DIGITAL MICROSYSTEMS, INC.

DSC 3/4 SYSTEM MANUAL

RELEASE: 01-01-81

VER 1.03

Notice

Table of Contents

1. Introduction

This manual describes Digital Microsystems' DSC/3 and DSC/4 computer systems. These systems are based on the Z-80 microcomputer chip. The DSC/3 is a single board system, while the DSC/4 is a MULTI-BUS compatible system capable of supporting up to 512 Kbytes of memory. The features of these two systems are summarized below:

Z-80A at 4 MHz
64K dynamic RAM on the DSC/3
128K to 512K dynamic RAM plus 1K static RAM on the DSC/4
1K PROM monitor
3 RS-232 ports, 1 RS-232 or RS-422 port
2 8-bit parallel ports
Up to 4 floppy disks, single or double density
single or double sided
8" and 14" hard disk options, 11 Mbytes to 28 Mbytes
CP/M, MP/M, OASIS, and HiNET operating systems

The DSC/3 and DSC/4 are available in these configurations:

DSC/3 FDS - floppy disk system
DSC/3 HDS - hard disk system
DSC/3 NS - network station

DSC/4 FDS - floppy disk system
DSC/4 HDS - hard disk system

Each of these systems is packaged differently; however, the bulk of the hardware and software for the different models is identical. This manual describes the characteristics of all these systems; distinctions between systems will be mentioned when necessary.

DSC/3 FDS - floppy disk system

This system consists of a processor board and 2 to 4 floppy drives. It can act as a complete stand-alone computer system. It can also be connected to the HiNET where it can act as either a network station or the network master.

DSC/3 HDS - hard disk system

This system consists of a processor board, a hard disk controller board, 1 to 4 floppy drives, and a hard disk. If the 11 Mbyte hard disk is selected, the system is packaged in the same cabinet as the DSC/3 FDS - the hard disk takes the place of one of the floppy disks. If the 14 Mbyte or 28 Mbyte disk is selected, the system is packaged in a larger cabinet. This system can be a stand-alone system, or a network master.

DSC/3 NS - network station

A network station consists of a processor board packaged inside an ADDS terminal. This system connects to the HiNET. It is an independent work station capable of supporting one user.

DSC/4 FDS and HDS systems

These systems are similar to their DSC/3 counterparts, except that they are multiple board systems. A minimal system contains two boards - the CPU board, and one 128 Kbyte memory board. Up to 4 memory boards may be included in a single system. Other Multibus boards may be added to the system to customize the system to the user's requirements.

Differences between the DSC/2 and the DSC/3 and DSC/4

The DSC/3 and DSC/4 represent a major advancement over the DSC/2. These new Digital Microsystems computers are inherently more reliable due to a substantial chip count reduction. The new models also have a networking capability. The new models are compatible with the DSC/2, except as pointed out below:

1. Double density diskettes written on the DSC/2 CANNOT be read on the DSC/3 or DSC/4. However, single density diskettes ARE compatible between the two machines.

2. The default CP/M port assignments have changed. On the DSC/2, the list device was assigned to port 1, while the reader and punch were assigned to port 2. On the DSC/3 and DSC/4, port 1 is used as the network interface, so the list is assigned to port 2, while the reader and punch are assigned to port 3. As usual, the port assignments can be changed by using the STAT command.

3. The SINGLE and DOUBLE commands are no longer available; their functions are now performed by the ASSIGN command. This command also provides many more new options.

4. There are no jumper options to change baud rates. All ports are initialized to 9600 baud. The SETBAUD command can be used to change a baud rate. If necessary, assembly code can also be used, or the BIOS can be modified. See section 3.7 or 4.3 .

5. A new and more flexible scheme is used to partition the hard disk into logical units. This scheme is described in section 3.6 .

2.1 Initial system check-out and power-up

When you first receive your DSC/3 or DSC/4, it is suggested that you perform a careful visual inspection of the hardware. The computer undergoes extensive tests at the factory and is carefully packaged for shipment, however, it still may be upset in transit.

If you have a floppy-disk or hard-disk based system, remove the 6 phillips screws holding the lid to the chassis and remove the lid. Check that all socketed integrated circuits are firmly seated. Check that all of the cable connections described in section 2.2 are in place, and that the cables themselves are not damaged in any way. Look for any loose material in the cabinet and listen for anything loose.

Check all the jumper connections on the CPU board. These are described in section 2.3. When all appears in order, check that the ON/OFF switch is in the OFF position, and attach the power cord to the back side of the cabinet. Check that you are plugging the computer into the correct voltage as indicated by the serial number tag on the rear of the cabinet. Unless you have a DSC/NS system, an RS-232 cable should be used to connect a 9600 baud terminal to port 0 on the back side of the cabinet. Port 0 is the upper-most of the four ports.

Turn the power switch to the ON position. You ~~should~~ hear the fan and the disk drives go on. Press the INT switch. While holding the INT switch down, press the RST switch. You ~~should also~~ see the message "PROM Monitor 1.xx" displayed on the screen of your terminal (1.xx is the version number). If not, there is a problem, ~~and you should~~ refer to chapter 5 of this manual.

The computer will now be waiting for you to enter a PROM monitor command. These commands are described in section 2.7. If you have a floppy disk attached to your system, the next step is to load and execute CP/M. Insert a CP/M system diskette (label side up, with the label in your hand) into drive A (the right hand drive). Close the door and type "BF". The system ~~should~~ bootstrap and respond with the message:

```
CP/M 1.4xx (or) CP/M 2.2xxx
A>
```

If this does not happen, refer to chapter 5.

2: Net User station with local disk storage

If you have a ~~network station system~~, you must connect your system to ~~the~~ HiNET. Attach an RS-422 cable between your system and the main network cable. Verify that the Network Master computer is operational. Type "BN". The network loader will be loaded into your station and display the following message:

```
HiNET 1.4xx

Login please...
  Name:
```

~~If this does not happen, refer to chapter 5. Otherwise,~~ type RETURN. The loader will ask for your name and password. The system ~~should~~ will then respond with the normal command prompt "A>".

type your *User Name followed by the RETURN (or NEW LINE) key.

HiNet will respond with

```
PASSWORD!
```

type your *Password followed by the RETURN (or NEW LINE) key. Note your password is not printed on the screen, and should not be given to other system users.

HiNet will respond with:

```
A>
```

You are now ready to execute in a familiar CP/M mode.

*Note: Your User Name and Password must have been entered into the HiNet Master Station by the system operations manager, before you can use them to login.

2.2 Cable connections

The cables listed below should be in place and securely installed with the correct polarization. The A.C. and D.C. power cables are keyed to prevent improper connection.

1. A.C. power--2 or 3 wires (AWG 18) twisted together

115 V.A.C.	230 V.A.C.
red--live conductor	red and white--live conductors
white--neutral conductor	
green--chassis ground	green--chassis ground

Note: Red and white wires are striped, with the opposite color on the line side of the power switch.

This cable carries current from the input line filter through the fuse and power switch to the fan, disk drives, courtesy outlet and the D.C. power supply. Connections to the drives and power supply transformer are made with 3-pin connectors. Other connections in the circuit are soldered. The chassis ground wire (green) connects the chassis ground of each disk drive and the ground of the DSC 3, including the logic ground connections of all connectors, to the grounding wire in the line cord.

2. D.C. power to drives--five separate wires (AWG 16) held together with cable ties:

green	ground (two wires)
blue	+5 volts
yellow	-12 volts
brown	+24 volts

This cable connects the disk drives to the D.C. power supply P.C. board. Connections are made with keyed six-pin Molex connectors.

