                 ; 0DAF 90        .
JR     NZ,Lb211         ; 0DB0 2003      .
DEC    A                 ; 0DB2 3D        =
LD     I,A               ; 0DB3 ED47      .G
Lb211: LD    HL,$C6C0    ; 0DB5 21C0C6    front leds = C0
LD     (LED1),HL        ; 0DB8 22FF4F    ".0
LD     SP,$DDC          ; 0DBB 31DC0D    1..
LD     BC,$FFE          ; 0DBE 01FE0F    rom to end chksum
LD     DE,$00           ; 0DC1 110000    ...
LD     HL,$00           ; 0DC4 210000    !..
Lb212: LD    A,(HL)      ; 0DC7 7E        ~
ADD    A,E               ; 0DC8 83        add up chksum
LD     E,A               ; 0DC9 5F        _
LD     A,D               ; 0DCA 7A        z
ADC    A,$00            ; 0DCB CE00      ..
LD     D,A               ; 0DCD 57        W test rom
INC    HL                ; 0DCE 23        #
DEC    BC                ; 0DCF 0B        .
LD     A,B               ; 0DD0 78        x
OR     C                 ; 0DD1 B1        .
JR     NZ,Lb212         ; 0DD2 20F3      .
LD     HL,($FFE)        ; 0DD4 2AFE0F    *..
SBC    HL,DE             ; 0DD7 ED52      .R
JP     NZ,Lb213         ; 0DD9 C2FA0C    jump if problem
LD     A,$F9            ; 0DDC 3EF9      1 in front led
LD     (LED1),A         ; 0DDE 32FF4F    2.0
LD     SP,$E08          ; 0DE1 31080E    1..
LD     A,$55            ; 0DE4 3E55      >U
Lb216: LD    HL,RINVEC   ; 0DE6 210078    ram start
LD     BC,$800          ; 0DE9 010008    ram length
Lb214: LD    (HL),A      ; 0DEC 77        w
RLCA   ; 0DED 07        .
INC    HL                ; 0DEE 23        #
DEC    C                 ; 0DEF 0D        .
JR     NZ,Lb214         ; 0DF0 20FA      .
DJNZ   Lb214            ; 0DF2 10F8      ..
LD     HL,RINVEC        ; 0DF4 210078    !.x

```

```

Lb215: LD      BC, $800           ; 0DF7 010008 ...
        CP      (HL)             ; 0DFA BE      .
        JP      NZ, Lb213        ; 0DFB C2FA0C ...
        RLCA                     ; 0DFE 07      .ram quick test
        INC     HL               ; 0DFF 23      #
        DEC     C                ; 0E00 0D      .
        JR      NZ, Lb215        ; 0E01 20F7    .
        DJNZ    Lb215           ; 0E03 10F5    ..
        RLCA                     ; 0E05 07      .
        JR      NC, Lb216        ; 0E06 30DE    0.
        LD      A, I             ; 0E08 ED57    .W
        OR      A                ; 0E0A B7      .
        JR      NZ, Lb217        ; 0E0B 2026    &
        LD      A, $A4           ; 0E0D 3EA4    >. figure 2
        LD      (LED1), A       ; 0E0F 32FF4F  2.0
        LD      HL, $E33        ; 0E12 21330E  !3.
        LD      SP, $8000       ; 0E15 310080  1..
        LD      DE, $00         ; 0E18 110000  ...
Lb219: PUSH    DE               ; 0E1B D5      .
        POP     IX              ; 0E1C DDE1    ..
        PUSH   IX              ; 0E1E DDE5    ..
        POP     IY              ; 0E20 FDE1    ..index register test
        PUSH   IY              ; 0E22 FDE5    ..
        POP     BC              ; 0E24 C1      .
        LD      A, D            ; 0E25 7A      z
        CP      B               ; 0E26 B8      .
        JP      NZ, Lb218        ; 0E27 C2F80C  ...
        LD      A, E            ; 0E2A 7B      {
        CP      C               ; 0E2B B9      .
        JP      NZ, Lb218        ; 0E2C C2F80C  ...
        INC     E               ; 0E2F 1C      .
        INC     D               ; 0E30 14      .
        JR      NZ, Lb219        ; 0E31 20E8    .
Lb217: LD      A, $B0           ; 0E33 3EB0    >. figure 3
        LD      (LED1), A       ; 0E35 32FF4F  2.0
        LD      SP, $EB1        ; 0E38 31B10E  1..
        IN      A, ($08)        ; 0E3B DB08    ..dden low
        LD      A, $D8          ; 0E3D 3ED8    >.
        LD      (COMMANDFDC), A ; 0E3F 320060  2.`
        LD      B, $0F          ; 0E42 060F    ..
Lb220: DJNZ    Lb220           ; 0E44 10FE    ..
        LD      A, (STAT2)      ; 0E46 3A0020  :. irq low
        AND     $80             ; 0E49 E680    ..
        JP      Z, Lb213        ; 0E4B CAFA0C  ... test fdc
        LD      A, $D0          ; 0E4E 3ED0    >.
        LD      (COMMANDFDC), A ; 0E50 320060  2.`
        LD      B, $0F          ; 0E53 060F    ..
Lb221: DJNZ    Lb221           ; 0E55 10FE    ..
        LD      A, (STATUSFDC) ; 0E57 3A0060  :.`
        LD      B, $0F          ; 0E5A 060F    ..
Lb222: DJNZ    Lb222           ; 0E5C 10FE    ..
        LD      A, (STAT2)      ; 0E5E 3A0020  :.
        AND     $80             ; 0E61 E680    .. irq high
        JP      NZ, Lb213        ; 0E63 C2FA0C  ...
        XOR     A               ; 0E66 AF      .
Lb225: LD      HL, TRACKFDC     ; 0E67 210160  !.`
        LD      B, $03          ; 0E6A 0603    ..
Lb223: LD      (HL), A         ; 0E6C 77      w
        INC     HL              ; 0E6D 23      #

```

```

INC      A           ; 0E6E 3C      <
DJNZ    Lb223       ; 0E6F 10FB     ..
LD      B,$03       ; 0E71 0603     ..
Lb224:  DEC      HL   ; 0E73 2B      +
DEC      A           ; 0E74 3D      =
CP      (HL)        ; 0E75 BE      .
JP      NZ,Lb213    ; 0E76 C2FA0C   ...
DJNZ    Lb224       ; 0E79 10F8     ..
INC      A           ; 0E7B 3C      <
JR      NZ,Lb225    ; 0E7C 20E9     .
LD      A,(TRACKFDC) ; 0E7E 3A0160   :.`
LD      (DATAFDC),A ; 0E81 320360   2.`
LD      A,$10       ; 0E84 3E10     >.
LD      (COMMANDFDC),A ; 0E86 320060   2.`
Lb226:  LD      A,(STAT2) ; 0E89 3A0020   :.
AND     $80         ; 0E8C E680     .. wait for irq to go low
JR      Z,Lb226     ; 0E8E 28F9     (.
LD      A,(STATUSFDC) ; 0E90 3A0060   :.`
IN      A,($03)     ; 0E93 DB03     ..ip enable
IN      A,($0D)     ; 0E95 DB0D     ..ip on
LD      B,$07       ; 0E97 0607     ..
Lb227:  DJNZ    Lb227 ; 0E99 10FE     ..
LD      A,(STATUSFDC) ; 0E9B 3A0060   :.`
AND     $02         ; 0E9E E602     ..
JP      Z,Lb213     ; 0EA0 CAFA0C   ...
IN      A,($0C)     ; 0EA3 DB0C     ..ip off?
LD      B,$07       ; 0EA5 0607     ..
Lb228:  DJNZ    Lb228 ; 0EA7 10FE     ..
LD      A,(STATUSFDC) ; 0EA9 3A0060   :.`
AND     $02         ; 0EAC E602     ..
JP      NZ,Lb213     ; 0EAE C2FA0C   ...
LD      A,I         ; 0EB1 ED57     .W
OR      A           ; 0EB3 B7       .
JR      Z,Lb229     ; 0EB4 2837     (7
LD      A,$99       ; 0EB6 3E99     >figure 4
LD      (LED1),A    ; 0EB8 32FF4F   2.0
LD      SP,$EED     ; 0EBB 31ED0E   1..
LD      B,$3F       ; 0EBE 063F     .?
IN      A,($06)     ; 0EC0 DB06     ..rxd high
IN      A,($04)     ; 0EC2 DB04     ..txd high
LD      A,(STAT2)   ; 0EC4 3A0020   :.
AND     B           ; 0EC7 A0       .
CP      B           ; 0EC8 B8       .
JP      NZ,Lb213     ; 0EC9 C2FA0C   ...
IN      A,($07)     ; 0ECC DB07     ..rxd low
LD      A,(STAT2)   ; 0ECE 3A0020   :.
AND     B           ; 0ED1 A0       .
CP      $1C         ; 0ED2 FE1C     ..
JP      NZ,Lb213     ; 0ED4 C2FA0C   ...
IN      A,($05)     ; 0ED7 DB05     ..txd low
LD      A,(STAT2)   ; 0ED9 3A0020   :.
AND     B           ; 0EDC A0       .
JP      NZ,Lb213     ; 0EDD C2FA0C   ...
IN      A,($06)     ; 0EE0 DB06     ..rxd high
LD      A,(STAT2)   ; 0EE2 3A0020   :.
AND     B           ; 0EE5 A0       .
CP      $23         ; 0EE6 FE23     .#
JP      NZ,Lb213     ; 0EE8 C2FA0C   ...
IN      A,($04)     ; 0EEB DB04     ..txd high

```

