dataspeed

THE CONDUCTOR™

Technical Manual

dataspeed INC

1302 NOE STREET, SAN FRANCISCO, CALIFORNIA 94131

TELEPHONE (415) 282 - 5616

641-8947

ARDEX DISTRIBUTING CO.
A DIVISION OF PARABOLA ELECTRONICS INC.
31 PROGRESS COURT, UNIT 12
SCARBOROUGH, ONT. M1G 3V5
TEL: 439-0474

The DATASPEED Conductor™ provides control of up to eight Shugart Assoc. compatible disk drives. It uses a Western Digital WD1971 LSI controller chip and a phase-locked loop. Either 8″ or 5¼″ disk drives are supported, depending on which of two supplied header plugs is inserted into the board. Timing is generated by an on-board 8MHZ crystal making CPU clock frequency unimportant.

The board comes fully assembled, tested and warranteed. However, the first section of this manual is dedicated to explaining the functions of the various circuit blocks on the Conductor. The second section provides a detailed description of the software required to operate the board.

WARRANTY

YOUR CONDUCTOR is warrantied against defects for 90 days from purchase. Any unit returned to **DATASPEED** within this period will be repaired or replaced without charge, providing there is no obvious electrical or mechanical damage. No other warranty is made or implied. **DATASPEED** will not be responsible for consequential damages.

1. HARDWARE DESCRIPTION

For the purpose of this discussion, the board is divided into four major functional blocks. These are the CPU Interface, the LSI Controller circuit, the Phase-Locked Loop and Disk Interface. Each block is treated in its own sub-section. Refer to the drawing in section 1.5 and component layout in section 1.6.

1.1 CPU INTERFACE

The CPU Interface consists of address and control decoding which provides selection of the various blocks on the board. It also includes the wait logic and interrupt logic.

1.1.1 ADDRESS DECODING

The board uses memory mapped I/O for the WD1791 registers and an I/O Port for drive control. The decoding logic for both is provided by IC's 28 and 29. Lines connected to IC 28 will be active when low and those connected to IC 29 will be active high. Although the board is permanently addressed at page F0 hex, for special purposes this page address can be changed by cutting these traces and reconfiguring them.

For simplicity, the board decodes the address without concern for A6 and A7. This means that an entire page of memory is dedicated to the disk controller. Refer to Table I for a breakdown of the address assignments. Address lines A2, A3 and A4 are of concern only to the Boot Prom. A5 is used to choose between the Prom and I/O. This scheme may seem wasteful until one considers that memory is generally not assignable in less than 256-byte blocks.

The page address signal thus far decoded is used to gate the various bus control signals to form chip selects 'LATCH STATUS' (LS), 'READ STATUS' (RS), 'PROM' (PROM), and the WD1791 'READ ENABLE' (RE) and 'WRITE ENABLE' (WE). The WD1791 chip select (CS) is permanently enabled to allow faster access. Since the on-board data bus is bi-directional, 'BUS DIRECTION' (BSDR) is formed by 'oring' all the input chip selects (RE,PROM,RS).

1.1.2 WAIT LOGIC

The INTERFACE LOGIC is designed to allow two types of data transfers to occur. The first is the conventional method where a status bit is polled to determine when the interface is ready followed by a read or write to the data port. This method is used in all transfers involving the control/status registers. In most cases, these registers can be assumed ready to transfer data and no status checking is necessary. The data register, however, offers both the conventional method and an optional faster method. When the wait logic is enabled, the data port can be read or written without regard for the condition of the DATA REQUEST (DRQ) flag. If DRQ is not active during a read or write to memory location F023 hex, the PRDY line is lowered holding the processor in a wait state until either DRQ or INTRQ (in the case of an error condition) releases it. This permits the use of an extremely small software loop capable of speeds required by the double density transfer rate. Two "one-shots" are used to ensure that the timing of the WD1791 control lines is processor independent.

1.1.3. INTERRUPT LOGIC

At the completion of each command, the WD1791 activates its INTRQ line. This line is gated by control bit 1 and presented to the PINT line of the processor. As long as the control bit is disabled, no interrupts will occur. When enable, however, INTRQ is effectively tied to the interrupt line of the processor. This function eliminates the need to poll for command completion in the data transfer loop further improving speed.

It should be noted that some users of the WD1791 have encountered difficulty in that the INTRQ generated by the WD1791 at the end of a seek command is either too short or non-existant. DATA-SPEED recommends polling for seek completion for this reason. No other INTRQ's are known to cause problems. Also, since the INTRQ Line is normally active, interrupts should only be enabled after a command has been issued to the 1791.

1.1.4 CONTROL AND STATUS

The CONDUCTOR contains two control registers and two status registers. One is for drive control and the other controls the WD1791. This section will cover the drive control and status register. Refer to section 1.2.1. for information on the WD1791 registers.

The drive control and status register (DCS) is the only I/O mapped register on the board. It is implemented as I/O Port F0 hex and consists of IC's 34 and 36. Refer to Table II for a summary of bit assignments.

The wait logic is controlled by Bit 0 of this register. By writing a '1' into this bit, the wait logic is disabled, permitting normal transfers to occur into and out of the data register. If this bit is '0', the wait logic is enabled and the board will enter a wait state if the data register is read or written and no DRQ is present. Certain board commands do not produce DRQ's such as HOME, SEEK, STEP IN and STEP OUT. Of these, only the SEEK command uses the data register. Therefore, if the wait logic is enabled and an attempt is made to store a track number into the data register in preparation for a seek command, the computer will enter a wait state which can only be exited by a reset. To prevent this, only enable the wait logic when necessary (during read or write commands).

Bit 1 is the interrupt mask bit. When this bit is a '1', an INTRQ will cause an interrupt request to be placed on the PINT bus line. If this bit is '0', interrupts are inhibited.

Bit 2 is connected to the HLT pin of the WD1791 and PLL control logic. It is used to cause the controller to wait for a certain period of time after a read or write command is issued before the transfer actually begins. The usual sequence is as follows:

Bit 2 is normally a '1' which causes the PLL to lock onto the write oscillator clock. It also inhibits the WD1791 from performing a data transfer. The HDLD status bit is checked to see if the head is currently loaded. If it is, the command (READ or WRITE) is given and HLT is set low. This allows the PLL to acquire phase synchronization with the data from the disk and also allows the controller chip to begin sampling this data. If HDLD is not active, the command is issued followed by a software delay loop which provides the required head load time. Please note that in a standard Shugart 8" drive, the head is loaded whenever the drive is selected. In this case, only a drive select operation requires the head load timeout. The drive can be easily modified, however, to recognize the HDLD signal, in which case the sequence stated above should be noted. Also note that the Shugart 5¼" drives use the HDLD signal to turn the motor on and off. If it is required to have the motor on continually, the drive must be modified. See the respective OEM manuals for these procedures.

Bit 3 is used to select the side on those drives offering double sided capability. When set to '1', side B is selected. When set to '0', side A is selected.

Bits 4, 5 and 6 are DRIVE SELECTS DS1, DS2 and DS3 respectively. Using the standard drive, these lines permit up to three double-sided drives to be used without any further modifications. If necessary, more drives can be added by using the encoded drive select option (Refer to the OEM manual). Also, if only single-sided disks are used, the side bit can become DRIVE SELECT 4 (DS4) by jumping to the proper connector pin.

Bit 7 is used to control the recording density. If this bit is '1', the recording method is FM (single density). When cleared, this bit causes the MFM recording method (double density) to be used. All timing adjustments are made automatically when this bit is changed.

The status bits 2, 3, 4, 5, 6 and 7 are a reflection of the condition of the HDLD line and control bits 3, 4, 5, 6, 7. This allows the use of masking operations to change control bits. Status bit 0 is the DRQ line from the WD1791 line. Status bit 1 is the INTRQ signal.

CONDUCTOR REGISTER MAP

MEMORY MAPPED

F000 - F01F ..... Boot Prom
F020 ..... WD1791 Control/Status
F021 ..... WD1791 Track Register
F022 ..... WD1791 Sector Register
F023 ..... WD1791 Data Register

I/O MAPPED

F0 ..... Disk Control/Status

NOTE: All other addresses in page F0 are reserved.

TABLE I

1.2 WESTERN DIGITAL FD1791

The heart of the CONDUCTOR is the FD1791 controller circuit from Western Digital. This circuit performs the head positioning, formatting, CRC checking and generation and address mark detection for the CONDUCTOR. It is capable of IBM 3740 and System 34 compatability which use FM and MFM

I/O PORT USAGE

CONTROL

BIT	MEANING	
0	Wait Logic Control	1 — disable 0 — enable
1	Int Logic Control	1 — enable 0 — disable
2	HLT	
3	Side Select	1 — side B 0 — side A
4	Drive Select 3 (DS3)	0 — select 1 — de-select
5	Drive Select 2 (DS2)	0 — select 1 — de-select
6	Drive Select 1 (DS1)	0 — select 1 — de-select
7	Double-Density Enable	0 — double 1 — single

STATUS

BIT	MEANING	
0	DRQ	1 — ready
1	INTRQ	1 — done
2	HLD	1 — head loaded
3	Side Select	
4	DS3	
5	DS2	
6	DS1	
7	Density	

TABLE II

encoding methods respectively. It also provides encoding and decoding of these recording formats plus generation of prewrite compensation information. All commands and data are transferred over an 8-bit bi-directional bus to one of four registers described by address lines A0-A1.

1.2.1 DISK REGISTERS

The four disk registers and their addresses are:

Command/Status F020
Track F021
Sector F022
Data F023

The command register is used to issue one of eleven commands to the controller (see Table III). These commands are divided into four groups or types. Type I commands are head positioning commands. Type II commands perform normal read and write functions. Type III commands are used for formatting. Type IV commands are used to check disk status.

1.2.2 COMMAND DESCRIPTION

Command words should only be loaded in the Command Register when the Busy status bit is off (Status bit 0). The one exception is the Force Interrupt command. Whenever a command is being executed, the Busy status bit is set. When a command is completed, an interrupt is generated and the Busy status bit is reset. The Status Register indicates whether the completed command encountered an error or was fault free. For ease of discussion, commands are divided into four types. Commands and types are summarized in Table 4.

1.2.3 TYPE I COMMANDS

The Type I Commands include the Restrore, Seek, Step, Step-in, and Step-Out commands. Each of the Type I Commands contains a rate field (r_0r_1), which determines the stepping motor rate as defined in Table 3.

Table 3. STEPPING RATES

CLK		2 MHz	2 MHz	1 MHz	1 MHz	2 MHz	1 MHz
DDEN		0	1	0	1		
R1	R0	TEST-1	TEST-1	TEST-1	TEST-1	TEST-0	TEST-0
0	0	3ms	3ms	6ms	6ms	Approx.	Approx.
0	1	6ms	6ms	12ms	12ms	200 µs	400 µs
1	0	10ms	10ms	20ms	20ms		
1	1	15ms	15ms	30ms	30ms		

The Type 1 Commands contain head load flag (h) which determines if the head is to be loaded at the beginning of the command. If h is 1, the head is loaded at the beginning of the command (HLD output is made active). If h is 0, HLD is deactivated. Once the head is loaded, the head will remain engaged until the FD1791 receives a command that specifically disengages the head. If the FD1791 is idle (busy is 0) for 15 revolutions of the disk, the head will be automatically disengaged (HLD made inactive).

The Type I Commands also contain a verification (V) flag which determines if a verification operation is to take place on the destination track. If V is 1, a verification is performed, if V is 0, no verification is performed.

During verification, the head is loaded and after an internal 15 ms delay, the HLT input is sampled. When HLT is active (logic true), the first encountered ID field is read off the disk. The track address of the ID field is then compared to the Track Register. If there is a match and a valid ID CRC, the verification is complete, an interrupt is generated and the Busy status bit is reset. If there is not a match but there is valid ID CRC, an interrupt is generated, and Seek Error Status bit (Status bit 4) is set and the Busy status bit is reset. If there is a match but not a valid CRC, the CRC error status bit is set (Status bit 3), and the next encountered ID field is read from the disk for the verification operation. If an ID field with a valid CRC cannot be found after four revolutions of the disk, the FD1791 terminates the operation and sends an interrupt, (INTRQ).

The Step, Step-In, and Step-Out commands contain an Update flag (U). When U is 1, the track register is updated by one for each step. When U is 0, the track register is not updated.

1.2.3.1 RESTORE (SEEK TRACK 0)

Upon receipt of this command the Track 00 (TROO) input is sampled. If TROO is active low indicating the Read-Write head is positioned over track 0, the Track Register is loaded with zeroes and an interrupt is generated. If TROO is not active low, stepping pulses at a rate specified by the r_1r_0 field are issued until the TROO input is activated. At this time the TR is loaded with zeroes and an interrupt is generated. If the TROO input does not go active low after 255 stepping pulses, the FD1791 terminates operation, interrupts, and sets the Seek error status bit. Note that the Restore command is executed when MR goes from an active to an inactive state. A verification operation takes place if the V flag is set. The h bit allows the head to be loaded at the start of command.