3. D.C. power to CPU (DSC 3) or to motherboard (DSC 4) and to optional HDC--six separate wires (AWG 16) held together with cable ties

green	pins 2 and 7	ground (two wires)
blue	pin 1	+5 volts
red	pin 3	+12 volts
yellow	pin 4	-12 volts
white	pin 8	zener-clipped 60/50 Hz (time base for CPU)

Connections from the power supply P.C. board to the HDC (Hard Disk Controller), if installed, and to the CPU (DSC 3) or

4. Reset and front panel interrupt switches to CPU (DSC 3) or motherboard (DSC 4)--four separate wires twisted together

green	ground
white	reset (switched to ground--normally open)
black	interrupt (switched to ground--normally open)
blue	interrupt (switched to ground--normally closed)

DSC 3: This cable connects to the CPU with a small six pin Molex connector. The connector is not mechanically keyed and must be attached with the green wire closest to the center of the CPU board.

DSC 4: Connections to the motherboard are made with a small four pin molex connector. The connector must be attached with the green wire toward the back of the cabinet.

5. CPU to RS 232 I/O board--twenty-six conductor ribbon cable

This cable connects J1 on the CPU to J5 on the RS 232 I/O board. Twenty-six pin hooded headers are used on the CPU and the RS 232 I/O board with pin 1 marks on both headers and on the connectors attached to the ribbon cable. The user's CRT is normally connected to the uppermost DB 25 connector on the I/O board. See section 2.3 for jumper configurations for the serial ports on the CPU board. See section 2.5 for jumper configurations for the I/O board for various I/O devices.

6. CPU to floppy disk drive(s)--fifty pin ribbon cable

This cable connects J2 on the CPU to the fifty pin edge connectors on the floppy disk drives. Pin 1 is on the left side of both drives and the CPU board (viewed from he front of the cabinet) and is marked on all connectors. The drive which has terminating resistors installed (section 2.4) must be attached to the connector furthest from the CPU on the cable.

7. CPU RS 422 HiNet connection (optional)--nine conductor ribbon cable or ten conductor twisted-pair ribbon cable

The nine pin network connector is connected to J5 on the CPU with this cable. Pin 1 is on the left side on the CPU and of the nine pin connector (viewed from front of cabinet) and is marked J5. See section 2.3 for jumper configuration for HiNet use.

8. CPU parallel port to HDC (optional)--twenty-six conductor ribbon cable

The CPU is interfaced to the HDC with a cable connecting J3 on the CPU (I/O port 01H) to P1h on the HDC. Pin 1 is on the left side on the CPU and on the right side on the HDC. The cables and connectors are marked.

9. CPU parallel port to Centronics printer connector (optional)--twenty-six conductor ribbon cable.

A Centronics-type interface is provided through J4 on the CPU (I/O port 02H). Assigning the twenty-six pins of J4 to the thirty-six pins of the Centronics connector on the back panel is done within the hood of the Centronics connector. Pin 1 of J4 is on the left and is marked and the Centronics connector is polarized.

10. HDC to hard disk (optional)--fifty pin ribbon cable

When a hard disk is installed it is connected to the HDC at P1d. Pin 1 is marked on all connectors.

2.3 CPU Board Jumper Connections

The ZSBC 3 and ZSBC 4 PC boards may be configured for various options by making connections between jumper pins with removable connecting jumper blocks. Their functions are described below. When jumper blocks are not being used to make connections they may be kept on one of the jumper pins.

Start-up options: Following a reset or when power is applied the CPU will start in one of four ways depending on the state of jumpers JP4, JP5, and the front panel interrupt switch. (This uses PROM versons 1.09 and above.)

JP4 open or connected	JP5 open or connected	Starting in PROM monitor (depress interrupt switch when releasing reset)
JP4 open	JP5 open	Boot from floppy disk A
JP4 conected	JP5 open	Boot from HiNET
JP4 open	JP5 connected	Boot from hard disk

JP5 should not be connected when the ZSBC 3 board is used inside a terminal and there is a wire connecting the top pin of JP5 to TP105.

Serial port 1 and HiNet options: Serial port 1 (address 29H) may be configured as an RS 232 asynchronous serial port for connection to a printer, modem, CRT, etc. by installing connecting jumper blocks at JP2 and JP3 to connect the two pins at each location that are closest to the top of the PC board (pins 1 and 2 at both JP2 and JP3). If this is done, the default circuit trace connections on the back of the board between pins 2 and 3 at both JP2 and JP3 must be cut with a sharp instument. Care must be taken not to damage nearby traces in the process. The baud rate of this serial port is software selectable through channel 1 of the counter-timer chip.

The interface to the HiNet network requires serial port 1 to be configured as an RS 422 synchronous serial port. This is accomplished by installing connecting jumper blocks at JP2 and JP3 to connect the two pins at each location that are closest to the bottom of the board (pins 2 and 3 at JP2 and JP3). No jumper blocks are required if the default traces on the back of the board have never been cut. Jumper blocks are to be installed at JP1 and JP11 only if the system is at either end of the HiNet network cable.

Parallel port 02H (Centronics): When the parallel port connection at J4 is used as an output port only (as with a Centronics printer) a connecting jumper block is installed at JP9 to permanently enable the data outputs.

Jumpers at JP6, JP7 and JP8 must be left as installed by Digital Microsystems. JP6 selects one of two patterns stored in the floppy disk interface phase-locked loop PROM. JP7 and JP8 allow the use of a 2716 type EPROM.

The following jumpers are found only on the DSC 4. They may be left as shipped from Digital Microsystems unless other bus masters (e.g. other CPU cards or DMA cards) are added to the system.

BCLOCK/: The DSC 4 provides an 8 MHz buffered bus clock signal to the system bus though JP201. This signal is necessary for the ZSBC 4 board to use bus slaves, for example, the 128K byte memory card. If the BCLOCK/ signal is to be supplied by another bus master the jumper block at JP201 must be removed and the default circuit trace on the back of the board must be carefully cut. This signal must have only one source.

BPRN/: Bus access conflicts in a Multibus (Tm) system may be resolved by a daisy-chain priority resolution scheme, as in the DSC 4 motherboard, or by some other scheme. When such conflicts over bus use arise, access to the bus is granted to the highest priority bus master by bringing its BPRN/ input low. If the ZSBC 4 CPU card is to be the only bus master pins 1 and 2 of JP206 may be left connected either by the default circuit trace or by a connecting jumper block, or the card may be left in the highest priority slot in the motherboard (furthest to the left). If another bus master is to have higher priority than the ZSBC-4 CPU, pins 2 and 3 of JP206 must be connected with a jumper block and the default trace between pins 1 and 2 must be cut. In this case the higher priority master must occupy the leftmost slot in the motherboard and the ZSBC-4 CPU card must be in the next slot to the right. If the other bus master responds to a CBRQ/ bus request from the ZSBC-4 card both will be able to use the bus, interleaving their accesses to bus slaves. The ZSBC-4 card will likewise respond to a CBRQ/ bus request from a lower priority bus master by granting it the bus after completing its current bus cycle. If no other master asserts CBRQ/ the ZSBC-4 will retain control of the bus resulting in quicker memory accesses.

If the user wishes the ZSBC-4 to relinquish and re-acquire the bus for every bus access (if used with a bus master that did not assert CBRQ/, for example) the default trace between pins 2 and 3 of JP202 may be cut, jumper pins installed and a jumper block used to connect pins 1 and 2. If the user wishes the ZSBC 4 not to interfere with a higher priority bus master, the default trace at JP203 may be cut so that the ZSBC-4 will not assert CBRQ/ and will wait until the higher priority master has given up control of the bus.

Interrupts: The ZSBC-4 board can accept two distinct interrupt signals from other devices on the bus or from the front panel interrupt switch. These inputs may each be connected to any of the eight Multibus bus interrupt lines INT0/ (highest priority) to INT7/ (lowest priority). The bus interrupt lines

are available at JP205 in order of priority with INT7/ (lowest priority) on pin 1 (leftmost pin). INT7/ is usually connected to pin 1 (leftmost pin) of JP204. A low logic level on this pin or execution of a HALT instruction will generate a vectored interrupt through the PIO integrated circuit if I/O pin A7 has been programmed to do so (and the Z80 interrupts have been enabled). In the DSC 4 INT7 on the bus is connected to the front panel interrupt switch through a debouncing RS latch.