```

Lb229: LD      A,I           ; 0EED ED57      .W
      OR      A           ; 0EEF B7       .
      JR      Z,Lb230     ; 0EF0 2840     (@
      LD      A,$92      ; 0EF2 3E92     figure 5
      LD      (LED1),A   ; 0EF4 32FF4F   2.O
      IM      1          ; 0EF7 ED56     .V
      LD      A,$C3      ; 0EF9 3EC3     >.
      LD      (RINVEC),A ; 0EFB 320078   2.x
      LD      HL,$F2E    ; 0EFE 212E0F   !..
      LD      ($7801),HL ; 0F01 220178   ".x
      LD      SP,$8000   ; 0F04 310080   1..
      LD      HL,$F32    ; 0F07 21320F   !2.
      LD      C,$06      ; 0F0A 0E06     ..rxd high
      IN      B,(C)      ; 0F0C ED40     .@???????
      EI
      NOP
      XOR     A           ; 0F0F 00       .
      NOP
      XOR     A           ; 0F10 AF       .
      NOP
      DI
      OR      A           ; 0F12 F3       .
      OR      A           ; 0F13 B7       .
      JP      NZ,Lb218    ; 0F14 C2F80C   ...
      INC     C           ; 0F17 0C       .
      EI
      IN      B,(C)      ; 0F19 ED40     .@rxd low
      NOP
      DI
      OR      A           ; 0F1C F3       .test sio bus
      OR      A           ; 0F1D B7       .look for shorting plug
      JP      Z,Lb218    ; 0F1E CAF80C   ...
      XOR     A           ; 0F21 AF       .
      DEC     C           ; 0F22 0D       .
      EI
      IN      B,(C)      ; 0F24 ED40     .@txd low
      NOP
      DI
      OR      A           ; 0F28 B7       .
      JP      NZ,Lb218    ; 0F29 C2F80C   ...
      JR      Lb230
      LD      A,$FF      ; 0F2E 3EFF     >.
      RETI
      ; 0F30 ED4D     .M

Lb230: LD      A,I           ; 0F32 ED57      .W
      OR      A           ; 0F34 B7       .
      JR      Z,Lb231     ; 0F35 2861     (a
      LD      A,$82      ; 0F37 3E82     >. figure 6
      LD      (LED1),A   ; 0F39 32FF4F   2.O
      LD      SP,$F98    ; 0F3C 31980F   1..
      LD      BC,$7FF    ; 0F3F 01FF07   ...
      LD      DE,$7801   ; 0F42 110178   ..x
      LD      HL,RINVEC  ; 0F45 210078   !.x
      LD      (HL),$00   ; 0F48 3600     6.
      LDIR
      LD      E,$C6      ; 0F4C 1EC6     ..
      LD      A,$01      ; 0F4E 3E01     >.

Lb235: LD      HL,RINVEC  ; 0F50 210078   !.x
      LD      BC,$800    ; 0F53 010008   ...
      EXX
      LD      DE,RINVEC  ; 0F57 110078   ..x
      EXX
      ; 0F5A D9       .

Lb234: LD      (HL),A    ; 0F5B 77       w

```

```

EX      AF,A'F'          ; 0F5C 08      .
EXX                      ; 0F5D D9      .
LD      HL,RINVEC      ; 0F5E 210078  !.x
LD      BC,$800        ; 0F61 010008  ...
XOR     A              ; 0F64 AF      .
Lb233: CP      (HL)     ; 0F65 BE      .
JR      Z,Lb232        ; 0F66 280A    (.
LD      A,E           ; 0F68 7B      {
SUB     L             ; 0F69 95      .
JP      NZ,Lb213      ; 0F6A C2FA0C  ...
LD      A,D           ; 0F6D 7A      z
SUB     H             ; 0F6E 94      .
JP      NZ,Lb213      ; 0F6F C2FA0C  ...
Lb232: INC     HL      ; 0F72 23      #
DEC     C             ; 0F73 0D      .
JR      NZ,Lb233      ; 0F74 20EF    .
DJNZ   Lb233         ; 0F76 10ED    ..
INC     DE            ; 0F78 13      .
EXX                      ; 0F79 D9      .
EX      AF,A'F'        ; 0F7A 08      .
CP      (HL)         ; 0F7B BE      .
JP      NZ,Lb213      ; 0F7C C2FA0C  ...
LD      (HL),$00      ; 0F7F 3600    6.
INC     HL            ; 0F81 23      #
DEC     C             ; 0F82 0D      .
JR      NZ,Lb234      ; 0F83 20D6    .
EX      AF,A'F'        ; 0F85 08      .
LD      A,E           ; 0F86 7B      {
XOR     $61          ; 0F87 EE61    .a
LD      E,A          ; 0F89 5F      _
LD      (LED2),A     ; 0F8A 320050  2.P
EX      AF,A'F'        ; 0F8D 08      .
DJNZ   Lb234         ; 0F8E 10CB    ..
RLCA                      ; 0F90 07      .
JR      NC,Lb235      ; 0F91 30BD    0.
LD      A,$C6        ; 0F93 3EC6    >.
LD      (LED2),A     ; 0F95 320050  2.P
Lb231: LD      HL,$86BF ; 0F98 21BF86  !..
LD      (LED1),HL   ; 0F9B 22FF4F  ".O
LD      A,I          ; 0F9E ED57    .W
OR      A            ; 0FA0 B7      .
JP      Z,Lb236      ; 0FA1 CA8400  ... jmp init
LD      C,$00        ; 0FA4 0E00    ..
Lb237: DEC     L      ; 0FA6 2D      -
JR      NZ,Lb237      ; 0FA7 20FD    .
LD      L,$90        ; 0FA9 2E90    ..
LD      A,C          ; 0FAB 79      y
XOR     $01          ; 0FAC EE01    ..
LD      C,A          ; 0FAE 4F      O
IN      A,(C)        ; 0FAF ED78    .x audio?
LD      B,$70        ; 0FB1 0670    .p
LD      A,(STAT1R)   ; 0FB3 3A0010  :..
LD      A,(STAT1)    ; 0FB6 3A0110  :..
AND     B            ; 0FB9 A0      .
CP      B            ; 0FBA B8      .
JR      NZ,Lb237      ; 0FBB 20E9    .
JP      Lb0          ; 0FBD C33B0D  .i.
NOP                      ; 0FC0 00      .
NOP                      ; 0FC1 00      .

```



```

NOP ; 0FC2 00 .
NOP ; 0FC3 00 .
NOP ; 0FC4 00 .
NOP ; 0FC5 00 .
NOP ; 0FC6 00 .
NOP ; 0FC7 00 .
NOP ; 0FC8 00 .
NOP ; 0FC9 00 .
NOP ; 0FCA 00 .
NOP ; 0FCB 00 .
NOP ; 0FCC 00 .
NOP ; 0FCD 00 .
NOP ; 0FCE 00 .
NOP ; 0FCF 00 .
NOP ; 0FD0 00 .
NOP ; 0FD1 00 .
NOP ; 0FD2 00 .
LD BC,$704 ; 0FD3 010407 ...
LD A,(BC) ; 0FD6 0A .
INC H ; 0FD7 24 $
LD B,B ; 0FD8 40 @
INC L ; 0FD9 2C ,
RLA ; 0FDA 17 .
SCF ; 0FDB 37 7
ADD HL,DE ; 0FDC 19 .
INC C ; 0FDD 0C .
LD A,($383C) ; 0FDE 3A3C38 :<8
CCF ; 0FE1 3F ?
LD A,$0D ; 0FE2 3E0D >.
LD C,$0F ; 0FE4 0E0F ..
DB $10,$1E ; 0FE6 101E ..
RRA ; 0FE8 1F .
DEC E ; 0FE9 1D .
LD H,$2E ; 0FEA 262E &.
LD ($2023),HL ; 0FEC 222320 "#
LD A,B ; 0FEF 78 x

```

```
DB "(C)1983,KSB."
```

```
DB $20,$01 ; version number
DB $C9,$C2 ; rom chksum
```

```

;?????????????????????????????????????????????????????????ø
;? DISASSEMBLED FILE - DONE WITH DISASM 1.0. ?
;? Date: 22-06-2008 Time: 16:41 ?
;? (c) 1996 Channex aka Lasse S. Tassing ?
;? Email: ltassing.ite.dk ?
;¿?????????????????????????????????????????????????????????

```

```

; ?[CODE]?????????????????????????????????????????????????????????
; LABEL INSTR. PARAMETER(s) ADR/OPCODE ASCII
; Indus syncromesh code loaded to drive from SDX. for version 1.2 rom
; included is routine that loads this code and sets $7809 to JP $78B4
; Code for firmware 1.1 is similar but starts starts 2 bytes lower and calls to
; firmware are different
;

```

```
ORG 7B84H
```

```

Lbb:   LD      HL,$00          ; 7B84 210000    !..
       LD      ($782C),HL    ; 7B87 222C78    ",x
       LD      A,C          ; 7B8A 79        Is High bit set in command?
       RLCA                ; 7B8B 07        .
       RET     NC          ; 7B8C D0        get out if not

       POP     HL          ; 7B8D E1        .
       SBC    A,A          ; 7B8E 9F        .
       LD      ($7814),A    ; 7B8F 321478    2.x
       RES    7,C          ; 7B92 CBB9    clear the high bit
       LD      B,$08        ; 7B94 0608    number of commands to check
       LD      HL,$7BB8     ; 7B96 21B87B    data where commands and pointers
Lb1a:  LD      A,(HL)       ; 7B99 7E        ~
       INC    HL          ; 7B9A 23        #
       CP     C            ; 7B9B B9        command match?
       JR     Z,Lb0a       ; 7B9C 2806    jump if it is
       INC    HL          ; 7B9E 23        point to next command
       INC    HL          ; 7B9F 23        #
       DJNZ  Lb1a         ; 7BA0 10F7    all 8 commands checked?
       JR     Lb2a         ; 7BA2 1808    ..
Lb0a:  LD      A,(HL)       ; 7BA4 7E        ~
       INC    HL          ; 7BA5 23        #
       LD      H,(HL)      ; 7BA6 66        f
       LD      L,A         ; 7BA7 6F        o
       CALL  Lb3a         ; 7BA8 CDB77B    go jump to command
       RET     Z           ; 7BAB C8        jump if command success

Lb2a:  LD      BC,$82       ; 7BAC 018200    load 82
       CALL  Lb4           ; 7BAF CD1200    go delay
       LD      A,$4E       ; 7BB2 3E4E     Load 'N'ak
       JP    SENDBYTEHS   ; 7BB4 C3CC7E    go send Nak and return to rom
Lb3a:  JP     (HL)         ; 7BB7 E9        Jump to command

       DB $4E             ; 7BB8 4E        N command
       WORD GETBLOCKHS   ; 7BB9 777C
       DB $4F             ; 7BBB 4F        O command
       WORD PUTBLOCKHS   ; 7BBC A97C
       DB $50             ; 7BBE 50        P put
       WORD PUTHS        ; 7BBF 2E7C
       DB $52             ; 7BC1 52        R read
       WORD READHS      ; 7BC2 1A7C
       DB $53             ; 7BC4 53        S status
       WORD STATUSHS    ; 7BC5 D07B
       DB $57             ; 7BC7 57        W write
       WORD WRITEHS     ; 7BC8 357C
       DB $21             ; 7BCA 21        ! format
       WORD FORMATHS    ; 7BCB D27C
       DB $23             ; 7BCD 23        # format skewed?
       WORD FORMATSPL   ; 7BCE D87C

STATUSHS:
       CALL  SENDACK19   ; 7BD0 CDB87E    Send Ack at 19200
       CALL  Lb36        ; 7BD3 CD6005    .`.
       CALL  Lb14        ; 7BD6 CD3509    .5.