SEEK

This command assumes that the Track Register contains the track number of the current position of the Read-Write head and the Data Register contains the desired track number. The FD1791 will update the Track register and issue stepping pulses in the appropriate direction until the contents of the Track register are equal to the contents of the data register (the desired track location). A verification operation takes place if the V flag is on. The h bit allows the head to be loaded at the start of the com-

mand. An interrupt is generated at the completion of the command.

1.2.3.3 STEP

Upon receipt of this command, the FD1791 issues one stepping pulse to the disk drive. The stepping motor direction is the same as in the previous step command. After a delay determined by the r_1r_0 field, a verification takes place if the V flag is on. If the u flag is on, the TR is updated. The h bit allows the head to be loaded at the start of the command. An interrupt is generated at the completion of the command.

1.2.3.4 STEP-IN

Upon receipt of this command, the FD1791 issues one stepping pulse in the direction towards track 76. If the u flag is on, the Track Register is incremented by one. After a delay determined by the r_1r_0 field, a verification takes place if the V flag is on. The h bit allows the head to be loaded at the start of the command. An interrupt is generated at the completion of the command.

1.2.3.5 STEP-OUT

Upon receipt of this command, the FD1791 issues one stepping pulse in the direction towards track 0. If the u flag is on, the TR is decremented by one. After a delay determined by the r_1r_0 field, a verification takes place if the V flag is on. The h bit allows the start of the command. An interrupt is generated at the completion of the command.

1.2.4 TYPE II COMMANDS

The type II Commands include the Read Sector(s) and Write Sector(s) commands. Prior to loading the Type II Command into the Command Register, the computer must load the Sector Register with the desired sector number. Upon receipt of the Type II command, the busy status Bit is set. If the E flag is 1 (this is the normal case) HLD is made active and HLT is sampled after a 15 msec delay. If the E flag is 0, the head is loaded and HLT sampled with no 15 msec dalsy. The ID field and Data Field format are shown below.

When an ID field is located on the disk, the FD1791 compares the Track number of the ID field with the Track Register. If there is not a match, the next encountered ID field is read and a comparison is again made. If there was a match, the Sector Number of the ID field is compared with the Sector Register. If there is not a Sector match, the next encountered ID field is read off the disk and comparisons again made. If the ID field CRC is correct, the data field is then located and will be either written into, or read from depending upon the command. The FD1791 must find an ID field with a Track number, Sector number, and CRC within four revolutions of the disk; otherwise, the Record not found status bit is set (Status bit 3) and the command is terminated with an interrupt.

Table 4. COMMANDS

		BITS							
TYPE	COMMAND	7	6	5	4	3	2	1	0
I	Restore	0	0	0	0	h	V	r_1	r_0
I	Seek	0	0	0	1	h	V	r_1	r_0
I	Step	0	0	1	u	h	V	r_1	r_0
I	Step In	0	1	0	u	h	V	r_1	r_0
I	Step Out	0	1	1	u	h	V	r_1	r_0
II	Read Command	1	0	0	m	X	E	0	0
II	Write Command	1	0	1	m	X	E	X	a_0
II	Read Address	1	1	0	0	0	1	0	0
III	Read Track	1	1	1	0	0	1	0	X
III	Write Track	1	1	1	1	0	1	0	0
IV	Force Interrrupt	1	1	0	1	I_3	I_2	I_1	I_0

X = Don't care
Note: Bits shown in TRUE form.

Table 5. FLAG SUMMARY

TYPE I

h = Head Load Flag (Bit 3)
- h = 1, Load head at beginning
- h = 0, Unload head at beginning

V = Verify flag (Bit 2)
- V = 1, Verify on last track
- V = 0, No verify

r_1r_0 = Stepping motor rate (Bits 1-0)

Refer to Table 1 for rate summary

u = Update flag (Bit 4)
- u = 1, Update Track register
- u = 0, No update

Table 6. FLAG SUMMARY

TYPE II

m = Multiple Record flag (Bit 4)
- m = 0, Single Record
- m = 1, Multiple Records

a_0 = Data Address Mark (Bit 0)
- a_0 = 0, FB (Data Mark)
- a_0 = 1, F8 (Deleted Data Mark)

E = 15 ms Delay
- E = 1, 15 ms delay
- E = 0, no 15 ms delay

Sector Length Field (hex)	Number of Bytes in Sector (decimal)
00	128
01	256
02	512
03	1024

Each of the Type II Commands contains an (m) flag which determines if multiple records (sectors) are to be read or written, depending upon the command. If m is 0, a single sector is read or written and an interrupt is generated at the completion of the command. If m is 1, multiple records are read or written with the sector register internally updated so that an address verification can occur on the next record. The FD1791 will continue to read or write multiple records and update the sector register until the sector register exceeds the number of sectors on the track or until the Force Interrupt command is loaded into the Command Register, which terminates the command and generates an interrupt.

1.2.4.1 READ COMMAND

Upon receipt of the Read command, the head is loaded, the Busy status bit set, and when an ID field is encountered that has the correct track number, correct sector number, and correct CRC, the data field is presented to the computer. The Data Address Mark of the data field must be found within 30 bytes in single density and 43 bytes in double density of the last ID field CRC byte; if not, the Record Not Found status bit is set and the operation is terminated.

When the first character or byte of the data field has been shifted through the DATA SHIFT REGISTER, it is transferred to the DATA REGISTER, and DRQ is generated. When the next byte is accumulated in the DSR, it is transferred to the DR and another DRQ is generated. If the Computer has not read the previous contents of the DR before a new character is transferred that character is lost and the Lost Data Status bit is set. This sequence continues until the complete data field has been inputted to the computer. If there is a CRC error at the end of the data field, the CRC error status bit is set, and the command is terminated (even if it is a multiple record command).

At the end of the Read operation, the type of Data Address Mark encountered in the data field is recorded in the Status Register (Bit 5) as shown below:

STATUS BIT 5	
1	Deleted Data Mark
0	Data Mark

1.2.4.2 WRITE COMMAND

Upon receipt of the Write command, the head is loaded (HLD active) and the Busy status bit is set. When an ID field is encountered that has the correct track number, correct sector number, and correct CRC, a DRQ is generated. The FD1791 counts off 11 bytes in single density and 22 bytes in double density from the CRC field and the Write Gate (WG) output is made active if the DRQ is services (i.e., the DR has been loaded by the computer). If DRQ has not been serviced, the command is terminated and the Lost Data status bit is set. If the DRQ has been serviced, the WG is made active and six bytes of zeros in single density and 12 bytes in double density are then written on the disk. At this time the Data Address Mark is then written on the disk as determined by the a_0 field of the command as shown below:

a_0	Data Address Mark (Bit 0)
1	Deleted Data Mark
0	Data Mark

The FD1791 then writes the data field and generates DRQ's to the computer. If the DRQ is not serviced in time for continuous writing the Lost Data Status Bit is set and a byte of zeros is written on the disk. The command is not terminated. After the last data byte has been written on the disk, the two-byte CRC is computed internally and written on the disk followed by one byte of logic ones in FM or 4E in MFM. The WG output is then deactivated.

1.2.5 TYPE III COMMANDS
READ ADDRESS

Upon receipt of the Read Address command, the head is loaded and the Busy Status is set. The next encountered ID field is then read in from the disk, and the six data bytes of the ID field are assembled and transferred to the DR, and a DRQ is generated for each byte. The six bytes of the ID field are shown below:

TRACK ADDR	SIDE NUMBER	SECTOR ADDRESS	SECTOR LENGTH	CRC 2	CRC 2
1	2	3	4	5	6

Although the CRC characters are transferred to the computer, the FD1791 checks for validity and the CRC error status bit is set if there is a CRC error. The Track Address of the ID field is written into the sector register. At the end of the operation an interrupt is generated and the Busy Status is reset.

Table 7. STATUS REGISTER SUMMARY

BIT	ALL TYPE I COMMANDS	READ ADDRESS	READ	READ TRACK	WRITE	WRITE TRACK
S7	NOT READY	NOT READY	NOT READY	NOT READY	NOT REACH	NOT READY
S6	WRITE PROTECT	0	0	0	WRITE PROTECT	WRITE PROTECT
S5	HEAD	0	RECORD TYPE FOUND	0	WRITE FAULT	WRITE FAULT
S3	CRC ERROR	CRC ERROR	CRC ERROR	0	CRC ERRIR	0
S2	TRACK 0	LOST DATA	LOST DATA	LOST DATA	LOST DATA	LOST DATA
S1	INDEX	DRQ	DRQ	DRQ	DRQ	DRQ
S0	BUSY	BUSY	BUSY	BUSY	BUSY	BUSY

1.2.5.1 READ TRACK

Upon receipt of the Read Track command, the head is loaded and the Busy Status bit is set. Reading starts with the leading edge of the first encountered index mark and continues until the next index pulse. As each byte is assembled it is transferred to the Data Register and the Data Request is generated for each byte. No CRC checking is performed. Gaps are included in the input data stream. The accumulation of bytes is synchronized to each Address Mark encountered. Upon completion of the command, the interrupt is activated.

1.2.5.2 WRITE TRACK

Upon receipt of the Write Track command, the head is loaded and the Busy Status bit is set. Writing starts with the leading edge of the first encountered index pulse and continues until the next index pulse, at which time the interrupt is activated. The Data Request is activated immediately upon receiving the command, but writing will not start until after the first byte has been loaded into the Data Register. If the DR has not been loaded by the time the index pulse is encountered the operation is terminated making the device Not Busy, the Lost Data Status Bit is set, and the Interrupt is activated. If a byte is not present in the DR when needed, a byte of zeros is substituted. Address Marks and CRC characters are written on the disk by detecting certain data byte patterns in the outgoing data stream as shown in the table below. The CRC generator is initialized when any data byte from F8 to FE is about to be transferred from the DR to the DSR in FM or by receipt of F5 in MFM.

1.2.6 TYPE IV COMMAND FORCE INTERRUPT

This command can be loaded into the command register at any time. If there is a current command under execution (Busy Status Bit set), the command will be terminated and an interrupt will be generated when the condition specified in the I_0 throuth I_3 field is detected. The interrupt conditions are shown below:

I_0 — Not-Ready-To-Ready Transition
I_1 — Ready-To-Not-Ready Transition
I_2 — Every Index Pulse
I_3 — Immediate Interrupt

NOTE: If I_0 - I_3 are 0, there is no interrupt generated but the current command is terminated and busy is reset. This is the only command that will clear the immediate interrupt.

STATUS DESCRIPTION

Upon receipt of any command, except the Force Interrupt command, the Busy Status bit is set and the rest of the status bits are updated or cleared for the new command. If the Force Interrupt Command is received when there is a current command under execution, the Busy status bit is reset, and the rest of the status bits are unchanged. If the Force Interrupt command is received when there is not a current command under execution, the Busy Status bit is reset and the rest of the status bits are updated or cleared. In this case, Status reflects the Type I commands.

Status varies according to the type of command executed as shown in Table 7.

1.2.7 FORMATTING THE DISK

Formatting the disk is accomplished by positioning the R/W head over the desired track number and issuing the Write Track command. Upon receipt of the Write Track command, the FD1791 raises the data request signal. At this point in time, the user loads the data register with desired data to be written on the disk. For every byte of information to be written on the disk, a data request is generated. This sequence continues from one index mark to the next index mark. Normally, whatever data pattern appears in the data register is written on the disk with a normal clock pattern. However, if the FD1791 detects a data pattern on F5 thru FE in the data register, this is interpreted as data address marks with missing clocks or CRC generation. For instance, in FM an FE pattern will

be interpreted as an ID address mark (DATA-FE, CLK-C7) and the CRC will be initialized. An F7 pattern will generate two CRC characters in FM or MFM. As a consequence, the patterns F5 thru FE must not appear in the gaps, data fields, or ID fields. Also, CRC's must be generated by an F7 pattern.

1.2.7.1 IBM 3740 FORMAT— 128 BYTES/SECTOR

Shown below is the IBM single-density format with 128 bytes/sector. In order to format a diskette, the user must issue the Write Track command, and load the data register with the following values. For every byte to be written, there is one data request.

Disks may be formatted in IBM 3740 or System 34 formats with sector lengths of 128, 256, 512, or 1024 bytes.

NUMBER OF BYTES	HEX VALUE OF BYTE WRITTEN
55	FF
6	00
1	FC (Index Mark)
26	FF
6	00
1	FE (ID Adress Mark)
1	Track Number
1	00
1	Sector Number (1 thru 1A)
1	00
1	F7 (2 CRC's written)
11	FF
6	00
1	FB (Data Address Mark)
128	Data (IBM uses E5)
1	F7 (2 CRC's written)
27	FF
247**	FF

*Write bracketed field 26 times
**Continue writing until FD1791 interrupts out. Approx. 247 bytes.