In addition to the interrupt inputs described above the ZSBC-4 will respond to interrupts from its counter-timer if supplied with a 60 Hz zener-clipped input TTL level signal on pin 77 of the bus. This pin is currently listed as reserved in Intel's specification of the Multibus but is used on some motherboards as -10 volts. If a ZSBC 4 circuit board is used in another Multibus system which does not provide a 60 Hz TTL signal on bus pin 77 the default trace at JP10 must be cut. Software which uses the real time clock, including Digital Microsystem's versions of CP/M and OASIS, will not operate without a 50/60 Hz input to the counter-timer integrated circuit.

The ZSBC 4 can interrupt another bus master by connecting any of the bus interrupt pins of JP205 to pin 2 of JP204. The interrupt is generated by setting I/O pin B3 of the PIO to a high logic state.

Multibus is a trade mark of Intel Corporation.

2.4 Floppy Disk Drive Jumper Options

Two types of floppy disk drives are currently used in DSC 3/4 systems: Shugart SA800R single sided 8 inch and the Shugart SA850R double sided 8 inch. Each of these drives has numerous options in its electronics which are enabled or disabled by various jumper connections on its PC board. In order to accommodate the NEC 765 LSI floppy disk controller chip, many of thse options must be jumpered differently from their state as shipped by Shugart (and also differently from the requirements of the DSC 2).

The following tables show the jumper requirements of the DSC 3/4 when using the SA800R and SA850R as the A or B drive.

SA800R

Jumpers installed as shipped: 800, X, T1, HL, A

All drives:	install	T2, DC, Wirewrap #1 (*)
	remove	Z, B, 18
A drive only:	install	DS1, C
B drive only:	install	DS2, Wirewrap #2

For all 800 drives and systems, jumpers T3,T4,T5, and T6 must be installed on the drive furthest from the controller on the signal cable.

SA850R

All drives:	install	2S, DC, 850, IW, S2, RS, M
		Wirewrap #3
	cut	B, Z (both on Jumper Block at F4)
A drive only:	install	DS1, C-HI
B drive only:	install	DS2, C-D

For all 850 drives and systems, the resistor pack should be installed at E5 on the drive furthest from the controller on the signal cable.

(*) See drive sketches in Appendix D.

2.5 RS 232 I/O Board Jumper Connections

RS 232 connections to the four serial channels of the DSC 3/4 are made through the four DB 25 connectors, J1 thru J4, on the RS 232 I/O board at the back of the computer. Port 0 is connected to the user's consol through J1. It is configured for a CRT terminal and uses only two RS 232 signals, transmitted data and received data. Port 1 is similarly configured if not being used for the RS 422 HiNet I/O port. Ports 2 and 3, corresponding to DB 25 connectors J3 and J4 respectively, may be configured for various RS 232 devices by inserting the correct jumper block in the sockets immediately above the corresponding DB 25 connector. If a socket must be installed, cut the default circuit traces before installation. Connections for the pins of the four DB 25 connectors when jumpered for use with a CRT (default traces) are given in the following table:

DB 25 connectors	I/O Board jumper blocks	J5 on I/O & J1 on ZSBC3/4	ZSBC 3/4 serial ports
J1 pin 1	chassis ground		
pin 2		pin 3	U2 pin 4 port 0 RxD
pin 3		pin 5	U1 pin 6 port 0 TxD
pin 4		pin 1	N.C. (would be CTS)
pin 5		pin 10	pullup (would be RTS)
pin 6		pin 6	pullup (would be DTR)
pin 7	signal ground	pin 13	logic ground
pin 8		pullup on I/O board	
pin 20		pin 2	N.C. (would be DSR)
J2 pin 1	chassis ground		
pin 2	JB pin 1-pin 16	pin 9	U2 pin 1 port 1 RxD
pin 3	JB pin 2-pin 15	pin 8	U1 pin 3 port 1 TxD
pin 4	JB pin 3-pin 14	pin 7	N.C. (would be CTS)
pin 5	JB pin 4-pin 13	pin 14	pullup (would be RTS)
pin 6	JB pin 5-pin 12	pin 16	pullup (would be DTR)
pin 7	signal ground	pin 13	logic ground
pin 8	JB pin 7-pin 10	pullup on I/O board	
pin 11	JB pin 8		
pin 14	JB pin 9		
pin 20	JB pin 6-pin 11	pin 4	N.C. (would be DSR)
J3 pin 1	chassis ground		
pin 2	JB pin 1-pin 16	pin 17	U2 pin 13 port 2 RxD
pin 3	JB pin 2-pin 15	pin 24	U1 pin 8 port 2 TxD
pin 4	JB pin 3-pin 14	pin 12	U4 pin 4 port 2 CTS
pin 5	JB pin 4-pin 13	pin 18	U3 pin 3 port 2 RTS
pin 6	JB pin 5-pin 12	pin 15	U1 pin 11 port 2 DTR
pin 7	signal ground	pin 13	logic ground
pin 8	JB pin 7-pin 10	pullup on I/O board	
pin 11	JB pin 8		
pin 14	JB pin 9		
pin 20	JB pin 6-pin 11	pin 11	U4 pin 13 port 2 DCD

```
J4 pin  1 chassis ground
   pin  2 JB pin  1-pin 16    pin 21    U2 pin 10 port 3 RxD
   pin  3 JB pin  2-pin 15    pin 25    U3 pin  8 port 3 TxD
   pin  4 JB pin  3-pin 14    pin 20    U4 pin  1 port 3 CTS
   pin  5 JB pin  4-pin 13    pin 26    U3 pin 11 port 3 RTS
   pin  6 JB pin  5-pin 12    pin 23    U3 pin  6 port 3 DTR
   pin  7 signal ground       pin 13    logic ground
   pin  8 JB pin  7-pin 10    pullup on I/O board
   pin 11 JB pin  8
   pin 14 JB pin  9
   pin 20 JB pin  6-pin 11    pin 22    U4 pin 10 port 3 DCD
```

On I/O board pullups connect through pin 19 to +12 volts.

The following are examples of jumper blocks for some commonly used RS 232 devices.

Texas Instruments 810 Printer:

The TI 810 printer has an onboard buffer. When the buffer is full, the printer sends a busy signal to the DSC 3/4 by lowering DTR on RS 232 pin 20. The serial I/O port on the ZSBC 3/4 will stop transmission automatically if this signal is connected to its CTS input. This is accomplished by connecting pin 14 to pin 6 instead of pin 3 on the corresponding jumper block.

The DNB option on the TI 810 must be enabled by setting the panel switches to:

1	2	3	4	5	6	7
low	high	high	low	low	high	high

Remove the processor board of the printer (green card ejectors) and jumper E5 to E6 instead of E4. Verify that the resistor network near the printer's RS 232 connector has not been pulled loose before replacing the card cage enclosure.

Teletype model 40:

The teletype model 40 uses RS 232 pin 14 to tell the computer that it is busy and no more characters can be sent to it. This signal may be connected to CTS on the proper serial I/O port of the DSC 3/4 by connecting pin 14 to pin 9 instead of pin 3 on the corresponding jumper block on the I/O board.

Modem:

Most modems require the following pairs of signals to be interchanged (compared to their arrangement for a CRT): RxD and TxD, CTS and RTS, and DCD and DTR.

CRT terminal:

The jumper block connections which are used with a CRT terminal are listed in the table above.

2.6 128K Byte RAM PC Board Jumpers

The 128k byte RAM board has four sets of jumper posts which may be used with connecting jumper blocks to select various options. All post locations have default circuit traces which, unless cut and jumpered otherwise, configure the board to operate as board 0 (lowest 128K bytes of the 1 Mbyte address space of the Multibus) in a DSC 4. If the board is to be used with other bus master Multibus boards or in a DSC 4 with more than one memory card the default traces may need to be cut and connecting jumper blocks installed.