       LD      C,$08      ; 7BD9 0E08     .. patched
       LD      A,($DENFLG); 7BDB 3A1778    :.x
       OR     A           ; 7BDE B7        .

```

```

JR      Z ,Lb9a          ; 7BDF 2805      (.
ADD     A,$FE           ; 7BE1 C6FE      ..
RRA                    ; 7BE3 1F         .
AND     $A0             ; 7BE4 E6A0     ..
Lb9a:  LD      B,A       ; 7BE6 47        G
NOP                    ; 7BE7 00        .
NOP                    ; 7BE8 00        .
NOP                    ; 7BE9 00        .
NOP                    ; 7BEA 00        .
NOP                    ; 7BEB 00        .
NOP                    ; 7BEC 00        . end patched

;      LD      A,($6000) ; 7BD9 3A0060   original version
;      AND     $40       ; 7BDC E640     .@
;      ADD     A,$FF     ; 7BDE C6FF     ..
;      SBC     A,A       ; 7BE0 9F        .
;      AND     $08       ; 7BE1 E608     ..
;      LD      C,A       ; 7BE3 4F        O
;      LD      A,($7817) ; 7BE4 3A1778   :.x
;      ADD     A,$FF     ; 7BE7 C6FF     ..
;      SBC     A,A       ; 7BE9 9F        .
;      AND     $20       ; 7BEA E620     .
;      LD      B,A       ; 7BEC 47        end original version

LD      A,(STATUS0)    ; 7BED 3A1E78   :.x
AND     $57            ; 7BF0 E657     .W
OR      B              ; 7BF2 B0        .
OR      C              ; 7BF3 B1        .
LD      HL,BUFFER     ; 7BF4 214279   !By
LD      (HL),A        ; 7BF7 77        w
INC     HL             ; 7BF8 23        #
CALL   SETDENFLG     ; 7BF9 CDF20B   ...
LD      A,($781F)     ; 7BFC 3A1F78   :.x
CPL                    ; 7BFF 2F        /
LD      (HL),A        ; 7C00 77        w
INC     HL             ; 7C01 23        #
LD      A,($781D)     ; 7C02 3A1D78   :.x
LD      (HL),A        ; 7C05 77        w
INC     HL             ; 7C06 23        #
LD      A,($7826)     ; 7C07 3A2678   :&x
SRL     A              ; 7C0A CB3F     .?
LD      (HL),A        ; 7C0C 77        w
LD      A,$43         ; 7C0D 3E43     >C
LD      HL,BUFFER     ; 7C0F 214279   Buffer
LD      B,$04         ; 7C12 0604     number of bytes to send
CALL   SIOHIGHSPEED ; 7C14 CD9A7E   go send
JP      Lb12a         ; 7C17 C36F02   jump to rom status routine to

READHS:
CALL   CHKSECTNUM    ; 7C1A CD3C0C   .<. read
JP      NZ,Lb14       ; 7C1D C29A0A   ...
CALL   SENDACK19     ; 7C20 CDB87E   ..~
CALL   Lb36          ; 7C23 CD6005   .`.
CALL   Lb37          ; 7C26 CD9504   ...
CALL   Lb16a         ; 7C29 CD927E   ..~
XOR    A              ; 7C2C AF        .
RET                    ; 7C2D C9        .

PUTHS:
CALL   Lb17a         ; 7C2E CD287E   .(~ Put
RET    NZ             ; 7C31 C0        .

```

```

JP          Lb18a          ; 7C32 C3727C .r|

WRITEHS:
CALL        Lb17a          ; 7C35 CD287E .(~ write
RET         NZ             ; 7C38 C0      .

CP          $45            ; 7C39 FE45   .E
JR          Z,Lb18a        ; 7C3B 2835   (5
CALL        DATAINVERT   ; 7C3D CD890A ...
CALL        Lb50           ; 7C40 CDA60A get bytes per sect in B 8 bit.
LD          C,B            ; 7C43 48     make 8 bit number
LD          B,$00          ; 7C44 0600   to a 16 bit number
DEC         C              ; 7C46 0D     where $00 =$0100
INC         BC             ; 7C47 03     in BC
LD          HL,(SECTORBUF) ; 7C48 2A2478 source sector buffer
LD          DE,BUFFER      ; 7C4B 114279 dest general buffer
LDIR       ; 7C4E EDB0   copy bytes over
CALL        Lb37           ; 7C50 CD9504 ...
CP          $45            ; 7C53 FE45   .E
JR          Z,Lb21a        ; 7C55 2813   (.
CALL        Lb50           ; 7C57 CDA60A ...
LD          HL,(SECTORBUF) ; 7C5A 2A2478 pointer to sector buffer
LD          DE,BUFFER      ; 7C5D 114279 general buffer
Lb22a: LD          A,(DE)    ; 7C60 1A     Cannot work. see $02C1
CP          (HL)           ; 7C61 BE
JR          NZ,Lb21a       ; 7C62 2006   get out if no match
DJNZ       Lb22a          ; 7C64 10FA
LD          A,$43          ; 7C66 3E43   Load 'C'omplete
JR          Lb18a          ; 7C68 1808   skip next 3 lines
Lb21a: LD          HL,$8C98 ; 7C6A 21988C Load 'P9' for front leds
CALL        UPLED-1-2-B-E ; 7C6D CD120B update front leds
LD          A,$45          ; 7C70 3E45   load 'E'rror
Lb18a: CALL        SENDBYTEHS ; 7C72 CDCC7E go send
XOR         A              ; 7C75 AF     .
RET         ; 7C76 C9     .