1.2.7.2 IBM SYSTEM 34 FORMAT - 256 BYTES/SECTOR

Shown below is the IBM dual-density format with 256 bytes/sector. In order to format a diskette, the user must issue the Write Track command and load the data register with the following values. For every byte to be witten, there is one data request.

NON-IBM FORMATS

Variations in the IBM format are possible to a limited extent if the following requirements are met: sector size must be a choice of 128, 256, 512 or 1024 bytes: gap size must be according to the following table. Note that the Index Mark is not required by the FD1791.

*Byte counts must be exact.
**Byte counts are minimum, except exactly 3 bytes of A1 must be written.

NUMBER OF BYTES	HEX VALUE OF BYTE WRITTEN
110	4E
12	00
3	F6
1	FC
50	4E
12	00
1	F5
1	FE
1	Track Number (0 thru 4C)
1	Side Number (0 or 1)
1	Sector Number (0 thru 1A)
1	01
1	F7
22	4E
12	00
3	F5
1	FB
256	40
1	F7
54	4E
598**	

Write bracketed field 26 times.
**Continue writing until FD1791 interrupts out. Approx. 598 bytes.

1.3 DISK INTERFACE

All control lines to the disk are driven by Tri-State Bus Drivers. (74LS368, 8T98 etc). Status lines from the disk are terminated by 150Ω resistors before being fed into 74LS14 Schmidt triggers. Only the following disk interface lines are connected:

Name	8″	5¼″
TG43/WCS	2	NC
Side	14	32
HDLD/Motor on	18	16
Index	20	8
Ready	22	NC
DS1	26	10
DS2	28	12
DS3	30	14
DS4	NC	NA
DIR	34	18
STEP	36	20
WD	38	22
WG	40	24
TR00	42	26
WP	44	28
RD	46	30

1.3.1 DRIVE SELECT AND HEAD LOAD

As shipped from the factory drive selects 1-3 are connected to the drive interface connector. However, an 8″ disk generally provides for 4 discrete drive select lines. If it is necessary to use DS4 it is recommended that the side select driver be jumpered to pin 32 of the 50 pin ribbon connec-

tor. This will permit the use of 4 single-sided drives without modification to the drives. If it is necessary to have more than 3 double-sided drives or more than 4 single-sided drives, the drive select lines can be encoded to provide up to eight drives. Refer to the drive manufacturers instructions to implement this option.

The Conductor expects the drive to load and unload its heads as a result of the condition of the Head load signal. Most drives, however, activate Head load as a result of selecting a drive. Most drives also permit a simple modification to permit the use of Head load. The ramifications are as follows:

If drive select loads the heads, then in order to unload the heads after a period of inactivity some sort of timer must be implemented. Alternatively, software can monitor the condition of the HDLD Bit in the status port and deselect the drives when inactive. The HDLD bit is automatically deactivated after 15 revolutions of the disk without activity.

If the drives are modified to monitor the HDLD signal then deselecting drive is unnecessary. However, it is still necessary to introduce a delay after issueing a command if HDLD was inactive in order to allow for Head load settle time. In extremely cost sensitive situations HDLD can be ignored, leaving one drive always selected with head(s) loaded. Also, because of the recovery technique incorporated into the phase locked loop, it is possible to ignore Head Load Settle time, although this practice is not recommended.

1.3.2 PRE-WRITE COMPENSATION (PRECOMP)

One of the many problems associated with magnetic recording is the phenomenon of "Bit Spread" or "Bit Shift". Certain combinations of "1's" and "0's" when read back from the disk will appear to have bits shifted from their nominal centers by as much as 400 ns. Since the VFO produces a Data Window which is 50% of read clock (1000 ns in double density for 8" disk), and the worst case bit shift is 800 ns, this leaves us with a 100 ns margin on each side of the window. This margin is unacceptable as the minimum margin should be 200 ns. Single density recording provides a 2000 ns window and therefore avoids this problem for 8" disk. Also, times are double for 5¼" disks. With the conductor the solution used is to move the bit an amount equal to it's known shift from nominal, but in the opposite direction, before writing. (i.e. Prewrite Compensation).

The Conductor accomplishes Pre-Comp by using an LS123 one-shot and a circuit which is capable of switching the timing components of this one-shot. The signals used to select the time are the early and late signals from the WD1791. These signals indicate which direction the current bit should be shifted. The DDEN line is included in the formula to prevent pre-comp in single density.

Refer to the Shugart Associates Double Density Design Guide for further explanation of Pre-Compensation.

1.4 PHASE - LOCKED LOOP

This discussion assumes an 8" disk. For 5¼ inch disks all times are doubled.

Data is encoded on the disk in either FM (single density) or MFM (double density) format. In the FM format, a clock is recorded every 4μs regardless of the bit that follows or preceeds it. Then 2μs later the data bit is written. If the data bit is one, a flux reversal takes place, and if it's 0, no reversal takes place. This is the easiest method to decode because the clocks are always present regardless of the data being read. Also, the bit cell is 4μs long as opposed to 2μs for MFM. In MFM encoding, the clocks are only present between consecutive 0 bits. This decreases the bit cell to 2μs and allows twice as many bits to be written in the same space — hence, double density.

In either case, the controller must provide some means of separating the clocks from the data and presenting this information to the WD1791.

One method for data separation involves using a series of one-shot and flip flops that provide a pulse for clock and a pulse for data. This method does not provide the Read Clock required for the WD1791, nor does it allow MFM Data Separation. Another method which solves both of these deficiencies involves using a shift register, a crystal and a prom. Data is shifted into the shift register and compared against the prom, which develops the window and the data gate. The problem with this method is that, while eliminating the need for Pre-Write Compensation, it does not provide for media interchange, speed variations or read-amp jitter.

The best method currently available for data separation involves the use of a phase-locked loop which is designed to acquire phase with the clocks and data coming from the disk. This allows a speed variation tolerance of greater than 20% in 8" FM decoding - well above the expected variation.

The PLL contains a variable frequency oscillator (VFO) in a special application which in effect causes the VFO to change the phase of its clock and not the effective frequency. The output of this VFO then is an 8 MHZ (for 8" disks) square wave which is locked to Raw Data from the disk. This 8 MHZ is used to clock the shift registers, etc. used to obtain the data and data window which is fed to the WD1791.

For a 5¼" disk, the frequency is 4 MHZ. Also, filter and timing components must be changed to reflect the slower data rate. This is done by changing a pre-stuffed header plug (supplied) containing the proper components. Also, the WRITE oscillator and the incidental WD1791 CLK signal must be divided in half. This is accomplished by moving the oscillator shorting plug to the post marked 5".

2. SOFTWARE DISCRIPTION

This description assumes that the programmer has read the sections on the WD1791 (sec 1.2) and on drive/side selection (sec 1.3.1) Refer to tables I & II for summaries of register usage. The examples given are excerpted from the DATASPEED CP/M CBIOS.

2.1 SELECTING A DRIVE OR SIDE

There are two methods of drive selection available with the conductor — Discreet selection and Encoded selection. Encoded selection allows up to 8 drives while discreet allows only 3 (4 if single sided). Basically, the drive select lines are manipulated by outputting select information on bits 4, 5 and 6 of Port F0 (hex). For discreet selection, Drive 1 will be selected when Bit 6 is low, Drive 2 is selected when Bit 5 is low and Drive 3 is selected when Bit 4 is low. If single sided drives are being used and the recommended hardware modification is installed on the board (sec 1.3.1), then Bit 3 may be set low to select Drive 4. With this method, it is possible to select more than one drive at a time. This is not desirable and should be avoided.

For Encoded selection, Bits 4, 5 and 6 become the compliment of the drive number with 111 being drive 0, 011 being drive 1, 101 being drive 2 etc. This method requires drives which are set up for decoding drive select lines. Refer to the manual for your drive to determine whether this is available or not.

When using double-sided drives, Bit 3 of Port F0 is used to select side A or B. When set to "0", side A is selected while setting to "1" selects side B.

All Bits in port F0 except 0, 1 and 2 are mirrored back to input port F0, allowing the currently selected drive, side or density to be read, changed and written back selectively. For example, to change sides of the disk regardless of the current side the following 8080 code may be used:

```
SIDECHNG    IN 0F0H     ; Read Status
                          Port
            XRI 8       ; Toggle Side
                          Select
            OUT 0F0H    ; Write to Con-
                          ductor
```

2.2 SELECTING DENSITY

The Conductor is capable of reading and writing information in either single or double density. To select single density mode, Bit 7 of Port F0 must be set to one. Double density is selected by setting this Bit to "0". A disk must be read and written using the same mode that it was formatted in. See section 2.5 for more information on formats. The density bit can be read back to determine the current state of the controller.

2.3 READING AND WRITING

There are two methods for reading and writing - the polled method and the interrupt method. Which method is used is dictated by speed considerations and the desirability of using interrupts.

2.3.1 THE POLLED METHOD

If the environment makes it undesirable to use the interrupt system and timing is not critical (i.e. system is not using 8" double density), the software may poll for status during a read or write. Basically, the sequence is as follows. After the drive, side and density are selected and the head is positioned over the proper track (see SEEK section 2.4), pointers are set up to point to the Buffer where the data is to be transferred to or from the Conductor data register; and the status register. The command register is then loaded with the proper command. After this, the HLT Bit is reset to allow data flow. Note that even during a write command, the WD1791 must still read sector headers to locate the proper sector, therefore HLT must be set low. The program may now enter the transfer loop. This loop consists of inputting port 0F0H and checking Bit 0 (DRQ). If DRQ is low, the program waits in a loop until it goes high. This is known as "Polling Ready." When DRQ goes high, data is transferred between the buffer and the data register, with the direction being determined by the type of command. After a byte is transferred the buffer pointer is incremented and the status port is checked for Bit 1 (INTRQ). If this Bit is a one, the operation is complete. Otherwise, the program must loop back for another byte. See listing 1.

This is the simplest form, and also the most time consuming. The programmer must take into account the processor speed when determining whether this method will be fast enough. The speeds required are as follows:

Disk-Type	Latency
8" double-density	13.5μs.
8" single-density	27μs.
5¼" double-density	27μs.
5¼" single-density	54μs.

2.3.2 THE WAIT/INTERRUPT METHOD

Some of the time required by the Polled method may be spared by eliminating the need to poll for DRQ. This is possible by activating the wait logic (set Bit 0 of Port F0 low). When this is done the computer will wait for DRQ automatically whenever a read or write to the data port is attempted. This wait logic must only be enabled during a read or write command or the computer may enter a permanent wait state requiring a reset or power-down. The Poll for INTRQ may also be eliminated by enabling the interrupt logic. (set Bit 1 of Port F0 to 1). This allows the computer to interrupt out (of the transfer loop) as a result of interrupt request (INTRQ). Now the read loop is reduced to read, store, increment and

jump. In 8080 code if the H&L registers are loaded with the start of the loop, a PCHL instruction can replace the jump, providing the tightest possible loop. This is the only method that can handle 8" Double Density in a 2 MHZ system.

The interrupt response vector will be FF (HEX). In an 8080 system, this means that a RST 7 will be performed. If this location is being used by other software, the read/write routine must save locations 38, 39 and 3A (HEX) and replace them with a jump to the disk done routine. If this vector is being used by another interrupting device, that device must be disabled from interrupting before the disk drivers are called, and reenabled at exit. The interrupt output may be jumpered to one of the Priority Interrupt lines for special applications.

When the done routine is entered, HLT, the wait logic and the interrupt logic should all be disabled, followed by a status read to determine if any errors occurred.

2.4 HEAD POSITIONING (SEEK)

Before a disk transfer can take place the drive, side and density must be selected and the head(s) must be placed over the proper track. Head positioning is accomplished by storing the desired track number in the data register (wait logic must be disabled) and storing the seek command in the command register. The controller will position the head over the requested track and signal the program when done by raising INTRQ. This procedure assumes that the controller knows where the head is currently positioned (i.e. the track register contains the current track). In order to insure this condition, the restore command must be issued as part of the power-on/reset Initialization routine.

2.5 FORMATTING

The controller uses a soft sectored format similar to most hard disks. This format is variable to a certain extent, but it is recommended that the IBM formats be used. Refer to section 1.2. for more information on formatting. Basically, a memory buffer is filled with the necessary data and written to disk using the Track Write command. This command starts writing from the beginning of the Index pulse until the beginning of the next index. If a disk is being formatted for the CP/M™ operating system, 128 byte sectors must be used in both single and double density. This is a deviation from the IBM format.

DataSpeed

INSTALLATION INSTRUCTIONS

The CONDUCTOR™ is shipped fully assembled and ready to plug into any S-100 bus system. It is designed to interface with most drives as they are shipped from the factory in their standard configurations. However, it is advisable to configure the "A" drive to enable stepper power from "Drive Select". Most drives will be configured to enable stepper power from "Head Load". Refer to the manual on your particular drive to implement this option. (For SHUGART SA800's, move the shorting plug from HL to DS). If you ordered a complete drive system from DATASPEED, this has already been done.