The eight sets of two posts located between U12 and U13 at the lower left corner of the PC board select which of the eight 128K byte blocks of memory space is occupied by the board. The upper two posts are connected by a default trace which must be cut if the board does not occupy the lowest 128K bytes. The pairs of posts are in order so that the board can be made to occupy the nth block of memory space by connecting the nth pair of posts from the top.

The timing of the Z80 CPU used in the ZSBC 4 is such that one CPU wait state may be avoided if the transfer acknowledge (XACK/) from slave boards is brought low slightly before valid data is present on the data bus. When used in the DSC 4 the 128K ram board normally provides this early version of XACK/ thru the default trace between the lower two posts of the three between U9 and U10. If another bus master is to be used with the 128K RAM board this trace may be carefully cut and the upper two post connected with a jumper block to provide a Multibus-conforming XACK/. This will cause the DSC 4 to run approxiamtely 20% slower and may interfere with timing dependent software and is therefore not recommended unless necessary.

When the 128K RAM board detects a parity error it generates an interrupt on INT0/, the highest priority interrupt line on the Multibus. The board also latches the error condition so that software may determine which memory board in system generated the error. This is accomplished by the first memory read operation to any address in either block 3 (60000H to 7FFFFH) or block 6 (C0000H to DFFFFH) of the Multibus address space. A default trace connecting the upper two pins of the set between U6 and U7, selecting a parity error response from block 3. If a parity error response from block 6 is desired the default trace must be cut and the lower two pins connected.

A memory board which has latched a parity error responds to a block 3 or block 6 read by pulling low one of the lower eight Multibus data lines. The data line to be lowered is selected by the eight pairs of jumper pins located between U21 and U22 and should correspond to the address block occupied by the board. A default trace configures the board to lower DAT0/. The pairs of pins are arranged in the following order, from bottom to top:

pair 1 (lowest)	DAT7/
pair 2	DAT5/
pair 3	DAT4/
pair 4	DAT6/
pair 5	DAT3/
pair 6	DAT2/
pair 7 (default)	DAT0/
pair 8 (highest)	DAT1/

Since a memory board configured to respond to a parity error read of block 3 or block 6 will lower XACK/ in response to such a read even if no parity error has occured, no system memory should occupy the address block which boards use to give error responses.

2.7 Care and Handling of Diskettes

DO keep your diskettes in storage envelopes. This protects them from becoming damaged by dust, dirt, and other messy things.

DO allow the diskette and the computer system to come to an equilibrium temperature before use.

Do provide adequate backups of all your programs and data. Organized, frequent backups are essential to protect against hardware, software, or human failures.

DO use high quality diskettes to reduce disk drive wear and to reduce the likelihood of read/write errors.

DO NOT handle the disk surface or attempt to remove the diskette from its jacket. A fingerprint on the surface of a diskette may render it completely unusable.

DO NOT turn the computer on or off when a diskette is in a drive.

DO NOT store the diskettes near heat, sunlight, or a magnetic field.

Write Protect Feature

The drives used in the DSC/3 and DSC/4 are equipped with a sensor to detect individual diskette write-protect. Diskettes with this option have a notch removed from the end opposite the label. The notch must be covered in order to write on the diskette.

2.8 The PROM Monitor

When the computer is powered on or when the RESET switch is pressed, the computer will begin executing a 1K PROM monitor program. If one of the auto-boot jumper options have been selected (see section 2.3), the PROM will immediately attempt to bootup from a floppy, a harddisk, or the network. Otherwise, the message "PROM Monitor 1.xx" will be displayed, and the command prompt ":" will also be displayed. This indicates that the monitor program is ready to accept a command. Valid commands are:

<CR>	Dump memory at 9000H
S addr	Set memory locations beginning at addr.
D [addr1 [addr2]]	Dump memory locations from addr1 to addr2.
F addr1 addr1 byte	Fill memory locations from addr1 to addr2 with byte.
G addr	Jump to addr.
I port	Input a byte from a Z-80 port.
O port byte	Output a byte to a Z-80 port.
M register mapbyte	Set a memory map register (applicable to the DSC/4 only).
T addr1 addr2	Test memory locations from addr1 to addr2.
BF	Boot from a floppy on unit 0
BN	Boot from the network
BH	Boot from the harddisk
BT addr	Test the floppy controller - send the command string beginning at addr to the controller. The string should be terminated by 0FFH. The result string will be placed in memory beginning at 9380H.
B1	Boot from a floppy on unit 1

The 'M' command can be used to set a memory map register on the DSC/4. The register to be set (0,1,2,...F) is given by 'register'. The value to be placed in the memory map register is given by 'mapbyte'. The high order bit of mapbyte determines whether the selected page is mapped into local memory or external memory. The remaining 7 bits select an external memory page; however, the high-order 3 of the 7 bits are low-true. Thus, the command 'M2 F3' will map logical addresses 2000-2FFFH to external physical addresses 3000-3FFFH. To activate the memory map, the command 'I 0' must be executed.

The 'T' command can be used to test a range of memory locations. When used in conjunction with 'M' commands, the 'T' command can be used to test the entire external memory on the DSC/4. However, on the DSC/3, it is only possible to test locations from 4000 to 93EF or 9400 to FFFF. Addresses from 0 to 3FFF are not accessible while the PROM is active; addresses from 93F0 to 93FF contain the stack for the PROM monitor. More exhaustive testing requires using the MEMTEST program.

3.1 CP/M Overview

CP/M stands for 'Control Program for Microcomputers'. It is an operating system which manages the resources of a microcomputer so that the computer is easy and convenient to use. CP/M is described in detail in the reference manuals in Appendix A. These manuals are:

An Introduction to CP/M Features and Facilities
CP/M Assembler (ASM) - User's Guide
CP/M Dynamic Debugging Tool (DDT) User's Guide
CP/M Context Editor (ED) User's Guide
CP/M System Alteration Guide
CP/M Interface Guide

CP/M resides on tracks 0 and 1 of a system diskette. Track 2 contains the disk directory. A double density diskette can contain up to 128 files; they are located on tracks 2 to 76. CP/M file names consist of two parts - a 'primary name' of up to 8 characters, and a 'secondary name' of up to 3 letters; the two parts of a name are separated by period. The secondary name usually denotes the type of file. Some file types are:

ASM	source file for ASM (the CP/M assembler)
HEX	object file produced by ASM
PRN	listing file produced by ASM
COM	binary core image - usually a command file
BAS	source file for the BASIC compiler
INT	object file produced by the BASIC compiler, used as source for the RUN package
SUB	source file for SUBMIT command

CP/M is divided into three major components:

(1) CCP (Console Command Processor)

This component handles communication with the user at the system console. The CCP prompts the user with:

x> (where 'x' is the currently active drive)

The user may then type a command, terminated by a carriage return. The CCP has the following built-in commands:

DIR [filemask]	list directory entries
ERA filemask	erase file(s)
REN newname=oldname	rename file
SAVE length filename	save contents of memory
TYPE filename	list an ASCII file

If the user types a name which is not a built-in command, the disk directory is searched for a COM file by that name. If found, the file is loaded into memory beginning at 100H, and the command is executed. The following commands are supplied as

part of DMS's standard CP/M package:

*PIP [command]	file copy utility
*STAT [filemask]	file status utility
*DDT [filename]	CP/M dynamic debugger
*ASM filename	CP/M assembler
*ED [filename]	CP/M editor
*SUBMIT filename	submit utility
*SYSGEN	system generation utility
LOAD filename	convert HEX to COM file
DUMP filename	display a binary file
SETTIME	set the time-of-day
TIME	display the time-of-day
ALLOC	list/modify the Disk Allocation Table
ASSIGN [diskname device]	assign device to A,B,C, or D
SETBAUD portname rate	set baud rate of a port
FDCOPY	disk copy utility
FORMAT	disk format utility
HARDCTRL	hard disk firmware loading program
HDFMAT3S	Shugart hard disk formatting program
HDFMAT3M	Memorex hard disk formatting program
H3OASINT	hard disk Oasis initialization program
READO [filename trk sectr num]	read from the hard disk to a file
WRUNO [filename trk sectr]	write a file onto the hard disk
DIRHARD	list out the hard disk partition names

(*) The commands preceded by an asterisk are described in 'Introduction to CP/M Features and Facilities' and other CP/M manuals. All other commands are described in sections 3.3 .