GETBLOCKHS:
CALL        SENDACK19      ; 7C77 CDB87E ..~ n command
CALL        Lb36           ; 7C7A CD6005 .`.
LD          HL,$33E        ; 7C7D 213E03 load table
LD          DE,BUFFER      ; 7C80 114279 to buffer
LD          BC,$0C         ; 7C83 010C00 number of bytes
LDIR       ; 7C86 EDB0   move BC bytes from HL to DE
LD          HL,$7947       ; 7C88 214779 !Gy
ADD         A,$FF          ; 7C8B C6FF   ..
SBC         A,A            ; 7C8D 9F     .
AND         $04            ; 7C8E E604   ..
LD          (HL),A         ; 7C90 77     w
INC         HL             ; 7C91 23     #
CALL        Lb58           ; 7C92 CDB40A Get bytes per sector
DEC         B              ; 7C95 05     .
LD          C,B            ; 7C96 48     H
LD          B,$00          ; 7C97 0600   ..
INC         BC             ; 7C99 03     .
LD          (HL),B        ; 7C9A 70     p
INC         HL             ; 7C9B 23     #
LD          (HL),C         ; 7C9C 71     q
LD          A,$43          ; 7C9D 3E43   Load 'C'omplete
LD          HL,BUFFER      ; 7C9F 214279 load buffer
LD          B,$0C         ; 7CA2 060C   number of bytes to send

```

	CALL	SIOHIGHSPEED		; 7CA4 CD9A7E	send bytes High speed
	XOR	A		; 7CA7 AF	.
	RET			; 7CA8 C9	.
PUTBLOCKHS:					
	CALL	SENDACK19		; 7CA9 CDB87E	..~ O command
	LD	B,\$0C		; 7CAC 060C	..
	LD	HL,BUFFER		; 7CAE 214279	!By
	CALL	Lb25a		; 7CB1 CD437E	.C~
	JP	NZ,Lb54		; 7CB4 C2A00A	
	CALL	Lb27a		; 7CB7 CDC47E	..~
	CALL	Lb36		; 7CBA CD6005	..`.
	LD	A,(\$7947)		; 7CBD 3A4779	:Gy
	OR	A		; 7CC0 B7	.
	JR	Z,Lb28a		; 7CC1 2801	(.
	INC	A		; 7CC3 3C	<
Lb28a:	LD	(\$DENFLG),A		; 7CC4 321778	2.x
	LD	(\$7816),A		; 7CC7 321678	2.x
	CALL	SETDENFLG		; 7CCA CDF20B	...lb10
	LD	A,\$43		; 7CCD 3E43	>C
	JP	Lb18a		; 7CCF C3727C	.r
FORMATHS:					
	LD	A,\$4E		; 7CD2 3E4E	format
	LD	L,\$00		; 7CD4 2E00	..
	JR	Lb29a		; 7CD6 180E	..
FORMATSPL:					
	LD	L,\$02		; 7CD8 2E02	.. # command
	LD	A,(\$DENFLG)		; 7CDA 3A1778	:.x
	OR	A		; 7CDD B7	.
	LD	A,\$04		; 7CDE 3E04	>.
	JR	Z,Lb29a		; 7CE0 2804	(.
	LD	A,\$02		; 7CE2 3E02	>.
	JR	Lb29a		; 7CE4 1800	..
Lb29a:	LD	(\$7D86),A		; 7CE6 32867D	
	PUSH	HL		; 7CE9 E5	.
	LD	HL,\$DENFLG		; 7CEA 211778	!.x
	LD	A,(HL)		; 7CED 7E	~
	SUB	\$02		; 7CEE D602	..
	JR	NZ,Lb30a		; 7CF0 2007	.
	LD	(HL),A		; 7CF2 77	w
	LD	(\$7816),A		; 7CF3 321678	2.x
	CALL	SETDENFLG		; 7CF6 CDF20B	...
Lb30a:	CALL	SENDACK19		; 7CF9 CDB87E	..~
	POP	HL		; 7CFC E1	.
	LD	A,(\$DENFLG)		; 7CFD 3A1778	:.x
	PUSH	AF		; 7D00 F5	.
	ADD	A,L		; 7D01 85	.
	ADD	A,A		; 7D02 87	.
	LD	L,A		; 7D03 6F	o
	LD	H,\$00		; 7D04 2600	&.
	LD	DE,\$7D88		; 7D06 11887D	..}
	ADD	HL,DE		; 7D09 19	.
	LD	A,(HL)		; 7D0A 7E	~
	INC	HL		; 7D0B 23	#
	LD	H,(HL)		; 7D0C 66	f
	LD	L,A		; 7D0D 6F	o
	LD	(\$782C),HL		; 7D0E 222C78	","x
	CALL	Lb36		; 7D11 CD6005	..`.
	POP	AF		; 7D14 F1	.
	LD	(\$DENFLG),A		; 7D15 321778	2.x

	LD	(\$7816),A	; 7D18	321678	2.x
	CALL	SETDENFLG	; 7D1B	CDF20B	...
	CALL	Lb15	; 7D1E	CDBE08	...
	LD	HL,\$FFFF	; 7D21	21FFFF	!..
	LD	(CAUX12),HL	; 7D24	222078	" x
	CALL	Lb58	; 7D27	CDB40A	...
	XOR	A	; 7D2A	AF	.
	LD	(TRACKNUM),A	; 7D2B	322278	2"x
	DEC	A	; 7D2E	3D	=
	LD	HL,BUFFER	; 7D2F	214279	!By
Lb32a:	LD	(HL),A	; 7D32	77	w
	INC	HL	; 7D33	23	#
	DJNZ	Lb32a	; 7D34	10FC	..
Lb43a:	CALL	Lb63	; 7D36	CDF407	...
	LD	A,\$05	; 7D39	3E05	>.
	LD	(CDLOOP2),A	; 7D3B	322878	2(x
Lb38a:	CALL	BUTTONS	; 7D3E	CD440B	.D.
	CALL	Lb64	; 7D41	CD1306	...
	LD	(\$781F),A	; 7D44	321F78	2.x
	AND	\$44	; 7D47	E644	.D
	JR	Z,Lb36a	; 7D49	281B	(.
	LD	B,A	; 7D4B	47	G
	AND	\$40	; 7D4C	E640	.@
	LD	A,B	; 7D4E	78	x
	JR	NZ,Lb37a	; 7D4F	2006	.
	LD	HL,CDLOOP2	; 7D51	212878	!(x
	DEC	(HL)	; 7D54	35	5
	JR	NZ,Lb38a	; 7D55	20E7	.
Lb37a:	CALL	Lb68	; 7D57	CDF60A	...
	LD	H,\$8E	; 7D5A	268E	&.
	CALL	Lb69	; 7D5C	CD050B	...
	LD	HL,STATUS0	; 7D5F	211E78	!.x
	SET	2,(HL)	; 7D62	CBD6	..
	JR	Lb41a	; 7D64	1819	..
Lb36a:	LD	A,(\$7D86)	; 7D66	3A867D	:.}
	LD	HL,TRACKNUM	; 7D69	212278	!"x
	CP	(HL)	; 7D6C	BE	.
	JR	Z,Lb42a	; 7D6D	2804	(.
	INC	(HL)	; 7D6F	34	4
	INC	(HL)	; 7D70	34	4
	JR	Lb43a	; 7D71	18C3	..
Lb42a:	CALL	Lb23	; 7D73	CD440B	.D.
	CALL	Lb44a	; 7D76	CDB67D	..}
	JR	NZ,Lb41a	; 7D79	2004	.
	LD	A,\$43	; 7D7B	3E43	>C
	JR	Lb45a	; 7D7D	1802	..
Lb41a:	LD	A,\$45	; 7D7F	3E45	>E
Lb45a:	CALL	Lb46a	; 7D81	CD887E	..~
	XOR	A	; 7D84	AF	.
	RET		; 7D85	C9	.
	NOP		; 7D86	00	.
	NOP		; 7D87	00	.
	SUB	B	; 7D88	90	pointer to first sector table.
	LD	A,L	; 7D89	7D	}
	AND	E	; 7D8A	A3	pointer to 2nd sector table
	LD	A,L	; 7D8B	7D	}
	LD	(HL),\$07	; 7D8C	3607	6.
	LD	(HL),\$07	; 7D8E	3607	6.

```

DB $04,$08,$0C           ; 7D90 04080C sector table single hs
DB $10,$01,$05           ; 7D93 100105
DB $09,$0D,$11           ; 7D96 090D11
DB $02,$06,$0A           ; 7D99 02060A
DB $0E,$12,$03           ; 7D9C 0E1203
DB $07,$0B,$0F           ; 7D9F 070B0F

DB $80                     ; 7DA2 80

DB $01,$0E,$09           ; 7DA3 010E09 sector table double hs
DB $04,$11,$0C           ; 7DA6 04110C
DB $07,$02,$0F           ; 7DA9 07020F
DB $0A,$05,$12           ; 7DAC 0A0512
DB $0D,$08,$03           ; 7DAF 0D0803
DB $10,$0B,$06           ; 7DB2 100B06

DB $80                     ; 7DB5 80

Lb44a: CALL    Lb58           ; 7DB6 CDB40A ...
      LD      A,B           ; 7DB9 78 x
      SRL    A              ; 7DBA CB3F .?
      DEC    A              ; 7DBC 3D =
      LD     ($7829),A       ; 7DBD 322978 2)x
      XOR    A              ; 7DC0 AF .
      LD     (TRACKNUM),A    ; 7DC1 322278 2"x
      LD     IY,BUFFER      ; 7DC4 FD214279 .!By
Lb59a: CALL    Lb63           ; 7DC8 CDF407 ...
      CALL   Lb76           ; 7DCB CDB807 ...
      LD     IX,($782C)     ; 7DCE DD2A2C78 .*,x
Lb580: LD      A,(IX+$00)   ; 7DD2 DD7E00 .~.
      LD     (SECTORFDC),A ; 7DD5 320260 2.`
      LD     A,$05         ; 7DD8 3E05 >.
      LD     (CDLOOP2),A   ; 7DDA 322878 2(x
Lb55a: LD      HL,(SECTORBUF) ; 7DDD 2A2478 sector buf pointer
      CALL   Lb78           ; 7DE0 CDD505
      JR     Z,Lb51a        ; 7DE3 2805 (.
      CALL   Lb80           ; 7DE5 CD6809 .h.
      JR     Lb53a         ; 7DE8 1805 ..
Lb51a: CALL    Lb14         ; 7DEA CD3509 .5.
      OR     $10           ; 7DED F610 ..
Lb53a: AND     $1C         ; 7DEF E61C ..
      JR     Z,Lb54a        ; 7DF1 281F (.
      LD     HL,CDLOOP2    ; 7DF3 212878 !(x
      DEC    (HL)          ; 7DF6 35 5
      JR     NZ,Lb55a      ; 7DF7 20E4 .
      CALL   Lb84           ; 7DF9 CD7F0C
      LD     (IY+$00),L    ; 7DFC FD7500 .u.
      INC    IY            ; 7DFE FD23 .#
      LD     (IY+$00),H    ; 7E01 FD7400 .t.
      INC    IY            ; 7E04 FD23 .#
      LD     HL,$8E98      ; 7E06 21988E load hl with F9 for leds
      CALL   Lb69           ; 7E09 CD050B ...
      LD     HL,$7829      ; 7E0C 212978 !)x
      DEC    (HL)          ; 7E0F 35 5
      JR     Z,Lb57a        ; 7E10 2813 (.
Lb54a: INC     IX          ; 7E12 DD23 .#
      LD     A,(IX+$00)    ; 7E14 DD7E00 .~.
      RLCA                ; 7E17 07 .

```

```

JR      NC,Lb580           ; 7E18 30B8      0.
LD      HL,TRACKNUM       ; 7E1A 212278  !"x
INC     (HL)              ; 7E1D 34      4
INC     (HL)              ; 7E1E 34      4
LD      A,($7D86)        ; 7E1F 3A867D  :.}
CP      (HL)              ; 7E22 BE      .
JR      NC,Lb59a         ; 7E23 30A3    0.
Lb57a:  JP      Lb85       ; 7E25 C38E04

Lb17a:  CALL    CHKSECTNUM ; 7E28 CD3C0C  .<.
        JP      NZ,Lb40    ; 7E2B C29A0A  ...
        CALL    SENDACK19 ; 7E2E CDB87E  ..~
        CALL    Lb61a      ; 7E31 CD3D7E  .=~
        JP      NZ,Lb26    ; 7E34 C2A00A  ...
        CALL    Lb27a      ; 7E37 CDC47E  ..~
        JP      Lb62a      ; 7E3A C3B704  ...
Lb61a:  CALL    Lb50       ; 7E3D CDA60A  ...
        LD      HL,(SECTORBUF) ; 7E40 2A2478  sector buf pointer
;-----
; This section is for Highspeed Send/recieve. 58 t states used to send/recieve each bit
; 68965.5 baud. Pokey divisor 6=68209-pal, 68730-ntsc/68965
RECIEVEHS
Lb25a:  PUSH    BC         ; 7E43 C5      number of bytes to get
        PUSH    HL         ; 7E44 E5      buffer pointer