Before installing the CONDUCTOR, it is advisable to remove all cards from the computer which are not necessary for the initial bring-up. You will require a CPU, 16K of RAM, some type of I/O and a front panel or monitor ROM.. Insure that the ROM cards address does not interfere with the CONDUCTOR's address (F000-F0FF). Also, since the CONDUCTOR is interrupt driven, disable any other interrupts before proceeding. This can be done by either cutting the traces on cards which drive the interrupt line or lifting the pin (if the board is socketed) which drives that line. These precautions are recommended in order to prevent interrupts which the software is not ready to service. After the initial bring-up, software can be written to account for these interrupts and they may be re-enabled.

If you are using a Z-80 card, insure that it has an 8080 I/O mode (mirrored address) capability and that it is enabled. The CONDUCTOR will not operate in Z-80 I/O mode without special software. If you require Z-80 I/O contact the factory for assistance.

You are now ready to install the conductor and perform a "bootstrap". WITH SYSTEM AND DRIVE POWER OFF connect a cable to the appropriate connector on the CONDUCTOR and the drive. Make sure that polarity of the cable is correct. Pin 1 on the CONDUCTOR is to the left when facing the component side of the board. Place the CONDUCTOR into a bus slot as close to the CPU as possible. Open the doors to all the drives and apply power to the drives, then apply system power. At this time, drive "A" should restore (move its heads to track 0). If itsdoes not, turn off power and go back and check your work. Insure that the cables are properly installed, then manually move the heads on drive "A" toward the spindle. Try the power up sequence again. You may find it necessary to close the door on drive "A" (WITHOUT A DISKETTE). If you cannot cause the drive to restore, call the factory for assistance.

Now you are ready to place your software diskette in drive "A" and close the door. After doing this, start your system at location F000(hex). If you are using a monitor ROM, a "GO" command should be used. For front panel operation examine F000 and press run. You may also set your power on Jump feature (if available) to F000.

The controller will force the first sector of the disk (track0, sector 1) into memory starting at location 0. At the end of the load, an interrupt to location 38(hex) will occur. It is up to the user to insure that the proper "bootstrap" code has been placed in the first sector. Please note that the first track is always single density.

If you ordered CP/M from DATASPEED, the proper code to complete the load is already on the disk. You should hear the disk perform 2 seek operations and stop. Refer to "Bringing up CP/M" for further information. If the disk will not perform the seeks, check your system, then call the factory for assistance.

DATASPEED INC 10-21-79

SUBJECT: CONDUCTOR DISK CONTROLLER EC TO REV A

PURPOSE: PHANTOM MODIFICATION. INSTALL TO ALLOW CONDUCTOR TO PHANTOM OUT MEMORY AND CO-RESIDE WITH 64K OF RAM

1. INSTALL WIRE FROM BUS PIN 67 TO PIN 2 OF IC-22.

DATASPEED INC 10-21-79

SUBJECT: CONDUCTOR DISK CONTROLLER EC TO REV B

PURPOSE: INDEX PULSE SHAPING - ALLOWS DATA STORAGE IN THE INDEX AREA.

1. PLACE A 20K OHM 1/4 WATT RESISTOR BETWEEN PINS 7 AND 16 OF IC-3.

2. PLACE A .01 uF CAPACITOR BETWEEN PINS 6 & 7 OF IC-3.

3. CHANGE R17 FROM 20K TO 33K OHMS.

4. CUT TRACE BETWEEN PINS 9, 10, AND 11 OF IC-3.

5. PLACE A WIRE BETWEEN PIN 9 OF IC-3 AND PIN 8 OF IC-3.

6. PLACE A WIRE BETWEEN PIN 10 OF IC-3 AND PIN 2 OF IC-41.

7. CUT TRACE LEADING FROM PIN 12 OF IC-41.

8. PLACE A WIRE BETWEEN PIN 12 OF IC-3 AND PIN 35 OF IC-30 (WD1791).

9. TACK LONG WIRES TO BOARD WITH TAPE OR SILICON RUBBER.

NOTES: ALL MODIFICATIONS ARE ON SOLDER SIDE OF BOARD. INSTALLING THIS OPTION WILL ALLOW THE CONDUCTOR TO READ DATA FROM BOARDS WHICH HAVE WRITTEN DATA IN THE INDEX AREA. IT ALSO ALLOWS THE USER TO MODIFY HIS FORMAT PROGRAM TO PLACE UP TO 52 SECTORS ON EACH TRACK. PLEASE NOTE THAT THIS IS NOT RECOMMENDED AND WILL REQUIRE CHANGES TO CPM AND OTHER SOFTWARE.

DATASPEED INC 10-21-79

SUBJECT: CONDUCTOR DISK CONTROLLER EC TO REV C

PURPOSE: PRE-COMP DISABLE OPTION FOR 5 1/4" DISKETTES.

INSTALL IF MINIDISKS ARE USED IN DOUBLE DENSITY.

1. CUT TRACE LEADING FROM PIN 2 OF IC-39.

2. PLACE WIRE BETWEEN PINS 2 AND 7 OF IC-39.

DataSpeed

BRINGING UP CP/M^{tm}

If you ordered the optional CP/M operating system, you may have to alter the console input and output drivers to conform to your system. The system is shipped with drivers included for a "standard" serial I/O board such as the Cromemco Tuart or the PTC 3P+S with port addressing as follows:

STATUS	00H	
DATA	01H	
XMIT READY	80H	
RCVR READY	40H	(logic 1 = TRUE)

No initialization code for serial I/O is provided in the distribution version. You must patch such code if it is necessary.

If your computer uses this I/O scheme, simply install the CONDUCTOR and perform a bootstrap as described in the installation instructions. The computer should read the operating system into memory and hang in a loop in the CONOUT routine (see CBIOS listing). Change the byte at location 3FF5(hex) from "C3" to "CA" hex and restart the computer at location 0. The system should reboot and display the CP/M prompt on your console. Refer to the CP/M alteration guide and other manuals for further information.

If your computer cannot use these I/O drivers, you must patch the BIOS portion of CP/M. Refer to the CBIOS.PRN and the system alteration guide for instructions.

If your serial I/O or other hardware requires initialization, it is recommended that you patch it into the boot program. Initially, however, such code must be "toggled" in using a front panel or executed in some other way.

DATASPEED INC.

DUAL DENSITY CP/M VERSION 1.4.5

This version of CP/M has been customized by DATASPEED INC. to allow semi-transparent density switching. The user may use single and double density diskettes on the same system by using the "SET" command. This command has two forms:

SET(cr)

and

SET X:Y (where X = A,B,C OR D and Y = S or D)

The first form displays the current density selected for each drive. The second form allows the user to set the density of any drive. For example, if a single density diskette were placed on the "B" drive, it could be read or written by typing:

SET B:S(cr)

The density selected will remain in effect until changed by the user or by a COLD start (warm starts do not effect density except on drive "A" which will be set to double density automatically).

User programs may also select the density of any drive. The density for each drive is stored in locations 40-43H for drives "A" - "D" respectively.

If bit 7 (sign bit) is set, it indicates that the respective drive is single density. All further reads and writes will be performed in single density. (NOTE: if the diskette in that drive is not formatted for the selected density, a "4" will be printed on the console followed by "BDOS ERROR ON x: BAD SECTOR" where x is the drive name.) If bit 7 is reset, that drive will be double density. It may be necessary to follow the density select with a drive select operation in some programs, even if the drive is already selected.

The following programs show methods of selecting density in 8080 assembly language and in BASIC.

```
          ;
          ; DENSITY SELECT
          ;
          ; DRIVE # IS IN "C" REGISTER
          ; DENSITY IS IN "B" REGISTER (0=Double, 80H=Single)
          ;
          ;
          ;
DENSEL:   LXI     D,40H     ; POINT TO DENSITY BYTES
          MOV     L,C       ; AND FORM POINTER TO BYTE FOR DRIVE
          MVI     H,0       ; GIVEN BY CREG
          DAD     D         ; H&L POINTS TO DENSITY BYTE
          MOV     A,M       ; PICK UP THE BYTE
          ANI     7FH       ; STRIP OFF OLD DENSITY
          ORA     B         ; "OR" IN NEW DENSITY
          MOV     M,A       ; AND PUT BACK IN TABLE
          RET               ; RETURN TO CALLER
          ; NOTE:  ONLY BIT 7 CAN BE MODIFIED.  OTHER BITS
          ; CONTAIN THE CURRENT TRACK NUMBER FOR THIS DRIVE.
          ; END (DENSEL)
```

BASIC CODE.

```
0010    REMARK  *  IF Z1 .LT. 128 THEN DENSITY=DOUBLE.
0020    REMARK  *  IF Z1 .GE. 128 THEN DENSITY=SINGLE.
0030    REMARK  *  Z2 CONTAINS DRIVE NUMBER (0-3).
0040    POKE (64+Z2), (PEEK(64+Z2) AND 127) OR (Z1 AND 128)
0050    RETURN
```

After changing the density, the system may have to be reinitialized (if the disk had previously been opened.) Otherwise it will be set to read only (R/O) by CP/M.

DataSpeed

DISKTEST VERSION 1.4.5

This program is an exerciser for the entire disk system. It includes tests for the controller, the media and the drive. It is designed to run under DATASPEEDS Version 1.4.5 of CP/M, but will run under earlier versions in double density only. The prompts are self-documenting.

Once running, the program will fill the disk with zeros, providing a check of the format, then perform a seek test. Following this, several patterns will be written to the entire disk. This portion of the test checks for bit-shift tolerence and general pattern sensitivity. Finally, another seek test is performed and the loop is terminated.

There are 2 types of error indications. One is a descriptive error message and the other is the error code from the CBIOS. Since the narrative type is self-explanatory, only the CBIOS codes are described.

CODE	MEANING
Ø	On a read this indicates that a deleted data mark was found. On a write, indicates that the disk is write protected.
1	LOST DATA - check CPU speed. CPU must run at 2MHZ minimum including any wait states.
2	CRC ERROR - this can be caused by bad diskette, dust, electrical noise, or failing data separator.
4	RECORD NOT FOUND - indicates that the ID field requested could not be located. Causes - wrong disk format or density, damaged diskette, failing data spparator or positioning error (SEEK ERROR)
6	this is a combination of errors 2 and 4. Indicates a CRC ERROR in the ID field. See error description for 2.

This version is assembled for 2MHZ operation. If 4MHZ is desired, reassemble with "TWOMHZ" set to 0.

INSTALLATION NOTES

BRINGING UP CP/M[tm]

SOFTWARE LISTINGS

INSTALLATION INSTRUCTIONS:

THE **CONDUCTOR** IS SHIPPED FULLY ASSEMBLED AND READY TO PLUG INTO ANY STANDARD S-100 SYSTEM. IT IS DESIGNED TO INTERFACE WITH MOST "SHUGART COMPATIBLE" DRIVES AS THEY ARE SHIPPED FROM THE FACTORY IN THEIR STANDARD CONFIGURAIONS. OTHER NON-SHUGART COMPATIBLE DRIVES SUCH AS THE PERSCI 277 WILL REQUIRE AN ADAPTER CABLE WHICH IS AVAILABLE FROM DATASPEED.

IT IS ADVISABLE TO CONFIGURE THE "A" DRIVE TO ENABLE STEPPER POWER FROM "DS" (DRIVE SELECT) RATHER THAN "HL" (HEAD LOAD). THIS IS AN EASY CHECK TO MAKE: ON THE SHUGART DRIVES, FOR INSTANCE, LOOK BENEATH THE DRIVE FOR THE TWO ADJACENT JUMPER PLUGS MARKED "DS" AND "HL". ENSURE THAT THE SHORTING PLUS IS JUMPERED FOR "DS" AND NOT "HL". IF YOU ORDERED YOUR DRIVE SYSTEM FROM **DATASPEED**, THIS HAS ALREADY BEEN DONE.

BEFORE INSTALLING THE CONDUCTOR, IT IS ADVISABLE TO REMOVE ALL CARDS FROM THE COMPUTER WHICH ARE NOT ABSOLUTELY NECESSARY FOR THE INITIAL BRING UP. YOU WILL NEED A CPU, 16K OF RAM ADDRESSED AT 0000H, SOME TYPE OF I/O AND EITHER A FRONT PANEL OR MONITOR PROGRAM. TWO IMPORTANT POINTS:

1. THE **CONDUCTOR** OCCUPIES MEMORY AT F000H-F023H, THEREFORE NO OTHER BOARD CAN OCCUPY THIS ADDRESS. PORT F0H IS ALSO USED FOR BOARD CONTROL.

2. THE **CONDUCTOR** EXPECTS TO SEE "MIRRORED" ADDRESSING. THAT IS, IF YOUR CPU BOARD DOES NOT PROVIDE THIS FEATURE, THE "CONDUCUTOR" WILL NOT WORK DIRECTLY. KNOWN CPU BOARDS WHICH DO NOT PROVIDE THIS FEATURE ARE THE CROMEMCO 3BC, DELTA CPU, JADE BIG "Z".