(2) BDOS - Basic Disk Operating System

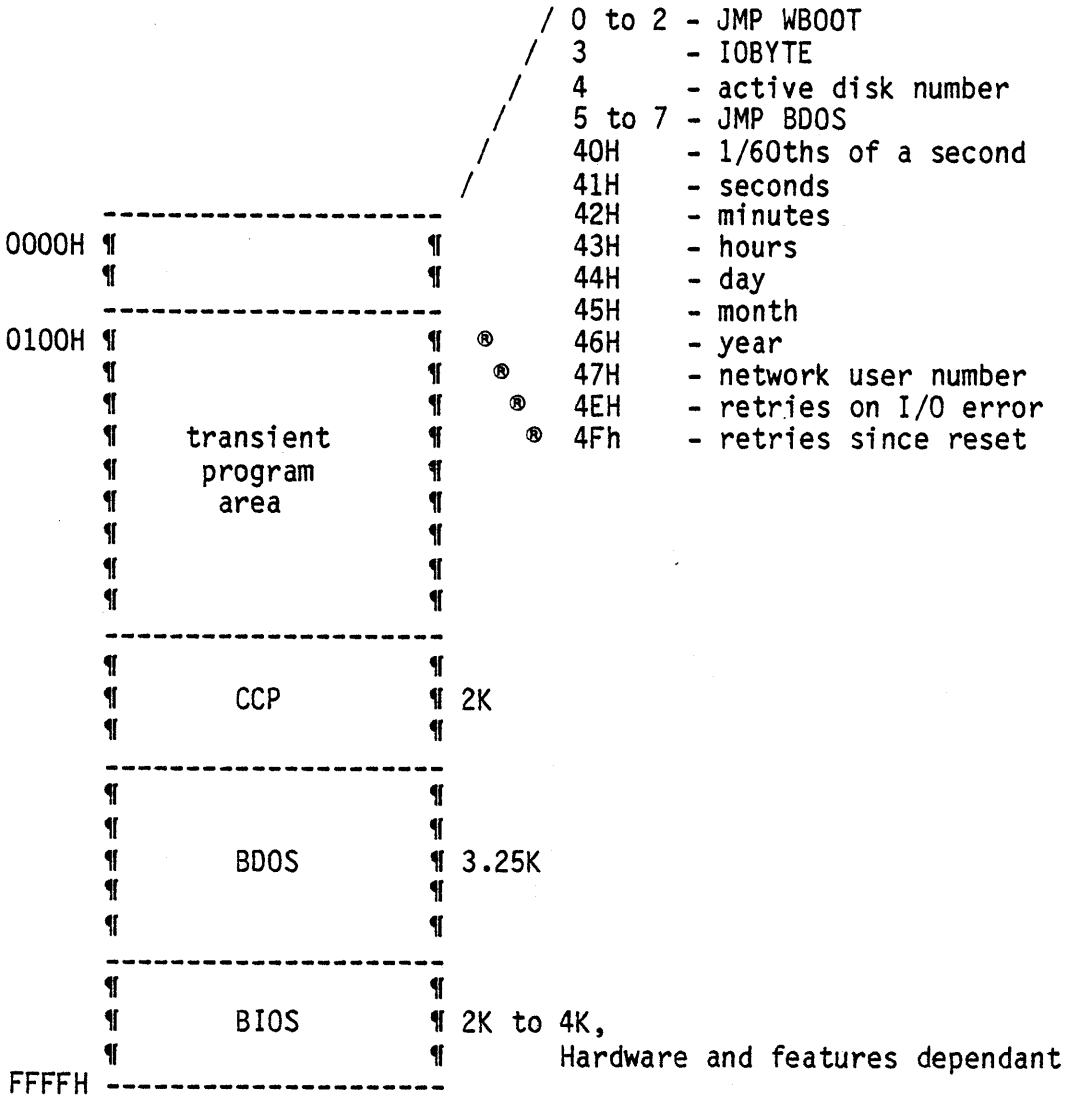
The BDOS provides the interface between the file definition and the physical location of the file on the disk. This information is kept in the directory and is rewritten every time the file is updated.

CP/M file allocation frees the user from specifying file size or actual disk addresses. CP/M allocates and de-allocates disk space as the need arises.

(3) BIOS - Basic I/O System

The BIOS provides the interface between the BDOS and the hardware. At the beginning of the BIOS are a series of jump vectors. These vectors are used by the BDOS to gain access to the BIOS. There are additional vectors which allow the user to gain access to some of the special features of Digital Microsystems' BIOS. The BIOS routines are described in section 3.4.

3.2 CP/M Memory Allocation



The total memory space (TPA) available to a user program varies from 54K to 56.5K, depending on the hardware configuration.

3.3 System Utility Programs

ALLOC

Initialize and/or Modify the Disk Allocation Table

Program: ALLOC (For hard disk systems only)

ALLOC allows the user to modify the Disk Allocation Table stored on Unit 0 of the hard disk. Any or all of the logical disk assign-ments can be set/reset to a size, name, optional password, and a control byte of the users choosing. See Section 3.6 for a comprehensive discussion of the Hard disk logical units.

ALLOC commands are as follows:

L - List the current allocation table, including current modifications, if any.

H - Print a command summary.

M - Modify an entry in the Disk Allocation Table.

S - Save the current Disk Allocation Table, with current modifications, as the new Disk Allocation Table.

Q - Abandon ALLOC program. Exit to system.

Z - Zero out (initialize) the current Disk Allocation Table.

It is very important to note that NO PERMANENT CHANGES IN THE DISK ALLOCATION TABLE ARE MADE UNTIL THE 'S' COMMAND IS USED.

An example follows on the next page.

Example:

```
A>ALLOC
ALLOC VER 1.xx

UNIT SIZE   NAME    PASSWORD CTRL
---- ---- -------- -------- ----
 00   01  SYSTEM             FF
 01   01  USER1A             FF
 02   02  USER1B             FF
 03   03  USER1C             FF
 04   04  USER1D             FF
 05   01  USER2A             FF
 06   02  USER2B             FF
 07   03  USER2C             FF
 08   04  USER2D             FF

Command?M                  <<---This command is for modifying the table.

What unit number?2
What size?2
What name?PRIVATE
What password?PISCES
What control?FF

UNIT SIZE   NAME    PASSWORD CTRL
---- ---- -------- -------- ----
 02   01  PRIVATE  PISCES    FF

Is data correct?Y

Command?L                  <<---This command lists out the modified table.

UNIT SIZE   NAME    PASSWORD CTRL
---- ---- -------- -------- ----
 00   01  SYSTEM             FF
 01   01  USER1A             FF
 02   02  PRIVATE   PISCES    FF
 03   03  USER1C             FF
 04   04  USER1D             FF
 01   01  USER2A             FF
 02   02  USER2B             FF
 03   03  USER2C             FF
 04   04  USER2D             FF

Command?S                  <<---This command saves the new table.
Function Completed
Command?Q                  <<---This command exits the program.

A>
```


ASSIGN

Assign a CP/M disk to a storage device

Program: ASSIGN [diskname devicename]

Arguments:

diskname	CP/M disk name A, B, C, or D
devicename	S0, S1, S2, ... S7 for single density floppy
	D0, D1, D2, ... D7 for double density floppy
	H1, H2, H3, ... , H63 for a hard disk unit
	Hard disk or HiNet partition name (8 chrs)

CP/M drives A, B, C, and D are initially assigned to default physical devices. The ASSIGN command can be used to change the disk assignments. The default assignments for a normal floppy disk system are:

A	assigned to D0 (the right hand drive, double dens)
B	assigned to D1 (the left hand drive, double dens)
C	assigned to S1 (the left hand drive, single dens)
D	assigned to D1 (i.e., the same as B)

If no arguments are specified, then the current disk assignments will be displayed.