        EX      DE,HL      ; 7E45 EB      .
        LD      A,B        ; 7E46 78      load bytes to get to A
        EXX                    ; 7E47 D9      save bc,de,hl to 2ndary
        LD      C,A        ; 7E48 4F      load bytes to get to C
        DEC     C          ; 7E49 0D      subtract 1
        LD      B,$00      ; 7E4A 0600    load highbyte with 00
        INC     BC         ; 7E4C 03      add 1
        INC     BC         ; 7E4D 03      add 1 allow for chksum byte
        EXX                    ; 7E4E D9      .
        LD      HL,$2000   ; 7E4F 210020  !.
        LD      A,(HL)     ; 7E52 7E      ~
        AND    $FB        ; 7E53 E6FB    mask out recieve bit
        OR     $03        ; 7E55 F603    ignore clockin/out
        LD      B,A        ; 7E57 47      load mask to B
        JR      Lb63a      ; 7E58 1804    jump always

Lb65a:  LD      A,C        ; 7E5A 79      move recieved byte to A
        LD      (DE),A     ; 7E5B 12      Move A to buffer
        INC     DE         ; 7E5C 13      Inc to next byte of buffer
        LD      A,B        ; 7E5D 78      x
Lb63a:  CP      (HL)       ; 7E5E BE      .
        JP      C,Lb63a    ; 7E5F DA5E7E  wait for start bit
        EX      AF,A'F'    ; 7E62 08      .
        EXX                    ; 7E63 D9      .
        DEC     BC         ; 7E64 0B      dec number of bytes to get
        LD      A,B        ; 7E65 78      x
        OR     C          ; 7E66 B1      = to zero?
        EXX                    ; 7E67 D9      swap registers with 2ndary
        EX      AF,A'F'    ; 7E68 08      swap a and F registers with 2ndary
        LD      C,$7F      ; 7E69 0E7F    Bit pattern for 8 bits to recieve
        LD      I,A        ; 7E6B ED47    timing
        LD      I,A        ; 7E6D ED47    instructions
Lb64a:  CP      (HL)       ; 7E6F BE      7 .
        CP      (HL)       ; 7E70 BE      7 .

```



```

NOP                                ; 7E71 00      4 .
NOP                                ; 7E72 00      4 .
LD      I,A                        ; 7E73 ED47    9 .58 t states for recieve loop
CP      (HL)                       ; 7E75 BE      7 .
RR      C                          ; 7E76 CB19    8 rotate bit to carry to byt
JR      C,Lb64a                    ; 7E78 38F5    7/12 get next bit till bit 8 th
EX      AF,A'F'                    ; 7E7A 08      .
JP      NZ,Lb65a                   ; 7E7B C25A7E  number of bytes to get = zero?
LD      A,C                        ; 7E7E 79      yes
POP     HL                          ; 7E7F E1      .
POP     BC                          ; 7E80 C1      .
PUSH    AF                          ; 7E81 F5      .
CALL    caculate-chksum            ; 7E82 CD810A  caculate chksum
POP     BC                          ; 7E85 C1      .
CP      B                          ; 7E86 B8      .
RET                                           ; 7E87 C9      .

Lb46a: PUSH    AF                    ; 7E88 F5      .
CALL    Lb50                       ; 7E89 CDA60A  ...
POP     AF                          ; 7E8C F1      .
LD      HL,BUFFER                  ; 7E8D 214279  !By
JR      SIOHIGHSPEED              ; 7E90 1808    jump always

Lb16a: PUSH    AF                    ; 7E92 F5      .
CALL    Lb50                       ; 7E93 CDA60A  ...
POP     AF                          ; 7E96 F1      .
LD      HL,(SECTORBUF)            ; 7E97 2A2478  sector buff pointer
;
;-----
;following 5 instructions have been swapped around by ICD for timing issues with SDX.
;original code follows
;-----
;
SIOHIGHSPEED
CALL    SENDBYTEHS                ; 7E9A CDCC7E  patched
LD      C,B                       ; 7E9D 48      save B
LD      B,$5A                     ; 7E9E 065A    count 5A
CALL    CDOWNDELAY                ; 7EA0 CD9E0C  Count DOWN DELAY
LD      B,C                       ; 7EA3 41      end patched

;SIOHIGHSPEED
; LD      C,B                       ; 7E9A 48      orignal version
; LD      B,$5A                     ; 7E9B 065A    .Z
; CALL    CDOWNDELAY                ; 7E9D CD9E0C  ...
; LD      B,C                       ; 7EA0 41      A
; CALL    SENDBYTEHS                ; 7EA1 CDCC7E  end orignal version

PUSH    BC                        ; 7EA4 C5      .
PUSH    HL                        ; 7EA5 E5      .
CALL    caculate-chksum            ; 7EA6 CD810A  ...
POP     HL                          ; 7EA9 E1      .
POP     BC                          ; 7EAA C1      .
PUSH    AF                          ; 7EAB F5      .
Lb68a: LD      A,(HL)               ; 7EAC 7E      7 load byte to send
CALL    SENDBYTEHS                ; 7EAD CDCC7E  17 send data
INC     HL                          ; 7EB0 23      11 point to next byte
DJNZ   Lb68a                      ; 7EB1 10F9    13/8 all data?
POP     AF                          ; 7EB3 F1      get chksum
CALL    SENDBYTEHS                ; 7EB4 CDCC7E  send chksum

```

```

RET ; 7EB7 C9 .
SENDACK19
Lb600: LD A,$41 ; 7EB8 3E41 Load A for ack
CALL SSIobyte ; 7EBA CD230A Send byte at 19200
LD A,($7814) ; 7EBD 3A1478 :.x
LD ($7815),A ; 7EC0 321578 2.x
RET ; 7EC3 C9 .

Lb27a: LD BC,$82 ; 7EC4 018200 ...
CALL Lb4 ; 7EC7 CD1200 ...
LD A,$41 ; 7ECA 3E41 >A

SENDBYTEHS
EXX ; 7ECC D9 4 .
CPL ; 7ECD 2F 4 invert data in A
LD D,A ; 7ECE 57 4 store byte to send in D
LD E,$02 ; 7ECF 1E02 7 base byte divided by 2
IN A,($05) ; 7ED1 DB05 11 send start bit
LD B,$08 ; 7ED3 0608 7 Number of bits
NEG ; 7ED5 ED44 8 delay
Lb70a: RR D ; 7ED7 CB1A 8 rotate bit to carry
LD C,E ; 7ED9 4B 4 load base byte
RL C ; 7EDA CB11 8 rotate carry to base Byte.X
IN A,(C) ; 7EDC ED78 12 send bit to sio buss
LD I,A ; 7EDE ED47 9 timing. 58 t states to send
NOP ; 7EE0 00 4 instructions
DJNZ Lb70a ; 7EE1 10F4 8/13 dec B and jump if not zero
LD I,A ; 7EE3 ED47 4 timing
LD I,A ; 7EE5 ED47 4 instructions
NEG ; 7EE7 ED44 8
IN A,($04) ; 7EE9 DB04 11 send stop bit
EXX ; 7EEB D9 4
RET ; 7EEC C9 10

```

```

;????????????????????????????????????????????????????????????

```

```

;? DISASSEMBLED FILE - DONE WITH DISASM 1.0. ?

```

```

;? Date: 22-06-2008 Time: 15:16 ?

```

```

;? (c) 1996 Channex aka Lasse S. Tassing ?

```

```

;? Email: ltassing.ite.dk ?

```

```

;????????????????????????????????????????????????????????????

```

```

;The following is used by syncromesh loader to get the Indus GT Firmware version number

```

```

;Syncromesh only works with Version 1.1 and version 1.2 Firmware

```

```

ORG 7F00H

```

```

LD C,$00 ; 7F00 0E00 ..
CALL Lba ; 7F02 CD0400 call routine 0. data returned
LD (TEMP),DE ; 7F05 ED53177F save contents of DE

```

```

; add stuff in here

```

```

LD DE,TEMP ; 7F09 11177F load location of data to DE
LD B,$02 ; 7F0C 0602 Num of bytes to send
LD C,$08 ; 7F0E 0E08 load routine number to C
LD A,$43 ; 7F10 3E43 'C'omplete
CALL Lba ; 7F12 CD0400 Call routine 8
OR A ; 7F15 B7 set flags
RET ; 7F16 C9 get out

```