IF ITEM #1 OR #2 IS A PROBLEM, STOP HERE AND REFER TO THE APPENDIX FOR THE SOLUTION BEFORE PROCEEDING FURTHER.

****

THE MOST IMPORTANT POINT TO REMEMBER IS TO UNDERSTAND HOW THE **CONDUCTOR** OPERATES. THE **CONDUCTOR** USES WHAT IS KNOWN AS "PROGRAMMED DATA TRANSFER" INSTEAD OF "DMA". ESSENTIALLY THE **CONDUCTOR** TRANSFERS ONE BYTE AT A TIME FROM THE DISK TO MEMORY THROUGH THE CPU. IN DOUBLE-DENSITY, ESPECIALLY AT 2 MHZ, THE DATA COMES OFF THE DISK SO FAST THAT IS IS VIRTUALLY IMPOSSIBLE TO SYNCHRONIZE THE CPU WITH THE DATA. THERE IS BARELY 16 MICRO SECONDS BETWEEN BYTES TO GET THE BYTE INTO THE ACCUMULATOR, STORE IT IN MEMORY, INCREMENT THE ADDRESS AND LOOP BACK TO GET THE NEXT BYTE. TO SOLVE THIS PROBLEM, THE **CONDUCTOR** USES THE FOLLOWING INSTRUCTION SEQUENCE ALONG WITH SOME FANCY HARDWARE TECHNIQUES TO ACHIEVE THE VERY FAST DATA TRANSFER METHOD:

FIRST: SINCE THE DISK CONTROLLER DATA REGISTER IS MEMORY MAPPED AT F022H, WE INITIALIZE THE "DE" REGISTERS WITH THIS ADDRESS, AND LOAD THE ACCUMULATOR WITH DATA FROM THIS REGISTER

SECOND: WE WILL TRANSFER THIS DATA TO MEMORY USING THE "BC" REGISTER AS THE POINTER, INCREMENTING THE "BC" REGISTER AFTER EACH MEMORY STORE.

THIRD: WE WILL USE A FINAL INSTRUCTION WHICH WILL LOOP US BACK TO THE FIRST INSTRUCTION UNTIL THE ENTIRE TRANSFER IS COMPLETE. HOWEVER, A NORMAL "JUMP" INSTRUCTION IS NOT FAST ENOUGH TO GET US BACK IN TIME SO WE USE A "PCHL" INSTURUCTION WITH THE "HL" REGISTERS HAVING PREVIOUSLY BEEN LOAED WITH THE ADDRESS OF THE FIRST INSTRUCTION.

SO HERE IS WHAT WE HAVE FROM A SOFTWARE POINT OF VIEW:

```
LOOP:   LDAX D      ; GET DATA BYTE FROM DATA REGISTER
        STAX B      ; STORE IT IN MEMORY ADDR BY "BC"
        INX B       ; INCREMENT MEMORY POINTER
        PCHL        ; JUMP BACK TO LOOP
```

NOW A CURSORY GLANCE AT THE ABOVE INSTRUCTION SEQUENCE WILL REVEAL TWO FLAWS. FIRST THERE IS NO WAY TO GET OUT OF THE ABOVE SEQUENCE AND SECOND THE EXECUTION TIME IS APPROXIMATELY 15.5 USEC (AT A 2MHZ CLOCK RATE) NOT 16 USEC SO HOW DOES THE CPU SYNCRHONIZE WITH THE 16 USEC TRANSFER TIME?

THE ANSWER TO THE SECOND PROBLEM IS AS FOLLOWS. THE **CONDUCTOR** USES THE "WAIT" LINE ON THE S-100 BUS TO PUT THE CPU INTO A WAIT STATE UNTIL THE DATA IS AVAILABLE IN THE WESTERN DIGITAL CONTROLLER CHIP; THAT IS, AFTER THE "PCHL" INSTRUCTION WHEN WE ATTEMPT TO GET THE NEXT DATA BYTE, THE WESTERN DIGITAL CONTROLLER CHIP WILL AUTOMATICALLY PUT THE CPU INTO A WAIT STATE FOR A FEW WAIT STATES JUST LONG ENOUGH UNTIL THE DATA BYTE IS AVAILABLE. IT WILL THEN RELEASE THE WAIT STATE AND THE CYCLE WILL REPEAT. THE CPU WILL THUS ALWAYS BE SYNCRHONIZED WITH THE DISK.

NOW HOW DO WE GET OUT OF THE INSTRUCTION LOOP? THIS ANSWER AGAIN IS FOUND IN THE **CONDUCTOR** HARDWARE DESIGN. BEFORE ENTERING THE TRANSFER LOOP, A RETURN ADDRESS IS STORED AT LOCATION 0038H. ONCE THE COMMAND FOR A DATA TRANSFER IS ISSUED AND THE TRANSFER LOOP IS ENTERED, THE CPU WILL REMAIN IN THAT LOOP **UNTIL** THE COMMAND IS COMPLETED. WHEN A COMMAND SUCH AS A "READ TRACK" IS COMPLETED THE WESTERN DIGITAL CONTROLLER CHIP WILL ISSUE AN INTERRUPT. THIS INTERRUPT WILL FORCE THE CPU TO LEAVE THE TRANSFER ROUTINE AND JUMP TO THE INTERRUPT SERVICE ROUTINE WHICH WILL THEN TRANSFER CONTROL BACK TO THE MAIN PROGRAM.

THE IMPORTANT POINT TO REMEMBER IS THAT THE CPU AFTER GETTING THE INTERRUPT, WILL LOOK FOR AN INTERRUPT INSTRUCTION TO PLACED ON THE DATA BUS. THE **CONDUCTOR** DOES NOT SUPPLY THE INTERRUPT INSTRUCTION SO THE CPU WILL NORMALLY SEE NOTHING OTHER THAN "FF" (HEX) WHICH IS THE DATA LINES LEFT HIGH BY THE CPU PULL-UP RESISTORS. HOWEVER "FF" (HEX) IS REALLY A "RST 7" INSTRUCTION WHICH CAUSES THE CPU TO BRANCH TO 0038H WHICH IS EXACTLY WHERE WE HAVE STORED OUR ROUTINE TO RETURN US TO THE MAIN PROGRAM.

THUS COMBINING SOME VERY CLEVER SOFTWARE AND HARDWARE TECHNIQUES, THE **CONDUCTOR** CAN TRANSFER DOUBLE-DENSITY DATA AT A 2

MHZ CLOCK RATE WITHOUT NEEDING BUFFERING THROUGH ON BOARD RAM.

YOU MUST ENSURE THAT YOUR CPU DOES IN FACT HAVE ~~FULL~~ PULL UP RESISTORS ON THE DATA LINES TO ENSURE THAT THE "RST 07" INSTRUCTION IS EXECUTED. IF YOUR CPU IS "I-EEE S-100" STANDARD, THEN YOU HAVE NOTHING TO WORRY ABOUT. CERTAIN CPU'S SUCH AS DELTA'S, DO NOT HAVE 1K PULL UP RESISTORS SO THEY MUST BE ADDED TO THE DATA "IN" LINES. OTHERWISE THE DATA LINES WILL FLOAT EITHER HIGH OR LOW AND THUS IT IS UNKNOWN WHAT THE STATE OF THE BUS WILL BE IMMEDIATELY AFTER THE INTERRUPT IS ISSUED.

wait = wait state p.8

pull up resistors =

***** THE INITIAL BOOT *****

HAVING ENSURED THAT ALL IS IN ORDER AND YOU ARE READY TO BOOT UP THE **CONDUCTOR**, USE THE FOLLOWING SEQUENCE UNTIL YOUR CONFIDENCE LEVEL IN THE SYSTEM IS ENSURED.

1. WITH YOUR DISK DRIVE POWER "ON" AND COMPUTER POWER "OFF", YOU SHOULD **NOT** SEE ANY OF THE DRIVE LIGHTS "ON". IF YOU DO, YOU PROBABLY HAVE YOUR CABLE INSTALLED BACKWARDS. REVERSE THE CABLE AT THE **CONDUCTOR** IF THIS OCCURS.

2. IF YOUR CPU HAS A "POWER-ON-JUMP" FEATURE, SET IT FOR F000 (HEX).

3. WITHOUT INSERTING ANY DISKETTES, TURN ON THE CPU POWER. YOU SHOULD NOW SEE EITHER ONE OR BOTH OF THE DISK DRIVE LIGHTS COME ON, THIS IS NORMAL. YOU MAY ALSO HEAR THE DRIVES RESTORE TO TRACK "00" IF THEY WERE NOT ALREADY THERE.

4. DEPRESS THE "RESET" BUTTON. THIS SHOULD RESTORE THE DRIVE(S) TO TRACK "00" IF THEY DID NOT DO SO ON POWER UP.

5. IF YOUR CPU DOES NOT HAVE A "POWER-ON-JUMP" FEATURE BUT YOU HAVE EITHER A FRONT PANEL OR MONITOR PROGRAM, ISSUE A "GO" COMMAND TO F000 (HEX). THIS WILL FORCE THE CPU TO EXECUTE THE 32 INSTRUCTIONS CONTAINED WITHIN THE **CONDUCTOR'S** BOOTSTRAP PROM AT ADDRESS F000-F01F (HEX). THESE INSTRUCTIONS CAUSE THE CPU AND CONTROLLER TO,

A) SELECT DRIVE #1 ONLY, DE-SELECTING ALL OTHER DRIVES
B) WAIT FOR A DISKETTE TO BE LOADED INTO DRIVE #1
C) WAIT FOR THE DOOR TO BE CLOSED
D) FINALLY, LOAD THE FIRST SECTOR OF THE FIRST TRACK (TRACK 00) INTO MEMORY STARTING AT LOCATION 0000 (HEX). THIS WILL CONTINUE UNTIL THE SECTOR IS READ AND THEN AN INTERRUPT WILL OCCUR WHICH SHOULD NORMALLY FORCE THE CPU TO LOCATION 0038 (HEX).

5. IF YOU ORDERED THE OPTIONAL **CP/M OPERATING SYSTEM** FROM **DATASPEED**, THE REMAINING CODE TO COMPLETE THE TRANSFER IS ALREADY ON THE DISK AND THE LOADING WILL CONTINUE WITH DRIVE #1 PERFORMING TWO "SEEK" OR STEP OPERATIONS.

****

Port
00 console
01 input port
01 output

BRINGING UP THE CP/M OPERATING SYSTEM

IF YOU ORDERED THE OPTIONAL **CP/M OPERATING SYSTEM**, YOU WILL HAVE TO "PATCH" THE INPUT/OUTPUT COMMANDS TO YOUR PARTICULAR CONFIGURATION. THE "PATCHING" PROCESS IS SIMPLE BUT THE PERMANENT "PATCHING" OF THE **CP/M** SYSTEM IS NOT SO SIMPLE ESPECIALLY IF YOU DO NOT HAVE PREVIOUS **CP/M** EXPERIENCE.

DATASPEED ISSUES ITS **CP/M** FOR THE **CONDUCTOR** IN A "STANDARDIZED" CONFIGURATION. THAT IS, IT IS ASSUMED THAT YOUR CONSOLE STATUS WILL BE LOCATED AT PORT "00", INPUT PORT AT PORT "01" AND OUTPUT PORT ALSO AT PORT 01. BIT 6 IS ASSUMED TO BE THE INPUT STATUS BIT (ACTIVE HIGH), BIT 7 IS ASSUMED TO BE THE OUTPUT STATUS BIT. SINCE IT CANNOT BE GUARANTEED THAT YOUR I/O CONFORMS TO THIS SCHEME, THE "CONOUT" ROUTINE HAS A "TRAP" INSTRUCTION WHICH CAUSES THE OUTPUT ROUTINE TO ENTER AN ENDLESS LOOP. THIS ENSURES THAT AFTER THE BOOT ROUTINE IS COMPLETED THE CPU DOES NOT READ THE WRONG OUTPUT OR INPUT PORT AND BEGIN WRITING ON THE DISK. IT IS AT THIS POINT THAT YOU CAN BEGIN THE PATCHING PROCESS.

REFERRING TO THE **CBIOS** LISTING, YOU WILL FIND AT THE END OF THE BIOS LISITNG A SECTION FOR "SIMPLE I/O HANDLERS ETC...". IN THE "CONOUT" ROUTINE, YOU WILL FURTHER FIND THE "TRAP" INSTRUCTION. AFTER INPUTTING THE STATUS PORT AND MASKING OUT THE OUTPUT STATUS BIT, WE WOULD EXPECT TO FIND EITHER A "JZ" OR A "JNZ" INSTRUCTION. INSTEAD WE HAVE SUBSTITUTED AN UNCONDITIONAL "JMP". THIS IS WHERE THE CPU IS NOW HUNG UP. WITH EITHER YOUR FRONT PANEL OR MONITOR PROGRAM, REGAIN CONTROL OF THE CPU. WITH A FRONT PANEL, DEPRESS THE STOP OR RESET SWITCH. WITH A MONITOR PROGRAM, DEPRESS THE RESET BUTTON AND GET BACK INTO THE MONITOR.