For double-sided floppies, drives 4 to 7 refer to the opposite sides of drives 0 to 3. Thus, on a normal dual disk system, drives 0 and 4 refer to the two sides of the right hand drive, while drives 1 and 5 refer to the two sides of the left hand drive.

Some examples follow on the next page.

Examples:

```
A>ASSIGN D SO              assign  D  to the right-hand  drive
 Assignment Accepted.      for a single density diskette.

A>ASSIGN        # does not follow

  CURRENT DISK ASSIGNMENTS
  ------------------------
  A - double density, unit 00       size: 482K bytes
  B - double density, unit 01       size: 482K bytes
  C - single density, unit 01       size: 241K bytes
  D - double density, unit 01       size: 482K bytes

A>ASSIGN B S7                ( Assignment to a local drive )
 Assignment accepted.

A>ASSIGN C USER1A     ( Assignment to a public disk partition )
 Assignment accepted.

A>ASSIGN D PRIVATE    ( Assignment to a private disk partition )

 ASSIGN VER 2.06
                                    A password is required.
   Enter Password:       <<--- Keyboard does not echo entry
 Assignment accepted.

A>ASSIGN

              Current Disk Assignments
              ------------------------

  A - double density, unit 00       size: 482K bytes
  B - single density, unit 07       size: 421K bytes
  C - network       , USER1A        size: 256K bytes
  D - network       , PRIVATE       size: 512K bytes
```


FDCOPY

Copy a diskette

Program: FDCOPY

Copy a double density single sided diskette from one drive to another. FDCOPY can copy an entire diskette, just the system tracks (0 and 1), or just the data tracks (2 to 76). FDCOPY reads up to 8 tracks from the source disk, and then copies them to the destination disk. This procedure is repeated until the copy operation is completed.

A noteworthy feature of FDCOPY is the Control-B option. If at any time during the data-entering portion of the program the user wishes to go back to the previous step, type a Control-B. This will immediately place you back to the previous step.

Example:

```
A>FDCOPY

Floppy Copy Program 1.xx

Source disk (0-7) :0
Destination disk:  (0-7) :0          <<--- Extraneous entry made
COPY: System tracks, Data tracks, or
      All  tracks?  (S,  D,  or A)?  ©B  <<--- ©B entered
Destination disk:  (0-7) :1          <<--- Back to previous question
COPY: System tracks, Data tracks, or
      All  tracks?  (S,  D,  or A)?  A
Hit return when ready <cr>        <<--- Carriage return entered
 Track xx copied

**JOB COMPLETED**

Source disk (0-7) :             <<--- Program will loop indefinitely
                                      or until a Control-C is used.
A>
```


Format a diskette

Program: FORMAT

Format a floppy diskette on drive 0, 1, 2, or 3. In a normal two-drive system, the right hand side is drive 0, while the left hand drive is drive 1. For double-sided drives, the opposite side of drive 0 is referred to as drive 4, while the opposite side of drive 1 is called drive 5. Program ASSIGN may be used to check the current drive assignments. The diskette is formatted in IBM 3740 single density format or IBM System 34 double density format. There are 77 tracks, 26 sectors per track, and either 128 or 256 bytes per sector. Track 0 is always formatted in single density.

In single density, sectors are written in numerically increasing order. Thus, on each track, sector n is immediately followed by sector n+1. However, CP/M uses a logical to physical sector map so that when sectors are requested in numerical order, they will actually be interleaved. This is done to minimize disk access time. In double density, the sectors are interleaved on the diskette itself, and CP/M's map is not used.

Double density diskettes formatted on the DSC/2 cannot be read by the DSC/3. If an attempt is made to read a DSC/2 double density diskette, a DENS error will result.

Example:

```
A>FORMAT

Floppy Disk Format Program 1.xx

Enter Disk Number (0-7) :3
Single or Double density? D
Type return to start.

FORMAT COMPLETE.

Enter Disk Number (0-7) :     <<--- Program will loop indefinitely
                                    or until a Control-C is entered.
```


Load Hard Disk Controller Firmware
(For hard disk systems only)

Program: HARDCTRL

Digital Microsystems HDC 1.x hard disk controllers are intelligent sub systems containing a Z-80, 1K ROM and 32K RAM. The ROM code implements basic Read, Write, and Seek commands on physical (1K byte) sectors. On reset, the ROM code will read Track 0, Head 0, sectors 1 and 2 (2 Kbytes) into RAM. This additional code implements functions such as error corrections and recovery, data buffering, logical disk partitions, logical sector sizes, and more complex commands that are available using just the ROM code.

After a disk format (see HDFMAT3x) this code must be replaced by running HARDCTRL. HARDCTRL contains as data the firmware to be loaded. HARDCTRL will inquire of the user the type and size of disk installed and incorporate this information with the firmware, and then will issue commands to the controller ROM code to write to Track 0, Head 0, sectors 1 and 2.

Example:

Note: Caps Lock key must be set.

```
A>HARDCTRL
DMS HARD DISK TEST PROGRAM VER DSC3M 2.2A
LOADING HARD DISK CONTROLLER PROGRAM AREA
WHAT IS HARD DISK SIZE?
11 MBYTE MEMOREX = 1
14 MBYTE = 3, WITH FIXED HEADS = B
28 MBYTE = 7, WITH FIXED HEADS = F
ENTER SIZE (1,3,7, OR F) - F
CONTROLLER FIRMWARE LOADED SUCCESSFULLY
A>
```


HDFMAT3M
HDFMAT3S

Format the hard disk
(For hard disk systems only)

Program: HDFMAT3S
HDFMAT3M

HDFMAT3S is used with any DSC 3/4 system to format a Shugart 400x fourteen-inch hard disk. HDFMAT3M is an identical program for the Memorex 101 eight-inch drive.

The physical act of formatting each of these drives varies in the number of tracks and heads, although each uses 1024 byte sectors. The program will inquire of the user several parameters to enable it to format the disk correctly.

Note that formatting a disk completely destroys all data and should obviously be preceeded by a backup. Formatting writes headers for each sector, and places 0E5h in all data bytes on the drive. This serves to 'erase' a drive to a known initial state. Note that with all of DMS's hard disk products some space is reserved on track 0 for system information such as controller firmware, defective sector table, operating systems, and disk parttioning information. All of this information should be reloaded immediately after a format. See HARDCTRL, ALLOC, and WRUNO for additional information. One option when running this program is to avoid formatting Track0, head 0 (Shugart) or all of Track 0 (Memorex) which will not disturb the controller firmware, partition table, or Bad Sector Table (which is unique to every drive).

Example:

Note: Set the CAPS lock key

```
A>HDFMAT3M
DMS HARD DISK TEST PROGRAM VER DSC3S 2.26

HARD DISK FORMAT ROUTINE
THIS ROUTINE DESTROYES DATA ON THE DISK.
DO YOU WISH TO CONTINUE (Y,N) Y
HOW MANY MOVABLE HEADS ON THIS DISK (4,8) 4
FIXED HEADS? (Y,N) Y
WHICH TRK WOULD YOU LIKE TO START ON? (0,1) 0
BUSY FORMATTING
DONE
A>
```


H3OASINT

Initalize hard disk for Oasis operating system
(For hard disk systems only)

Program: H3OASINT

Example:

Note: Caps Lock key must be set.

```
A>H3OASINT
LOADING HARD DISK USER PROGRAM AREA VERSION C-7-24-79
WHAT IS HARD DISK SIZE?
14 MBYTE = 3, WITH FIXED HEADS = B
28 MYTE = 7,  WITH FIXED HEADS = F
ENTER SIZE (3,7,B, OR F) - F
02 00 00 01 00 00 01 00 01 00 00 01 00 00 01 00
A>
```


SETBAUD

Change baud rate

Program: SETBAUD portnumber baudrate

Arguments:

portnumber	0, 1, 2, or 3
baudrate	110, 300, 600, 1200, 1800, 2400, 4800, or 9600

Change the baud rate of a selected port. The only restriction is that ports 2 and 3 must always share the same baud rate. Normally, each port is initialized to 9600 baud.