```

TEMP: BD $00,$00 ; 7F17 0000 temp storage

```

; ?[CODE]??

; LABEL INSTR. PARAMETER(s) ADR/OPCODE ASCII

the following is placed at \$7F00 by the X command.

;Starts at \$7F00

;Loads both syncro code and ramcharger code (if ramcharger is present).

; SDX version only loads syncro code. several changes made to code for SDX version

; are noted. SDX version doesn't have ramcharger code segments.

ORG 7F00H

DI ; 7F00 F3 .
LD HL, (\$7FE3) ; 7F01 2AE37F reset ram
LD (\$7804),HL ; 7F04 220478 vectors
LD (\$7807),HL ; 7F07 220778 to \$CA0
LD (\$780A),HL ; 7F0A 220A78 ".x
LD HL, (\$7FD7) ; 7F0D 2AD77F \$7840 pointer to free ram
LD DE, (\$7FE1) ; 7F10 ED5BE17F \$7B84
LD (HL),E ; 7F14 73 s
INC HL ; 7F15 23 #
LD (HL),D ; 7F16 72 r
EI ; 7F17 FB .
LD HL, \$7FCD ; 7F18 21CD7F syncro load flag
LD A, (HL) ; 7F1B 7E ~
INC (HL) ; 7F1C 34 add 1 to syncro flag
OR A ; 7F1D B7 set flags
JR Z, Lb0b ; 7F1E 2864 go to syncro load

;the following 6 bytes have been NOP'ed for SDX version of syncromesh

DEC A ; 7F20 3D nop for sdx
JR Z, Lb888 ; 7F21 2833 nop nop for sdx
DEC A ; 7F23 3D nop for sdx.
JR Z, Lb0b ; 7F24 285E nop nop for sdx

DI ; 7F26 F3 .
LD HL, (\$7FD7) ; 7F27 2AD77F \$7840
LD DE, (\$7FD9) ; 7F2A ED5BD97F \$7eed
LD (HL),E ; 7F2E 73 s
INC HL ; 7F2F 23 #
LD (HL),D ; 7F30 72 r
LD A, (\$7FCE) ; 7F31 3ACE7F
OR A ; 7F34 B7 .
JR Z, Lb1b ; 7F35 2814 (.
LD HL, (\$7FDD) ; 7F37 2ADD7F FBD9
LD (\$7807),HL ; 7F3A 220778 ".x
LD HL, (\$7FDF) ; 7F3D 2ADF7F FC12
LD (\$780A),HL ; 7F40 220A78 ".x
LD HL, (\$7FDB) ; 7F43 2ADB7F FC31
LD (\$7804),HL ; 7F46 220478 ".x

JR Lb2c ; 7F49 1806 ..
Lb1b: LD HL, (\$7FE1) ; 7F4B 2AE17F 7b84. Start of syncro memory
LD (\$780A),HL ; 7F4E 220A78 Place command tie in
Lb2c: EI ; 7F51 FB .
LD A, \$43 ; 7F52 3E43 >C
SCF ; 7F54 37 7
RET ; 7F55 C9 .

Lb888: LD HL, (\$7FD5) ; 7F56 2AD57F 2f5 change pointers
LD (\$7FD1),HL ; 7F59 22D17F to load the ramcharger
LD HL, (\$7FD3) ; 7F5C 2AD37F FBD9 code
LD (\$7FCF),HL ; 7F5F 22CF7F " ..

```

LD      A, (STAT1R)      ; 7F62 3A0010    :..
LD      A, (STAT1)      ; 7F65 3A0110    :..
AND     $10              ; 7F68 E610      Track button
JR      NZ, Lb3c         ; 7F6A 2010      if track button held, dont lo
LD      HL, $FFFF       ; 7F6C 21FFFF    !..
LD      B, (HL)         ; 7F6F 46        F
XOR     A                ; 7F70 AF        zero A
Lb4c:   LD      (HL), A   ; 7F71 77        test if ramcharger is present
        CP      (HL)     ; 7F72 BE        .
        JR      NZ, Lb3c ; 7F73 2007      .
        INC     A        ; 7F75 3C        <
        JR      NZ, Lb4c ; 7F76 20F9      .
        LD      (HL), B  ; 7F78 70        p
        DEC     A        ; 7F79 3D        set a to $FF
        JR      Lb5c     ; 7F7A 1801      ..
Lb3c:   XOR     A        ; 7F7C AF        zero a
Lb5c:   LD      ($7FCE), A ; 7F7D 32CE7F   save a to ram present flag
        LD      A, $43   ; 7F80 3E43     >C
        SCF     ; 7F82 37        Set Carry Flag
        RET     ; 7F83 C9        .

Lb0b:   LD      DE, ($7FD1) ; 7F84 ED5BD17F  369
        LD      HL, ($7FCF) ; 7F88 2ACF7F   B784
        PUSH   HL        ; 7F8B E5        .
        LD      A, E     ; 7F8C 7B        {
        OR     A        ; 7F8D B7        .
        JR      Z, Lb6c  ; 7F8E 280B     (.
        LD      B, A     ; 7F90 47        G
        LD      E, $00   ; 7F91 1E00     ..
        ADD    A, L     ; 7F93 85        .
        LD      L, A     ; 7F94 6F        o
        LD      A, H     ; 7F95 7C        |
        ADC    A, $00   ; 7F96 CE00     ..
        LD      H, A     ; 7F98 67        g
        JR      Lb7c     ; 7F99 1804     ..
Lb6c:   LD      B, $00   ; 7F9B 0600     ..
        DEC     D        ; 7F9D 15        .
        INC     H        ; 7F9E 24        $
Lb7c:   EX     (SP), HL  ; 7F9F E3        .
        PUSH   DE        ; 7FA0 D5        .
        EX     DE, HL    ; 7FA1 EB        .
        LD      C, $07   ; 7FA2 0E07     de=buffer, b=number to get. c=
        CALL   Lba       ; 7FA4 CD0400   Jump to routine #7 Get sio Blo
        POP    BC        ; 7FA7 C1        .
        POP    HL        ; 7FA8 E1        .
        JR      Z, Lb9c  ; 7FA9 2808     (.
        LD      A, ($7FCE) ; 7FAB 3ACE7F   :..
        OR     A        ; 7FAE B7        set flags
        LD      A, $4E   ; 7FAF 3E4E     Nak
        JR      NZ, Lb10a ; 00B1 2018     .
Lb9c:   LD      ($7FCF), HL ; 7FB3 22CF7F   "..
        LD      ($7FD1), BC ; 7FB6 ED43D17F .C..
        LD      A, B     ; 7FBA 78        x
        OR     C        ; 7FBB B1        .
        JR      Z, Lb11c ; 7FBC 2804     finished loading syncro code?
        LD      HL, $7FCD ; 7FBE 21CD7F   no
        DEC    (HL)     ; 7FC1 35        dec flag back to one
Lb11c: LD      A, $41   ; 7FC2 3E41     yes load 'A'ck
        LD      C, $06   ; 7FC4 0E06     routine 6 send byte

```

```

CALL    Lba                ; 7FC6 CD0400    go do do it
LD      A,$43              ; 7FC9 3E43    load 'C'omplete
Lb10a  SCF                  ; 7FCB 37     set carry flag
RET     ; 7FCC C9     .

NOP     ; 7FCD 00     flag for syncro code or ramcha
NOP     ; 7FCE 00     ramcharger ram present flag
DB $84,$7B ; 7FCF 847B    start of syncro code .
DB $69,$03 ; 7FD1 6903    length of syncro code
DB $D9,$FB ; 7FD3 D9FB    start of ramcharger code.
DB $F5,$02 ; 7FD5 F502    length for ramcharger code. ze
DB $40,$78 ; 7FD7 4078    word Pointer for free ram star
DB $ED,$7E ; 7FD9 ED7E    word end of syncro code +1. fr
DB $31,$FC ; 7FDB 31FC    word jump to ram
DB $D9,$FB ; 7FDD D9FB    word Jump address Main wait lo
DB $12,$FC ; 7FDF 12FC    Word Jump address for ram vect
; over writes one below if ramcharger is present
DB $84,$7B ; 7FE1 847B    word Jump address for ram vect
DB $A0,$0C ; 7FE3 A00C    word Points to RET instruction

```

Lb12:

```

;?????????????????????????????????????????????????????????ø
;? DISASSEMBLED FILE - DONE WITH DISASM 1.0. ?
;? Date: 30-07-2008 Time: 18:20 ?
;? (c) 1996 Channex aka Lasse S. Tassing ?
;? Email: ltassing.ite.dk ?
;¿?????????????????????????????????????????????????????????

```

```

; ?[CODE]?????????????????????????????????????????????????????????
; LABEL INSTR. PARAMETER(s) ADR/OPCODE ASCII
;This code is loaded in to the ramcharger ram.
ORG FBD9H

```

; 7807 external main loop extra code

```

LD      L,$60                ; FBD9 2E60    .`
LD      A,(STAT1)           ; FBDB 3A0110  :..
AND     L                    ; FBDE A5     .
CP      L                    ; FBDF BD     id and error button
JR      NZ,Lb0d              ; FBE0 200F    .
LD      A,($7836)            ; FBE2 3A3678  :6x
OR      L                    ; FBE5 B5     .
LD      ($7836),A           ; FBE6 323678  26x
LD      A,$FF                ; FBE9 3EFF    >.
LD      ($7819),A           ; FBEB 321978  2.x
LD      ($781A),A           ; FBEE 321A78  2.x
Lb0d:  LD      HL,$FECD       ; FBF1 21CDFE  !..
LD      A,(HL)              ; FBF4 7E     ~
INC     A                    ; FBF5 3C     <
JR      Z,Lb1d               ; FBF6 2810    (.
DEC     A                    ; FBF8 3D     =
JR      NZ,Lb2d              ; FBF9 2006    .
LD      A,($7819)            ; FBFB 3A1978  :.x
OR      A                    ; FBFE B7     .
JR      Z,Lb3d               ; FBFF 2810    (.
Lb2d:  LD      (HL),$FF       ; FC01 36FF    6.
LD      A,$00                ; FC03 3E00    >.
LD      ($7837),A           ; FC05 323778  27x
Lb1d:  LD      HL,$838E       ; FC08 218E83    bF for front leds = track

```