NOW EXAMINE MEMORY AT THE LOCATIONS IN THE CBIOS LISTING CORRESPONDING TO THE CBIOS "CONOUT" ROUTINE. IF YOUR I/O CORRESPONDS EXACTLY TO THE ENTIRE CONST, CONIN AND CONOUT ROUTINES, REPLACE THE UNCONDITIONAL "JMP" (C3) INSTRUCTION WITH A "JZ" (CA) INSTRUCTION. THIS IS NORMALLY THE ONLY CHANGE YOU HAVE TO MAKE. (YOU MAY NEED A "JNZ" (C8) SO BE CAREFUL). NOW JMP TO LOCATION 0000 (HEX). (WITH A FRONT PANEL, EXAMINE 0000 AND RUN. WITH A MONITOR PROGRAM, ISSUE A "GO" TO 0000 (HEX)).

CP/M SHOULD NOW DISPLAY ITS PROMPT ("A>") ON YOUR CONSOLE. YOU SHOULD BE ABLE TO DO A DIRECTORY LISTING OF ALL THE FILES ON THE DISK BY TYPING "DIR" FOLLOWED BY A CARRIAGE RETURN. IF YOU HAVE REACHED THIS POINT, THE EASY PART IS OVER. NOW YOU MUST PERMANENTLY PATCH THE I/O ROUTINES.

REFER TO THE CP/M MANUALS FOR THE "SYSGEN" AND "MOVCPM" PROGRAMS AND STUDY CAREFULLY HOW THEY INTERRELATE AND HOW TO MODIFY YOUR SYSTEM. REMEMEBER, THE TWO PARTS OF THE SYSTEM YOU MAY HAVE TO MODIFY FOR MOST APPLICATIONS ARE THE BOOT AND BIOS. IN THE BOOT ROUTINE, THERE ARE SEVERAL "NOP" INSTRUCTIONS. THESE CAN BE REPLACED WITH ANY CODE YOU DESIRE. SUCH CODE CAN BE USED FOR INITIALIZING YOUR I/O, CLEARING A SCREEN MONITOR ETC. THE BIOS PORTION IS WHERE YOU MAKE YOUR PERMANENT CHANGES FOR INPUT/OUTPUT PORT ASSIGNMENTS, ADD PRINTER DRIVERS ETC.

THE CONDUCTOR

OPTIONAL MODIFICATIONS

THE FOLLOWING ARE MODIFICATIONS WHICH MAY BE NECESSARY IF YOU DO NOT USE A "STANDARD" CONFIGURATION IN YOUR SYSTEM.

PHANTOM OPTION:

IF YOU ARE USING THE FULL 64K OF MEMORY BESIDES THE **CONDUCTOR**, YOU MUST INSTALL THE PHANTOM OPTION. YOU MUST ALSO ENSURE THAT YOUR MEMORY BOARD HAS THE CAPABILITY OF MONITORING THE PHANTOM LINE. WHAT HAPPENS IS AS FOLLOWS, IF YOU MAKE THIS MOD. WHEN THE CONDUCTOR IS ADDRESSED, THE PHANTOM LINE (S-100 BUS, PIN 67) IS PULLED LOW. YOUR MEMORY BOARD WHICH IS MONITORING THIS LINE, SEE THIS AND DE-SELECTS ITSELF ONLY WHILE THE **CONDUCTOR** IS BEING ADDRESSED.

MODIFICATION: INSTALL A WIRE FROM S-100 BUS, PIN 67, TO PIN #2 OF IC-22. LOOK CAREFULLY AS REV-A **CONDUCTOR** BOARDS HAVE JUMPER PADS INSTALLED FOR THIS PURPOSE.

NON-MIRRORED ADDRESSING:

BEFORE THE DEVELOPMENT OF THE Z-80 CHIP, THE 8080 PROCESSOR USED MIRRORED ADDRESSING. THAT IS DURING I/O, THE UPPER 8 BITS OF THE ADDRESS BUS MIRRORED OR DUPLICATED THE LOWER 8 BITS OF THE ADDRESS BUS. THE **CONDUCTOR** USES CONTROL BITS FROM BOTH HALVES OF THE ADDRESS BUS AND ALSO ASSUMES THAT BOTH HALVES WILL BE IDENTICAL. THE Z-80 CHIP HOWEVER HAS UNIQUE INPUT/OUTPUT INSTRUCTIONS WHICH CAN MAKE THE TWO HALVES QUITE DIFFERENT. MOST CPU BOARD DESIGNERS HAVE BUILT IN THE OPTION OF SELECTING EITHER MIRRORED OR NON-MIRRORED ADDRESSING MODES.

NOW IF YOUR CPU BOARD DOES NOT HAVE THIS CAPABILITY OR IF YOU WISH TO USE THE Z80 I/O INSTRUCTIONS, THE FOLLOWING CHANGES ARE REQUIRED:

1) EVERY PLACE YOU DO AN INPUT OR OUTPUT WITH THE CONDUCTOR IN THE BIOS AND BOOT ROUTINES, YOU MUST REPLACE THESE INSTRUCTIONS VERY CAREFULLY WITH THE APPROPRIATE Z80 CODES. FOR INSTANCE, WHEN INPUTTING STATUS FROM THE CONDUCTOR YOU WOULD NORMALLY USE THE FOLLOWING CODE:

```
DB FO          IN DPORT
```

THIS MUST BE REPLACED WITH THE FOLLOWING CODE:

```
3E FO          MVI A,DPORT
DB FO          IN DPORT
```

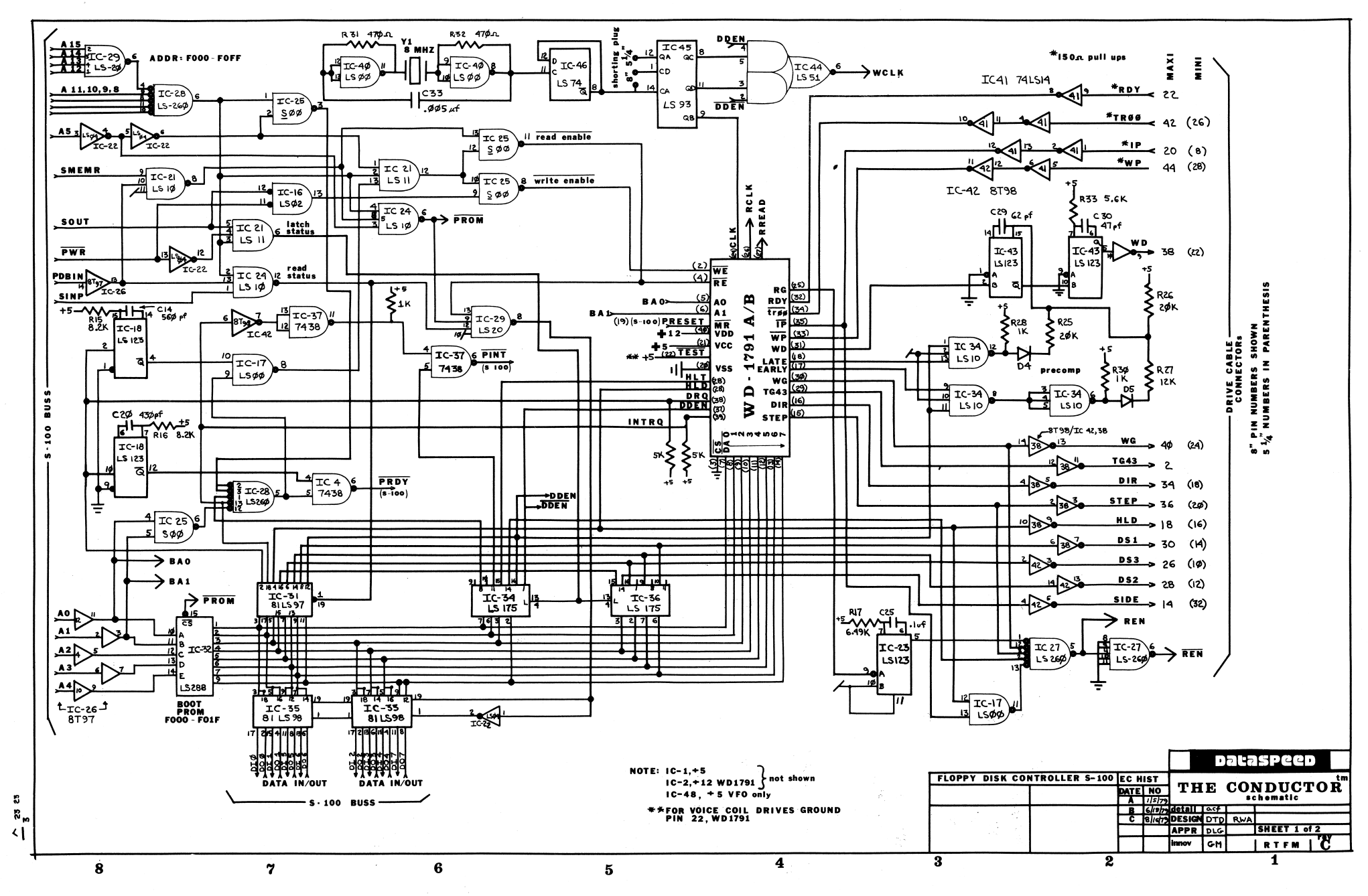
WHAT HAPPENS WITH Z80 I/O (NON-MIRRORED ADDRESSING) IS THAT THE BYTE CONTAINED IN THE ACCUMULATOR DURING AN INPUT IS PLACED ON THE UPPER EIGHT BITS OF THE ADDRESS WHILE THE REQUESTED PORT NUMBER IS PLACED ON THE LOWER 8 BITS. THUS PUTTING "F0" IN THE ACCUMULATOR BEFORE INPUTTING "F0" ENSURES THAT THE "NON-MIRRORED ADDRESSING" WILL LOOK LIKE MIRRORED ADDRESSING.

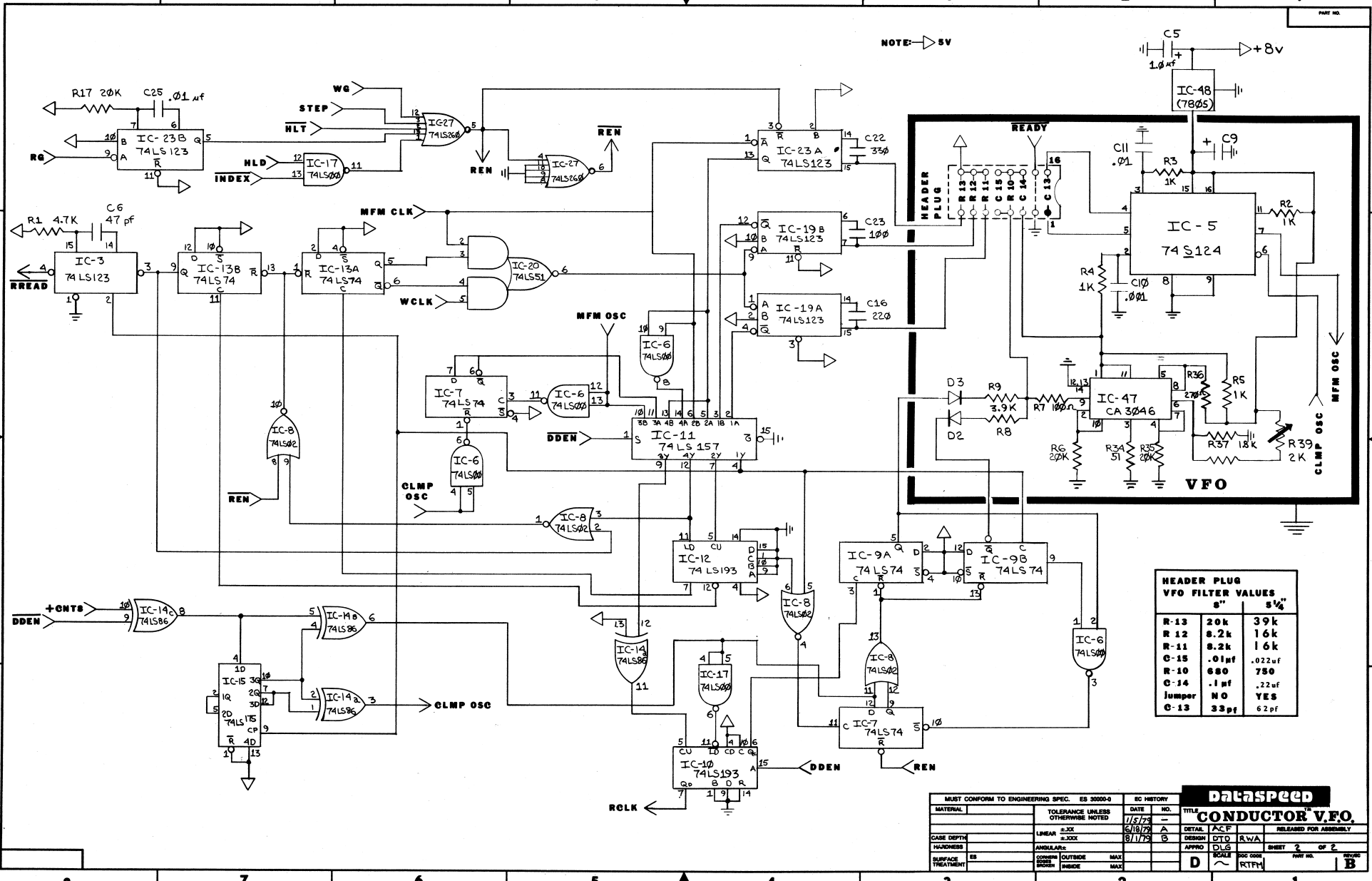
2) A SIMILAIR SITUATION OCCURS WITH OUTPUT INSTRUCTIONS BUT THE MODIFICATIONS REQUIRED BECOME MORE COMPLEX AS YOU HAVE TO BE VERY CAREFUL TO ENSURE THAT YOU DO NOT CRASH THE "BC" OR "DE" REGISTERS WHILE DOING THE I/O.