Example:

SETBAUD 3 300	Set the baud rate of port 3 to 300. Note that the baud rate of port 2 will also be set to 300.

SETTIME

Set date and time

Program: SETTIME

Set the current date and time. The date must be entered using the format MM/DD/YY. For example, 07/04/76 is a valid date. The time is entered using the format HH:MM. Thus, 17:08 indicates 5:08 PM. The date and time are maintained by the BIOS in low core at locations 40H to 46H. These locations can be examined by the user at any time.

Example:

```
A>SETTIME

Enter date  MM/DD/YY - 07/04/76
Enter time  HH:MM - 17:08

A>
```


TIME

Display date and time

Program: TIME

Display the current date and time. If the date and time have not yet been set, the date will be displayed as 'XXX-00-00' and the time will indicate the elapsed time since the system was powered-on or reset.

Example:

```
A>TIME

Date  Jul-04-76
Time  17:08:23
```

READO

Read from unit 0 of the hard disk to a CP/M file
(For hard disk systems only)

Program: READO [filename trk sectr numsecs]

This program reads data contained from unit 0 of the hard disk into any valid CP/M file. (It does not have to exist already.) This is accomplished by first obtaining a track and sector number on unit 0 from the user. Next, the number of sectors to read is obtained. Now the information is read into memory. Upon a successful read from the hard disk, READO requests a CP/M file name. The file is opened, all the sectors read from the hard disk are written to the file, and the file is closed. All numbers used in READO are hexadecimal. As shown above, READO can accept all the information from the command line. If this option is used, any incorrect data will reset the program back to otaining its data from the console, otherwise, no console input should be required.

READO is designed for such tasks as backing up Bad Sector Tables, backing up the Disk Allocation Table. Control-C aborts the program at any time.

Example:

```
A>READO

READO VER 1.00
Hard disk to Floppy disk CP/M filewrite utility program.

Starting track  number (in hex)? 1
Starting sector number (in hex)? 1
Number of sectors to read (in hex)? 8

  TRACK  SECT  TOTAL SECTORS
  -----  ----      -----
    01    01        08

Is data correct?Y

Data Read from Hard Disk

Enter file name - BADSECT.TBL

File opened.
File closed.


Data written to Floppy disk.

A>
```


WRUNO

Write a CP/M file to unit 0 of the hard disk
(For hard disk systems only)

Program: WRUNO [filename trk sectr]

This program writes data contained in any valid CP/M file to unit 0 of the hard disk. This is accomplished by first obtaining the file name from the user, and reading the file into program memory, sector by sector. Next, a track and sector number on unit 0 is obtained from the user. This will point to the location on the disk where the file contents are to be placed. All numbers used in WRUNO are hexadecimal. As shown above, WRUNO can accept all the information from the command line. If this option is used, any incorrect data will reset the program back to otaining its data from the console, otherwise, no console input should be required.

WRUNO is designed for such tasks as placing Bad Sector Tables, loading the Disk Allocation Table, or loading CP/M (either Master or Station versions), onto unit 0 of the hard disk. Control-C aborts the program at any time.

Example:

```
A>WRUNO

WRUNO VER 1.00
CP/M Floppy disk file to hard disk write utility program.

Enter file name - BADSECT.TBL
File found
File in memory.  Sectors read: 08

Starting track  number (in hex)? 1
Starting sector number (in hex)? 1

  TRACK  SECT  TOTAL SECTORS
  -----  ----      -----
    01    01        08

Is data correct and ready to be written to the hard disk?Y

Data Written to Hard Disk

A>
```

DIRHARD

Print a directory of the hard disk partitions
(For hard disk systems only)

Program: DIRHARD

This program lists out the hard disk partition units as stored in the allocation table. If the user wishes to assign himself to any one of the hard disk partitions, DIRHARD will list out all the partition names (excluding the name of unit 0). The names listed will be the ones required for partition assignments. No passords can be listed out by DIRHARD. These must be obtained from the Disk Allocation Table itself.

Example:

```
A>DIRHARD
 DIRHARD VER 1.xx

              Current Hard Disk Partitions
              ----------------------------

USER1A   256K bytes      PRIVATE   512K bytes     USER1C    1M byte
USER1D     2M bytes      USER2A    256K bytes     USER2B  512K bytes
USER2C     1M bytes      USER2D      2M bytes

A>
```


3.4 BIOS routines

The BIOS is the portion of the operating system which interacts directly with the hardware. At the beginning of the BIOS are a series of jump instructions which are used by the BDOS to gain access to the BIOS routines. The jump instruction at locations 0 to 2 can be used to locate the beginning of the BIOS. Each of the standard BIOS routines is described below.

Offset	Name	Description
00H	CBOOT	Initialize ports 1, 2, and 3, the real-time clock, and front-panel interrupts. Then jump to WBOOT.
03H	WBOOT	Load the CCP and the BDOS, and execute the CCP.
06H	CONST	Check console status
09H	CONIN	Console input
0CH	CONOUT	Console output
0FH	LIST	List output
12H	PUNCH	Punch output
15H	READER	Reader input
18H	HOME	Move the head of the current disk to track 0. This routine should be called once before a new diskette is read or written.
1BH	SELDSK	Select a disk drive. The disk parameter table for the current disk is written over the standard table in the BDOS.
1EH	SETTRK	Select a disk track. The seek operation is delayed until READ or WRITE is called.
21H	SETSEC	Select a disk sector.
24H	SETDMA	Set the DMA address.
27H	READ	Read a sector from a disk.
2AH	WRITE	Write a sector to a disk.

The READ and WRITE routines perform I/O on the currently selected disk, track, and sector. Normally, CP/M performs I/O in blocks of 128 bytes. Larger blocks may be read or written by using the SETBYT routine (see below).

Single density floppy disk I/O

If necessary, a seek to the proper track is performed. The DMA chip is programmed to read/write 128 bytes. The controller is told to perform the desired operation. When the DMA chip has completed the data transfer, it interrupts the CPU. If any errors are encountered, the IOERR routine is called.

Double density floppy disk I/O

Double density I/O is complicated by the fact that CP/M thinks that sectors contain 128 bytes, but double density sectors actually contain 256 bytes. Thus, it is necessary to have one 256 byte read buffer, and one 256 byte write buffer. These buffers begin at locations FE00H and FF00H. For a READ, the buffer is first consulted to see if the desired sector is already in memory. If so, the data can be immediately transferred. Disk writes are handled similarly. However, the last sector written may remain in the write buffer until the buffer is flushed. The buffer is flushed whenever a warm-boot is performed, and also when no disk accesses have occured for more than one second.

User BIOS calls

The routines in the BIOS may be called by a user program. However, a routine should never be called directly. Instead, the user should call the location in the jump table at the beginning of the BIOS. The order of the jump table is never changed, although the size of the routines in the BIOS may change from time to time. The address of the jump table is stored in locations 1-2. For example, to call the CONOUT routine, use:

```
LHLD  1        ; HL = base of vectors + 3
LXI   D,0CH-3  ; offset of JMP CONOUT
DAD   D        ; compute address of CONOUT
PCHL           ; jump to CONOUT
```

Non-standard BIOS routines

In addition to the standard BIOS routines described above, there are several special BIOS routines which the user may access directly. These are available only in Digital Microsystems' version of CP/M.

Offset	Name	Description
60H	IOERR	Process an I/O error - print an error message, and wait for the user to type RETURN or CTRL-C. If CTRL-C is typed, perform a warm-boot, otherwise return to the caller.
63H	CPMMAP	A pointer to the current disk map is returned in

HL, and the current map byte is returned in A. The upper 2 bits of the map byte indicate whether the current disk is single density (00), double density (01), a hard disk (10), or a network disk (11). The lower 6 bits are the disk unit number. The ASSIGN program uses this information.