```

LD      (TLEDNUM1),HL      ; FC0B 223A78  ":x
CALL    UPLED-1-2-B-E     ; FC0E CD120B  ...
Lb3d:   RET                ; FC11 C9      .
;-----
;      780a external command tie in
LD      A,C                ; FC12 79      y
AND     $7F                 ; FC13 E67F   remove high bit
CP      $4F                 ; FC15 FE4F   .0 command 0
JR      Z,Lb5d              ; FC17 280C   (.
CP      $24                 ; FC19 FE24   .$
JR      C,Lb5d              ; FC1B 3808   8.
CP      $57                 ; FC1D FE57   .WRITE
JR      NZ,Lb6d             ; FC1F 2009   .
LD      A,$FF               ; FC21 3EFF   >.
JR      Lb7d                ; FC23 1806   ..
Lb5d:   LD      A,$FF       ; FC25 3EFF   >.
LD      ($FECD),A          ; FC27 32CDFE 2..
Lb6d:   XOR     A           ; FC2A AF     zero A
Lb7d:   LD      ($FED0),A   ; FC2B 32D0FE set flag
JP      Lbb                ; FC2E C3847B jump to syncro code
;-----
;tie in for something 7803
POP     HL                  ; FC31 E1     .
LD      ($FECE),HL         ; FC32 22CEFE "...
PUSH    AF                 ; FC35 F5     .
LD      A,($FECD)          ; FC36 3ACDFE :..
OR      A                  ; FC39 B7     .
JR      Z,Lb9d              ; FC3A 284D   (M
CALL    Lb36                ; FC3C CD6005 .`.
CALL    Lb58                ; FC3F CDB40A get bytes per sector in B
LD      L,B                 ; FC42 68     h
DEC     L                   ; FC43 2D     -1
LD      H,$00               ; FC44 2600   &.
INC     HL                  ; FC46 23     make 8 bit number into 16bit n
LD      ($FED1),HL         ; FC47 22D1FE "...
ADD     HL,HL               ; FC4A 29     *2
LD      E,L                 ; FC4B 5D     save *2
LD      D,H                 ; FC4C 54     T
ADD     HL,HL               ; FC4D 29     *4
ADD     HL,HL               ; FC4E 29     *8
ADD     HL,HL               ; FC4F 29     *16
ADD     HL,DE               ; FC50 19     add *2 for total of * 18
LD      ($FED3),HL         ; FC51 22D3FE "...
LD      A,$12               ; FC54 3E12   sect per track
LD      ($FED5),A          ; FC56 32D5FE 2..
XOR     A                   ; FC59 AF     zero A
LD      ($FED6),A          ; FC5A 32D6FE 2..
LD      ($FF69),A          ; FC5D 3269FF 2i.
EX      DE,HL               ; FC60 EB     .
LD      HL,$FBD9            ; FC61 21D9FB start of this code
LD      IX,$FED8            ; FC64 DD21D8FE end of this code
LD      BC,$03              ; FC68 010300 ...
Lb13d: SBC     HL,DE         ; FC6B ED52   .R
JR      C,Lb12d             ; FC6D 3813   8.
LD      (IX+$00),$FF        ; FC6F DD3600FF .6..
INC     IX                  ; FC73 DD23   .#
LD      (IX+$00),L          ; FC75 DD7500 .u.
INC     IX                  ; FC78 DD23   .#
LD      (IX+$00),H          ; FC7A DD7400 .t.

```

	ADD	IX,BC	; FC7D DD09	..
	INC	A	; FC7F 3C	<
	JR	Lb13d	; FC80 18E9	..
Lb12d:	LD	(\$FED7),A	; FC82 32D7FE	2..
	XOR	A	; FC85 AF	.
	LD	(\$FECD),A	; FC86 32CDFE	2..
Lb9d:	POP	AF	; FC89 F1	.
	OR	A	; FC8A B7	.
	JR	NZ,Lb14d	; FC8B 2005	.
	CALL	Lb15d	; FC8D CD96FC	...
	JR	Lb16d	; FC90 1803	..
Lb14d:	CALL	Lb17d	; FC92 CDABFC	...
Lb16d:	RET		; FC95 C9	.
Lb15d:	LD	A,(\$FED0)	; FC96 3AD0FE	:..
	OR	A	; FC99 B7	.
	JR	NZ,Lb18d	; FC9A 200A	.
	CALL	Lb19d	; FC9C CDC1FC	...
	JR	NZ,Lb18d	; FC9F 2005	.
	LDIR		; FCA1 EDB0	..
	XOR	A	; FCA3 AF	.
	JR	Lb20d	; FCA4 1804	..
Lb18d:	XOR	A	; FCA6 AF	zero A
	CALL	Lb21d	; FCA7 CDC9FE	...
Lb20d:	RET		; FCAA C9	.
Lb17d:	CALL	Lb21d	; FCAB CDC9FE	...
	JR	NZ,Lb22d	; FCAE 2010	.
	CALL	Lb19d	; FCB0 CDC1FC	...
	JR	NZ,Lb23d	; FCB3 200A	.
Lb24d:	LD	A,(DE)	; FCB5 1A	.
	CPL		; FCB6 2F	/
	LD	(HL),A	; FCB7 77	w
	INC	DE	; FCB8 13	.
	INC	HL	; FCB9 23	#
	DEC	BC	; FCBA 0B	.
	LD	A,B	; FCBB 78	x
	OR	C	; FCBC B1	.
	JR	NZ,Lb24d	; FCBD 20F6	.
Lb23d:	XOR	A	; FCBF AF	.
Lb22d:	RET		; FCC0 C9	.
Lb19d:	LD	A,(TRACKNUM)	; FCC1 3A2278	:"x
	OR	A	; FCC4 B7	.
	JR	NZ,Lb25d	; FCC5 200D	.
	LD	A,(SECTORN)	; FCC7 3A2378	:#x
	DEC	A	; FCCA 3D	=
	JR	NZ,Lb25d	; FCCB 2007	.
	LD	A,(\$FF69)	; FCCD 3A69FF	:i.
	OR	A	; FCD0 B7	.
	JP	NZ,Lb26	; FCD1 C2D9FD	...
Lb25d:	CALL	Lb27d	; FCD4 CDEEFD	...
	JR	Z,Lb28d	; FCD7 2861	(a
	LD	HL,\$FED6	; FCD9 21D6FE	!..
	LD	A,(HL)	; FCDC 7E	~
	OR	A	; FCDD B7	.
	JR	NZ,Lb29d	; FCDE 2001	.
	INC	(HL)	; FCE0 34	4
Lb29d:	LD	A,(\$FED8)	; FCE1 3AD8FE	:..

	INC	A	; FCE4 3C	<
	JR	Z,Lb30d	; FCE5 284B	(K
	LD	A,(\$FED7)	; FCE7 3AD7FE	:..
	LD	C,A	; FCEA 4F	O
	LD	B,\$00	; FCEB 0600	..
	LD	L,C	; FCED 69	i
	LD	H,B	; FCEE 60	^
	ADD	HL,HL	; FCEF 29)
	ADD	HL,HL	; FCF0 29)
	ADD	HL,BC	; FCF1 09	.
	LD	C,L	; FCF2 4D	M
	LD	B,H	; FCF3 44	D
	LD	DE,\$FED7	; FCF4 11D7FE	...
	ADD	HL,DE	; FCF7 19	.
	PUSH	HL	; FCF8 E5	.
	LD	DE,\$05	; FCF9 110500	...
	ADD	HL,DE	; FCFC 19	.
	EX	DE,HL	; FCFD EB	.
	POP	HL	; FCFE E1	.
	PUSH	HL	; FCFF E5	.
	DI		; FD00 F3	.
	LDDR		; FD01 EDB8	..
	INC	HL	; FD03 23	#
	LD	(HL),\$FF	; FD04 36FF	6.
	INC	HL	; FD06 23	#
	EX	DE,HL	; FD07 EB	.
	POP	HL	; FD08 E1	.
	INC	HL	; FD09 23	#
	INC	HL	; FD0A 23	#
	LDI		; FD0B EDA0	..
	LDI		; FD0D EDA0	..
	LD	HL,\$FED6	; FD0F 21D6FE	!..
	LD	A,(\$FED7)	; FD12 3AD7FE	:..
	DEC	A	; FD15 3D	=
	CP	(HL)	; FD16 BE	.
	JR	C,Lb31d	; FD17 3801	8.
	INC	(HL)	; FD19 34	4
Lb31d:	LD	HL,(\$FEDE)	; FD1A 2ADEFE	*..HL=source,DE= dest
	LD	DE,(\$FED9)	; FD1D ED5BD9FE	[..BC=bytes to move for LDIR
	LD	(\$FED9),HL	; FD21 22D9FE	"..
	LD	(\$FEDE),DE	; FD24 ED53DEFE	.S..
	LD	BC,\$FED3	; FD28 ED4BD3FE	.K..
	OUT	(\$0F),A	; FD2C D30F	enable ram \$0000,\$7FFF
	LDIR		; FD2E EDB0	..
	OUT	(\$0E),A	; FD30 D30E	disable ram \$0000,\$7FFF
Lb30d:	CALL	Lb32d	; FD32 CD06FE	...
	JP	NZ,Lb33d	; FD35 C2EDFD	...
	JR	Lb34d	; FD38 184C	.L
Lb28d:	PUSH	HL	; FD3A E5	.
	CALL	Lb35d	; FD3B CDBAFE	...
	POP	HL	; FD3E E1	.
	LD	A,(\$FED8)	; FD3F 3AD8FE	:..
	LD	B,A	; FD42 47	G
	LD	A,(TRACKNUM)	; FD43 3A2278	:"x
	CP	B	; FD46 B8	.
	JR	Z,Lb36d	; FD47 2844	(D
	LD	E,L	; FD49 5D]
	LD	D,H	; FD4A 54	T

	DEC	DE	; FD4B 1B	.
	INC	HL	; FD4C 23	#
	LD	C, (HL)	; FD4D 4E	N
	INC	HL	; FD4E 23	#
	LD	B, (HL)	; FD4F 46	F
	PUSH	BC	; FD50 C5	.
	INC	HL	; FD51 23	#
	LD	C, (HL)	; FD52 4E	N
	INC	HL	; FD53 23	#
	LD	B, (HL)	; FD54 46	F
	PUSH	BC	; FD55 C5	.
	PUSH	HL	; FD56 E5	.
	LD	BC, \$FEDC	; FD57 01DCFE	...
	OR	A	; FD5A B7	.
	SBC	HL, BC	; FD5B ED42	.B
	LD	C, L	; FD5D 4D	M
	LD	B, H	; FD5E 44	D
	POP	HL	; FD5F E1	.
	EX	DE, HL	; FD60 EB	.
	DI		; FD61 F3	.
	LDDR		; FD62 EDB8	..
	POP	HL	; FD64 E1	.
	LD	(\$FEDB), HL	; FD65 22DBFE	"..
	POP	HL	; FD68 E1	.
	PUSH	HL	; FD69 E5	.
	POP	IX	; FD6A DDE1	..
	LD	(\$FEDE), HL	; FD6C 22DEFE	"..
	LD	DE, (\$FED9)	; FD6F ED5BD9FE	.[..
	LD	BC, \$FED3	; FD73 ED4BD3FE	.K..
	OUT	(\$0F), A	; FD77 D30F	..
Lb37d:	LD	A, (DE)	; FD79 1A	.
	LDI		; FD7A EDA0	..
	LD	(IX+\$00), A	; FD7C DD7700	.w.
	INC	IX	; FD7F DD23	.#
	JP	PE, Lb37d	; FD81 EA79FD	.y.
	OUT	(\$0E), A	; FD84 D30E	..
Lb34d:	LD	A, (TRACKNUM)	; FD86 3A2278	:"x
	LD	(\$FED8), A	; FD89 32D8FE	2..
	EI		; FD8C FB	.
Lb36d:	LD	A, (\$FED5)	; FD8D 3AD5FE	:..
	LD	B, A	; FD90 47	G
	LD	HL, (\$FEDB)	; FD91 2ADBF	*..
	LD	IX, (\$FED9)	; FD94 DD2AD9FE	*..
	LD	DE, (\$FED1)	; FD98 ED5BD1FE	.[..
	LD	A, (SECTORN)	; FD9C 3A2378	:#x
	LD	C, A	; FD9F 4F	O
Lb40d:	LD	A, (HL)	; FDA0 7E	~
	CP	C	; FDA1 B9	.
	JR	Z, Lb38d	; FDA2 2819	(.
	RLCA		; FDA4 07	.
	JR	C, Lb39d	; FDA5 3807	8.
	ADD	IX, DE	; FDA7 DD19	..
	INC	HL	; FDA9 23	#
	DJNZ	Lb40d	; FDAA 10F4	..
	JR	Lb33d	; FDAC 183F	.?
Lb39d:	LD	A, (\$FED5)	; FDAE 3AD5FE	:..
	LD	E, A	; FDB1 5F	-
	LD	D, \$00	; FDB2 1600	..
	OR	A	; FDB4 B7	.