3) FINALLY, THE **CONDUCTOR** BOOT PROM ON BOARD ASSUMES YOU WILL BE USING MIRRORED ADDRESSING. THEREFORE YOU WILL HAVE TO CHANGE ITS CODE. YOU CAN EITHER "BURN" IN A NEW PROM WITH THE APPROPRIATE CODE AND LOCATE IT ELSEWHERE IN MEMORY OR YOU CAN ORDER THE SPECIAL Z80 PROM FROM **DATASPEED.**

****

THE EASIEST SOLUTION TO THE Z80 I/O PROBLEM IS TO CALL **DATASPEED** AND ORDER A REPLACEMENT DISKETTE AND PROM SPECIFICALLY SET UP FOR Z80 I/O.


ADDR: F000 - F0FF
R31 470Ω
Y1 8 MHZ
R32 470Ω
C33 .005 µf
shorting plug
WCLK
DDEN
*150Ω pull ups
IC41 74LS14
IC42 8T98
MAXI
MINI
*RDY 22
*TR00 42 (26)
*IP 20 (8)
*WP 44 (28)
WD 38 (22)
WG 40 (24)
TG43 2
DIR 39 (18)
STEP 36 (20)
HLD 18 (16)
DS1 30 (14)
DS3 26 (10)
DS2 28 (12)
SIDE 14 (32)
REN
read enable
write enable
PROM
latch status
read status
PRESET
PINT (S-100)
PRDY (S-100)
INTRQ
WD-1791 A/B
RCLK
RRED
BA0
BA1
TEST
HLT
HLD
DRQ
DDEN
S-100 BUSS
DRIVE CABLE CONNECTORS
8" PIN NUMBERS SHOWN
5 1/4" NUMBERS IN PARENTHESIS
BOOT PROM F000 - F01F
DATA IN/OUT
DATA IN/OUT
S-100 BUSS
NOTE: IC-1,+5
IC-2,+12 WD1791 } not shown
IC-48, +5 VFO only
**FOR VOICE COIL DRIVES GROUND PIN 22, WD1791
FLOPPY DISK CONTROLLER S-100
EC HIST
DATE NO
A 1/5/79
B 6/9/79
C 8/14/79
Dataspeed
THE CONDUCTOR tm
schematic
detail
DESIGN
APPR
Innov
SHEET 1 of 2
RTFM
REV C



HEADER PLUG VFO FILTER VALUES	8"	5¼"
R-13	20k	39k
R 12	8.2k	16k
R-11	8.2k	16k
C-15	.01mf	.022uf
R-10	680	750
C-14	.1 mf	.22uf
Jumper	NO	YES
C-13	33pf	62pf

MUST CONFORM TO ENGINEERING SPEC. ES 30000-0

MATERIAL

TOLERANCE UNLESS OTHERWISE NOTED
LINEAR ±.XX ±.XXX
ANGULAR±:

CASE DEPTH
HARDNESS
SURFACE TREATMENT ES
CORNER EDGES BROKEN OUTSIDE MAX INSIDE MAX

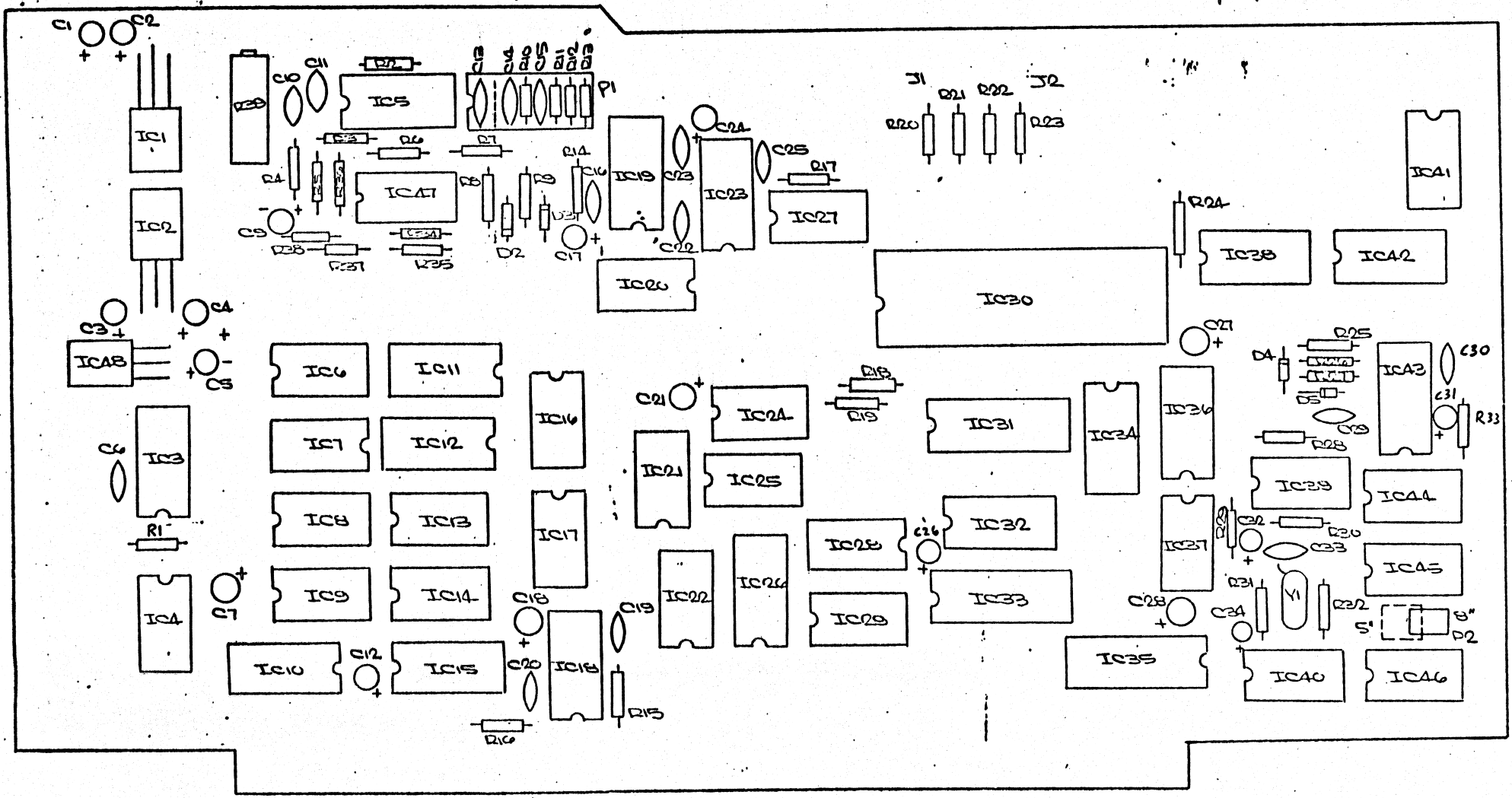
EC HISTORY

DATE	NO.
1/5/79	—
6/18/79	A
8/1/79	B

Dataspeed

TITLE CONDUCTOR V.F.O.

DETAIL	ACF		RELEASED FOR ASSEMBLY
DESIGN	DTD	RWA	
APPRO	DLG		SHEET 2 OF 2
SCALE	DOC CODE	PART NO.	REVISIO
D	RTFM		B



dataspeed

1302 NOE STREET, SAN FRANCISCO, CALIFORNIA 94131

TELEPHONE (415) 282 · 5616

```
SMLPRM  MACRO-80 3.31   29-Jun-79       PAGE    1

                        TITLE   SMLPRM
                        SUBTTL  32 byte PROM in the Conductor

                        .Z80
00F0                    FDPORT  EQU     0F0H    ;Location of memory mapped and I/O ports
F000                    ORIGIN  EQU     0F000H  ;Execution address
0000                    DENSTY  EQU     0       ;DENSITY=80H; Single density DENSTY=01 Double density

F020                    COMREG  EQU     FDPORT+80H+20H
00F0                    CTLREG  EQU     FDPORT

                        .PHASE  ORIGIN
F000  3E 35             SMLPRM: LD      A,35H OR DENSTY
F002  D3 F0                     OUT     (CTLREG),A
F004  11 F020                   LD      DE,COMREG       ;2
F007  21 F01B                   LD      HL,RLOOP
F00A  01 0000                   LD      BC,0
F00D  1A                HWAIT:  LD      A,(DE)
F00E  17                        RLA
F00F  DA F00D                   JP      C,HWAIT
F012  3E 84                     LD      A,84H   ;Read command
F014  12                        LD      (DE),A
F015  1E 23                     LD      E,23H
F017  3E 32                     LD      A,32H OR DENSTY
F019  D3 F0                     OUT     (CTLREG),A
F01B  FB                RLOOP:  EI
F01C  1A                RLoop:  LD      A,(DE)
F01D  02                        LD      (BC),A
F01E  03                        INC     BC
F01F  E9                        JP      (HL)

0020                    LENGTH  EQU     $-SMLPRM
                        GLOBAL  LENGTH,FDPORT,ORIGIN,DENSTY
                        .DEPHASE
                        END     SMLPRM

SMLPRM  MACRO-80 3.31   29-Jun-79       PAGE    S
32 byte PROM in the Conductor

Macros:

Symbols:
COMREG  F020    CTLREG  00F0    DENSTY  0000I   FDPORT  00F0I
HWAIT   F00D    LENGTH  0020I   ORIGIN  F000I   RLOOP   F01B
SMLPRM  F000

No  Fatal error(s)
```

```
;
; CBIOS BOOT LOADER FOR 8" DRIVES
;
; THIS LOADER IS INTERRUPT DRIVEN AND SELF-MODIFYING
; THE ONLY RECOMMENDED MODIFICATIONS ARE TO THE
; END OF THE PROGRAM WHERE IT IS POSSIBLE TO
; ADD ANY I/O INITIALIZATION THAT MAY BE
; NECESSARY. THERE IS ALSO SPACE JUST BEFORE THE
; ORG 38H STATEMENT IF NEEDED.
;

LOADPOINT EQU 2900H
WBOTE   EQU     LOADPOINT+1500H
        ORG     0
START   XRA     A
        STA     38H
        LXI     SP,100H
        LXI     D,0F022H
        LXI     H,LOOP1
        LXI     B,LOADPOINT     ; THIS POINTS TO BOTTOM OF CCP - RELOCATES
        MVI     A,2
        STAX    D
        INX     D
        MVI     A,94H
        STA     0F020H
        MVI     A,0B2H
        OUT     0F0H
        EI
LOOP1:  LDAX    D
        STAX    B
        INX     B
        PCHL
        ORG     38H
        JMP     0
        MVI     A,0B5H
        OUT     0F0H
        MVI     A,1
        STA     0F022H
        STA     0F023H
        MVI     A,11H
        STA     0F020H
WAIT1:  IN      0F0H
        ANI     2
        JNZ     WAIT1
        LXI     H,(INT1*256)+JMP
        SHLD    38H
        LXI     H,LOOP2
        MVI     A,94H
        STA     0F020H
        MVI     A,0B2H
        OUT     0F0H
        EI
LOOP2:  LDAX    D
        STAX    B
        INX     B
        PCHL
INT1:   XRA     A       ; SELECT DRIVE 0 FOR CP/M
        STA     4
        ORG     7DH
        JMP     WBOTE
        END
```

```
;                   SKELETAL CBIOS FOR FIRST LEVEL OF CP/M
ALTERATION
;
;       NOTE :  MSIZE DETERMINES WHERE THIS CBIOS IS LOCATED
MSIZE     EQU       16                  ;CP/M VERSION MEMORY SIZE
IN KILOBYTES
PATCH     EQU       MSIZE*1024-2*256          ;START OF THE CBIOS
PATCH
;
;         WE WILL USE A SCRATCH AREA STARTING AT 40H
;          FOR HOLDING THE VALUES OF:
;                   TRACK     =  LAST SELECTED TRACK
;                   SECTOR  =  LAST SELECTED SECTOR
;                   DMAAD     =  LAST SELECTED DMA ADDRESS
;                   DISKNO    =  LAST SELECTED DISK NUMBER
;         (NOTE THAT ALL ARE BYTE VALUES EXCEPT FOR DMAAD)
;
DSIDED    EQU       -1        ; DOUBLE SIDED VERSION
SCRAT     EQU       40H                 ;START OF SCRATCH AREA
TRACK     EQU       SCRAT               ;CURRENT TRACK ON DRIVE 0
TRAK1     EQU       TRACK+1             ;CURRENT TRACK ON DRIVE 1
TRAK2     EQU       TRAK1+1
TRAK3     EQU       TRAK2+1
SECTOR    EQU       SCRAT+4             ;CURRENTLY SELECTED
SECTOR
DMAAD     EQU       SCRAT+5             ;CUURRENT DMA ADDRESS
DISKNO    EQU       SCRAT+7             ;CURRENT DISK NUMBER
DUMMY     EQU       DISKNO+1  ;MUST BE 0 FOR DOUBLE ADD
COM       EQU       DUMMY+1   ; COMMAND (R/W)
;
;
          ORG       PATCH     ;ORIGIN OF THIS PROGRAM
CBASE     EQU       (MSIZE-16)*1024 ;BIAS FOR SYSTEMS LARGER THAN
16K
CPMB      EQU       CBASE+2900H         ;BASE OF CP/M (= BASE OF
CPP)
CPM2      EQU       CPMB+(25*128)
BDOS      EQU       CBASE+3106H         ;BASE OF RESIDENT PORTION
OF CP/M
CPML      EQU       $-CPMB              ;LENGTH OF THE CP/M
SYSTEM IN BYTES
NSECTS    EQU       CPML/128  ;NUMBER OF SECTORS TO LOAD ON WARM
START
COMREG    EQU       0F020H  ;1791 COMMAND REGISTER/STATUS
TRKREG    EQU       0F021H    ;   "   TRACK REGISTER
SECREG    EQU       0F022H    ;   "    SECTOR REGISTER
DATREG    EQU       0F023H  ; DATA REGISTER
DPORT     EQU       0F0H
;
;         JUMP VECTOR FOR INDIVIDUAL SUBROUTINES
```

```
            JMP     GOCPM               ;COLD START
WBOTE:
            JMP     WBOOT               ;WARM START
            JMP     CONST               ;CONSOLE STATUS
            JMP     CONIN               ;CONSOLE CHARACTER IN
CO:         JMP     CONOUT              ;CONSOLE CHARACTER OUT
            JMP     LIST                ;LIST CHARACTER OUT
            JMP     PUNCH               ;PUNCH CHARACTER OUT
            JMP     READER              ;READER CHARACTER IN
            JMP     HOME                ;MOVE HEAD TO HOME
POSITION
            JMP     SELDSK              ;SELECT DISK
            JMP     SETTRK              ;SET TRACK NUMBER
            JMP     SETSEC              ;SET SECTOR NUMBER
            JMP     SETDMA              ;SET DMA ADDRESS
            JMP     READ                ;READ DISK
            JMP     WRITE               ;WRITE DISK
;
STAB:       DB      4EH,1AH,3FH,3,7,0F2H,0C0H

DTAB:       DB      0CH,30H,3FH,4,0FH,0E0H,80H

;           INDIVIDUAL SUBROUTINES TO PERFORM EACH FUNCTION
;
WBOOT:      ;SIMPLEST CASE IS TO READ THE DISK UNTIL ALL SECTORS
LOADED
;
            LXI     SP,80H              ;USE SPACE BELOW BUFFER
FOR STACK
            LDA     TRACK
            ORI     80H
            STA     TRACK
            MVI     C,0                 ;SELECT DISK 0
            CALL    SELDSK
            CALL    HOME                ;GO TO TRACK 00

            MVI     A,2         ; START WITH SECTOR 2 - LOADER IS
IN 1
            STA     SECTOR
            LXI     H,CPMB
            SHLD    DMAAD
            MVI     C,0
            MVI     B,NSECTS
WBOT2:      PUSH    B
            CALL    SETTRK
            CALL    READ
            JNZ     WBOOT
            POP     B
            LDA     SECTOR
            INR     A
```

```
            CPI     27
            JNZ     WBOT1
            MVI     A,1
            INR     C
WBOT1:      STA     SECTOR
            LHLD    DMAAD
            LXI     D,80H
            DAD     D
            SHLD    DMAAD
            DCR     B
            JNZ     WBOT2
GOCPM:      MVI     A,0C3H      ; SET UP WARM BOOT JUMP INSTRUCTION
            STA     0           ; AT LOCATION 0
            LXI     H,WBOTE
            SHLD    1
            STA     5           ; AND CBIOS ENTRY POINT AT 5
            LXI     H,BDOS
            SHLD    6
            LXI     H,80H
            LXI     SP,80H
            SHLD    DMAAD
            LDA     TRACK
            ANI     7FH
            STA     TRACK

            LDA     4H          ; GET CURRENT DISK
            MOV     C,A         ; AND GO TO CPM
            JMP     CPMB
;
;
;           I/O DRIVERS FOR THE DISK FOLLOW
;
HOME:       ;MOVE TO THE TRACK 00 POSITION OF THE CURRENT DRIVE
;SET UP H,L TO POINT TO WORD WITH TRACK FOR SELECTED DISK
            LXI     D,TRACK
            LHLD    DISKNO
            IF      DSIDED
            CALL    MASK
            ENDIF
            DAD     D
            MOV     A,M
            ANI     80H
            MOV     M,A
            MVI     A,09H       ; RESTORE COMMAND
            STA     COMREG
HWAIT       IN      DPORT
            ANI     2
            JNZ     HWAIT
            RET
;
```

```
SELDSK: ;SELECT DISK GIVEN BY REGISTER C
;MAKE SURE DUMMY IS 0 (FOR USE IN DOUBLE ADD TO H,L)
          XRA       A
          MOV       B,A
          STA       DUMMY
          MOV       A,C
          STA       DISKNO
          LXI       H,DRTABL
          DAD       B
          MOV       C,M
DSKB      LXI       D,TRACK ; SET UP POINTER TO GET
          LHLD      DISKNO    ; CURRENT TRACK FOR THIS DRIVE
          IF        DSIDED
          PUSH      H
          CALL      MASK
          ENDIF
          DAD       D
          MOV       A,M                 ; GET CURRENT TRACK
          ANI       7FH                 ; STRIP DENSITY BIT
          STA       TRKREG    ; AND UPDATE TRACK REG
          IF        DSIDED
          POP       H
          DAD       D
          ENDIF
          MOV       A,M
          LXI       H,DTAB
          ANI       80H                 ; DENSITY
          JZ        DCOPY     ; IF ANI SET Z FLAG THEN DOUBLE
          LXI       H,STAB    ; ELSE SINGLE
DCOPY:    ORA       C
          OUT       DPORT
          LXI       D,CBASE+3115H
          MOV       A,M
          STAX      D
          INX       H
          MVI       E,3AH
          MVI       C,6
MOV1:     MOV       A,M
          STAX      D
          INX       D
          INX       H
          DCR       C
          JNZ       MOV1
          MVI       A,35
WAIT      MVI       C,82H
WAIT1     DCR       C
          JNZ       WAIT1
          DCR       A
          JNZ       WAIT
          RET
```

```
DRTABL:   DB        35H,55H,3DH,5DH ; DRIVES A, B, C, AND D

MASK:     MOV       A,L
          ANI       1
          MOV       L,A
          RET
;
SETTRK: ;SET TRACK GIVEN BY REGISTER C
;FIRST REFERENCE CORRECT TRACK INDICATOR ACCORDING TO
;SELECTED DISK
          LXI       D,TRACK   ;ADDRESS OF TRACK FOR DISK 0
          LHLD      DISKNO    ;FIND OUT WHICH DISK IS SELECTED
          IF        DSIDED
          CALL      MASK
          ENDIF
          DAD       D
          MOV       A,M       ;DESIRED TRACK
          ANI       7FH
          CMP       C
          RZ
          MOV       A,H
          ANI       80H
          ORA       C
          MOV       M,A
          ANI       7FH
          STA       DATREG    ; AND THE DATA REGISTER FOR A SEEK
          MVI       A,19H     ; SEEK COMMAND
          STA       COMREG
          CALL      HWAIT
          MVI       A,16
          JMP       WAIT

;
SETSEC: ;SET SECTOR GIVEN BY REGISTER C
          MOV       A,C
          STA       SECTOR
          RET
;
SETDMA: ;SET DMA ADDRESS GIVEN BY REGISTERS B AND C
          MOV       L,C       ;LOW ORDER ADDRESS
          MOV       H,B       ;HIGH ORDER ADDRESS
          SHLD      DMAAD     ;SAVE THE ADDRESS
          RET
;
;
WRITE     MVI       A,0A0H
          JMP       RW
READ:     ;PERFORM READ OPERATION.
          ;THIS IS SIMILAR TO WRITE, SO SET UP READ COMMAND AND
```

```
USE
          ;COMMON CODE IN WRITE
          MVI       A,80H     ; SINGLE SECTOR READ COMMAND
RW
; THIS IS THE COMMON CODE SECTION

          STA       COM
          DI
          LHLD      38H       ; SAVE THE INTERRUPT VECTOR
          PUSH      H
          LHLD      3AH       ; ON THE STACK
          PUSH      H
          LXI       H,RDONE   ; SET UP NEW INTERRUPT VECTOR
          SHLD      39H
          MVI       A,0C3H    ; JMP INSTRUCTION
          STA       38H
          MVI       D,10      ; ALLOW 10 RETRIES
RETRY:    PUSH      D

          LDA       SECTOR
          STA       SECREG
          LHLD      DMAAD     ; SET UP SECTOR AND DMA ADDRESS
          LXI       B,DATREG  ; B POINTS TO DATA REGISTER
          XCHG                ; D POINTS TO MEMORY
          LDA       COM       ; GET THE COMMAND
          CPI       0A0H      ; IS IT A WRITE?
          JNZ       R1        ; NO - MUST BE READ
          LXI       H,WLOOP   ; H&L CONTAIN AN INDIRECT JUMP
ADDRESS
          CALL      SETINT
          EI
WLOOP:    LDAX      D         ; GET THE DATA
          STAX      B         ; STORE IT IN THE DATA REG
          INX       D         ; NEXT
          PCHL      ; JUMP BACK TO WLOOP

RDONE:    LDA       COMREG    ; CHECK STATUS
          POP       D
          POP       D
          MOV       B,A
          ANI       7CH
          JZ        RDONE1    ; NO ERRORS
R0:       DCR       D
          JNZ       RETRY     ; RETRY IF NOT GT 10
          MOV       A,B
          RRC
          RRC
          ANI       1FH
          ORI       30H       ; REPORT ERROR TYPE(S)
          MOV       C,A
```

```
            CALL        CO
            MVI         A,1
RDONE1      PUSH        PSW
            IN          DPORT
            ANI         0F8H
            ORI         5
            OUT         DPORT
            POP         PSW
            POP         H
            SHLD        3AH
            POP         H
            SHLD        38H
            ORA         A
            EI
            RET

R1          LXI         H,RLOOP ; INDIRECT JUMP ADDRESS
            CALL        SETINT
            EI
RLOOP       LDAX        B           ; GET DATA
            STAX        D           ; STORE IN MEMORY
            INX         D           ; NEXT
            PCHL                    ; JUMP BACK TO RLOOP


SETINT      STA         COMREG      ;SAVE THE COMMAND WORD FROM A
RSETINT     IN          DPORT       ; GET THE STATUS
            ANI         0F8H        ; TOGGLE CONTROL BITS
            ORI         02H
            OUT         DPORT
            RET

;
; SIMPLE I/O HANDLES FOR CONSOLE, READER/PUNCH, ETC.
; MAY NEED TO BE PATCHED BY USER
;
CONST:      ;CONSOLE STATUS, RETURN 0FFH IF READY, 0 IF NOT
            IN          0
            ANI         40H
            RZ
            MVI         A,0FFH
            RET

CONIN:      ; READ CONSOLE INTO A
            CALL        CONST
            JZ          CONIN
            IN          1
            ANI         7FH
            RET
```

```
CONOUT: ; PUT CONSOLE FROM REG. C

LIST:   ; PUT LIST DEVICE FROM C
;
PUNCH:  ; PUT PUNCH DEVICE FROM C
          IN        0
          ANI       80H
          JNZ       CONOUT
          MOV       A,C
          OUT       1
          RET
READER: ; READ READER DEVICE TO A

          MVI       A,1AH
          RET
          END
```