```

SBC      HL,DE          ; FDB5 ED52      .R
LD       DE,($FED1)    ; FDB7 ED5BD1FE  [...
JR       Lb40d         ; FDBB 18E3      ..
Lb38d: LD       A,(TRACKNUM) ; FDBD 3A2278   : "x
OR       A             ; FDC0 B7        .
JR       NZ,Lb41d     ; FDC1 2020     .
LD       A,(SECTORN)  ; FDC3 3A2378   : #x
DEC     A             ; FDC6 3D        =
JR       NZ,Lb41d     ; FDC7 201A     .
PUSH    IX           ; FDC9 DDE5     ..
POP     HL           ; FDCB E1        .
LD      BC,$80       ; FDCC 018000   ...
LD      DE,$FF6A     ; FDCF 116AFF   .j.
LDIR    A,$FF        ; FDD2 EDB0     ..
LD      ($FF69),A    ; FDD4 3EFF     >.
Lb26d: CALL Lb35d     ; FDD9 CDBAFE   ...
LD      IX,$FF6A     ; FDDC DD216AFF !.j.
LD      DE,$80       ; FDE0 118000   ...
Lb41d: PUSH IX       ; FDE3 DDE5     ..
POP     HL           ; FDE5 E1        .
LD      C,E          ; FDE6 4B        K
LD      B,D          ; FDE7 42        B
LD      DE,(SECTORBUF) ; FDE8 ED5B2478 .[$x
XOR     A            ; FDEC AF        .
Lb33d: RET           ; FDED C9        .

Lb27d: LD      DE,$05 ; FDEE 110500   ...
LD      HL,$FED8     ; FDF1 21D8FE   !..
LD      A,($FED6)    ; FDF4 3AD6FE   :..
LD      B,A          ; FDF7 47        G
ADD     A,$FF        ; FDF8 C6FF     ..
JR      NC,Lb42d     ; FDFA 3009     0.
LD      A,(TRACKNUM) ; FDFC 3A2278   : "x
Lb43d: CP      (HL)   ; FDFE BE        .
JR      Z,Lb42d     ; FE00 2803     (.
ADD     HL,DE        ; FE02 19        .
DJNZ   Lb43d        ; FE03 10FA     ..
Lb42d: RET           ; FE05 C9        .

Lb32d: CALL Lb15     ; FE06 CDBE08   turn motor on
CALL Lb63           ; FE09 CDF407   step to track
LD      A,$05       ; FE0C 3E05     >.
LD      (CDLOOP2),A ; FE0E 322878   2(x
Lb52d: LD      A,($780F) ; FE11 3A0F78   :.x
LD      HL,$FFEA    ; FE14 21EAFD   !..
CALL Lb111         ; FE17 CDD805   read sector data
JR      NZ,Lb47d    ; FE1A 2005     .
CALL Lb14          ; FE1C CD3509   reset FDC
JR      Lb49d       ; FE1F 1878     .x
Lb47d: CALL Lb80     ; FE21 CD6809   wait for FDC
AND     $1C         ; FE24 E61C     ..
JR      Z,Lb51d     ; FE26 2808     (.
LD      HL,CDLOOP2 ; FE28 212878   !(x
DEC     (HL)        ; FE2B 35        5
JR      NZ,Lb52d    ; FE2C 20E3     .
JR      Lb49d       ; FE2E 1869     .i
Lb51d: LD      A,($7815) ; FE30 3A1578   :.x
CPL     A           ; FE33 2F        /

```

	AND	\$04	; FE34 E604	..
	LD	B,A	; FE36 47	G
	LD	A, (\$DENFLG)	; FE37 3A1778	:.x
	ADD	A,A	; FE3A 87	.
	OR	B	; FE3B B0	.
	LD	L,A	; FE3C 6F	o
	LD	H,\$00	; FE3D 2600	&.
	LD	DE,\$7D88	; FE3F 11887D	..}
	ADD	HL,DE	; FE42 19	.
	LD	E,(HL)	; FE43 5E	^
	INC	HL	; FE44 23	#
	LD	D,(HL)	; FE45 56	V
	EX	DE,HL	; FE46 EB	.
	LD	A, (\$FFEC)	; FE47 3AECFF	:..
	LD	B,A	; FE4A 47	G
Lb54d:	LD	A,(HL)	; FE4B 7E	~
	INC	HL	; FE4C 23	#
	CP	B	; FE4D B8	.
	JR	Z,Lb53d	; FE4E 2805	(.
	RLCA		; FE50 07	.
	JR	NC,Lb54d	; FE51 30F8	0.
	JR	Lb49d	; FE53 1844	.D
Lb53d:	LD	(\$FEDB),HL	; FE55 22DBFE	"..
	PUSH	HL	; FE58 E5	.
	CALL	Lb76	; FE59 CDB807	load track to FDC
	POP	IX	; FE5C DDE1	..
	LD	HL, (\$FED9)	; FE5E 2AD9FE	*..
Lb57d:	LD	A,(IX+\$00)	; FE61 DD7E00	..~.
	BIT	7,A	; FE64 CB7F	..
	JR	Z,Lb56d	; FE66 280C	(.
	LD	A, (\$FED5)	; FE68 3AD5FE	:..
	CPL		; FE6B 2F	/
	LD	E,A	; FE6C 5F	_
	LD	D,\$FF	; FE6D 16FF	..
	INC	DE	; FE6F 13	.
	ADD	IX,DE	; FE70 DD19	..
	JR	Lb57d	; FE72 18ED	..
Lb56d:	LD	(SECTORFDC),A	; FE74 320260	2.`
	LD	A,\$05	; FE77 3E05	>.
	LD	(CDLOOP2),A	; FE79 322878	2(x
	PUSH	HL	; FE7C E5	.
Lb62d:	POP	HL	; FE7D E1	.
	PUSH	HL	; FE7E E5	.
	CALL	Lb78	; FE7F CDD505	read sector
	JR	Z,Lb59d	; FE82 2805	(.
	CALL	Lb80	; FE84 CD6809	wait for fdc
	JR	Lb60d	; FE87 1805	..
Lb59d:	CALL	Lb14	; FE89 CD3509	reset FDC
	OR	\$10	; FE8C F610	..
Lb60d:	AND	\$1C	; FE8E E61C	..
	JR	Z,Lb61d	; FE90 280B	(.
	LD	HL,CDLOOP2	; FE92 212878	!(x
	DEC	(HL)	; FE95 35	5
	JR	NZ,Lb62d	; FE96 20E5	.
	POP	HL	; FE98 E1	.
Lb49d:	OR	\$FF	; FE99 F6FF	..
	JR	Lb63d	; FE9B 181C	..
Lb61d:	POP	BC	; FE9D C1	.
	LD	B,(IX+\$00)	; FE9E DD4600	.F.

```

INC      IX                ; FE A1 DD23      .#
LD       A, ($FFEC)       ; FE A3 3AECFF   :..
CP       B                ; FE A6 B8       .
JR       NZ, Lb57d        ; FE A7 20B8     .
LD       BC, $FED3       ; FE A9 ED4BD3FE .K..
LD       HL, ($FED9)     ; FE AD 2AD9FE   *..
Lb64d:  LD       A, (HL)   ; FE B0 7E       ~
        CPL                ; FE B1 2F       /
        LD       (HL), A   ; FE B2 77       w
        INC      HL        ; FE B3 23       #
        DEC      BC        ; FE B4 0B       .
        LD       A, B      ; FE B5 78       x
        OR       C         ; FE B6 B1       .
        JR       NZ, Lb64d ; FE B7 20F7     .
Lb63d:  RET                ; FE B9 C9       .

Lb35d:  LD       A, (TRACKNUM) ; FE BA 3A2278   : "x
        SRL      A         ; FE BD CB3F     .?
        CALL    Lb134      ; FE BF CDC30A   update track led display
        LD       (TLEDNUM1), HL ; FE C2 223A78   " :x
        CALL    UPLED-1-2-B-E ; FE C5 CD120B   ...
        RET                ; FE C8 C9       .

Lb21d:  LD       HL, ($FECE) ; FE C9 2ACEFE   *..
        JP      (HL)      ; FE CC E9       .

        DB $FE          ; FE CD FE

```