66H SELMEM Select a memory page. This routine is applicable only to the DSC/4. The memory map register number (0 to F) is given by the upper 4 bits of B. The contents of the register is given by C. The memory map is discussed in section 4.2.

69H SETBYT Normally, disk READs and WRITEs are performed on 128 byte blocks. However, by calling SETBYT, the user can perform I/O on larger sized blocks. The block length is given by the contents of HL. Before the BDOS can be used again, SETBYT must be re-called, with HL = 80 (this is the normal CP/M block size). Otherwise, a ENDT error is likely to occur when the BDOS is used.

3.5 Bootstrap Procedure

When the DSC/3 or DSC/4 is powered on or reset, the CPU immediately begins executing the PROM monitor. If an auto-boot option has been selected, then the CPU will immediately attempt to bootup from a device. The various bootstrap procedures are described in detail below:

Floppy Boot

Sectors 1 and 2 of track 0 are read into memory beginning at location 9000H. If this is successful, the CPU jumps to location 9000H. The following steps are then performed:

1. Pages 0 through F are mapped to physical addresses 0 through 0FFFFH. However, pages 0 and 9 are temporarily kept in local memory. This step is applicable to the DSC/4 only.

2. The BIOS is read from track 1, sectors 10-26 into the appropriate memory locations.

3. The remainder of the bootstrap loader is moved to page 4, and the CPU jumps to 4000H.

4. Pages 0 and 9 are now mapped to external memory, and the memory map is enabled on the DSC/4. Also, the PROM is disabled on the DSC/3.

5. The cold boot portion of the BIOS is executed. I/O ports 1, 2, and 3 are initialized, the real-time clock is enabled, and the remainder of CP/M is read into memory and executed.

3.6 Using a Hard Disk

The Winchester hard disks currently available with the DSC 3/4 are the Memorex 101 eight inch with 10.6 Megabytes and the Shugart 4004/8 fourteen inch with 13.7/27.4 Megabytes formatted.

The absolute file addressing limit for CP/M 1.4 is 4 Megabytes and for CP/M 2.2 is 8 Megabytes. Therefore the hard disk must be partitioned into smaller logical units for use with CP/M. Operations that cause frequent directory searches will run more efficiently in small partitions as the number of CP/M sectors read and processed for each directory search will be smaller than for large partitions.

Physically, each of the hard disks has a different number of heads (4 or 8), tracks (202 or 242) and sectors per track (11 or 17). Each physical sector is 1024 bytes. CP/M works with logical sectors of 128 bytes, up to 128 sectors per track and up to 256 tracks. The firmware executing in the hard disk controller treats each disk as a linear array of 1024 byte blocks. Blocks are numbered from 0 on up to the maximum available. When a CP/M request for unit, track, and sector (128 bytes) is received, it is mapped onto this array. The physical sector is read, and the appropriate 128 bytes are extracted and returned to CP/M. For simplicity, CP/M is told that all hard disk logical units have 128 sectors per track and as many tracks as appropriate for that logical unit's size.

Each of the hard disks available can be divided into at most 64 partitions (units). Each partition must be one of the following sizes:

1 - 256K bytes
2 - 512K bytes
3 - 1024K bytes
4 - 2048K bytes
5 - 4096K bytes
6 - 8192K bytes (CP/M 2.2 only)

The sum of the sizes of all the partitions must obviously be less than the total disk space available. After making allowance for spare sectors for defective sector mapping, total available disk space is as follows:

Memorex 101	10,560K bytes
Shugart 4004	13,600K bytes
Shugart 4008	27,200K bytes

The first disk partition (numbered 0) is reserved for use by the operating system and controller. It is not a CP/M structured disk and contains at specifically assigned locations information such as:

Disk controller firmware
Disk allocation table
Disk error statistics
CP/M and BIOS for Network Master
CP/M and BIOS for Network Stations
CP/M and BIOS for single CPU system

The size of this partition should be 1 (256K bytes). Because this partition does not have a CP/M directory structure, information that must be put on it during system installation (or after a hard disk format) must be done so directly. A program is provided to transfer CP/M files (usually from a floppy) to the system partition and is called WRUNO (WRite UNit 0). WRUNO assumes unit 0 to contain 16 tracks with 128 sectors, each of 128 bytes. This maps to blocks 0 through 255 of any hard disk. WRUNO must be told explicitly which logical track and sector to write to.

Before any high level software can communicate with the hard disk (this includes WRUNO), the firmware must be loaded onto blocks 0 and 1. This is accomplished with a program called HARDCTRL. When executed, HARDCTRL will ask the user for the size of the hard disk so the firmware will know the physical parameters to use.

In order to inform the controller firmware of the location and size of all logical units, and to inform CP/M of their size, a disk allocation table is maintained on unit 0 (Block 15). This table contains one entry for each of 64 units, with the following information:

(Position in table implies unit number, 0 first)

1 byte - size	0 = not assigned
8 bytes - name	alphanumeric characters
6 bytes - password	password (optional)
1 byte - control information	(read only, public, etc)

The entries in this table directly control the size and physical location of the logical units and should be changed with care. Each logical unit physically starts after the number of blocks required for all lower numbered units. Changing the size of any unit will displace all higher numbered units and therefore should only be done after a backup and format.

The construction or modification of the disk partition table is accomplished by a program named ALLOC. For demonstration purposes a default partition table is available in a file named ALLOC.TAB and may be directly written onto the hard disk using the WRUNO utility, as shown here: A>WRUNO ALLOC.TAB 0 79

The ASSIGN command can be used to inform CP/M of the user's desired assignment of logical partition to CP/M disk name (A:, B:, C:, or D:). See Section 3.3 for a description of ASSIGN.

3.7 Altering CP/M

For alterations to CP/M, consult the Digital Research CP/M alteration guide. Several options are provided as part of the standard BIOS.

1. The default baud rates on ports 1,2, and 3 can be changed by altering locations FFF3h to FFF6h in the BIOS. Normally, these locations contain the CTC chip command strings 45h,02h and 45h,0Dh. The first 2 bytes refer to port 1, while the second 2 bytes refer to ports 2 and 3. See sections 4.3 for a description of these CTC command strings.

2. The default IOBYTE is 54h. It is in location FFF7h. It can be changed to reflect any of the IOBYTE combinations described in section 3.8 .

3. The default type-ahead buffer is in locations F980h to F99Fh. Normally, the buffer is initially empty. However, it can be initialized if it is necessary to execute a sequence of commands when a cold-boot is executed. Type-ahead characters should be placed in the buffer beginning at F980h, and the number of characters in the type-ahead buffer should be put in location F97Eh.

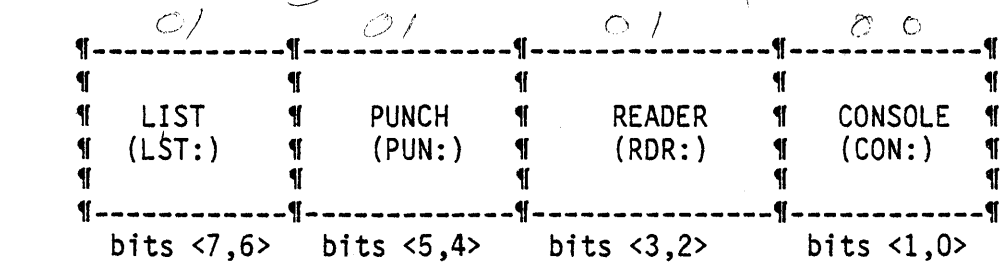
For example, follow the procedure shown below to generate a diskette which sets the default baud rate on ports 2 and 3 to 300 baud.

(Put a system diskette in drive A, and a freshly formatted diskette in drive B.)

```
A>SYSGEN
( Use SYSGEN to copy the operating system from drive A to memory. )

A>SAVE 45 CPM.COM
A>DDT CPM.COM
DDT VERS 1.4
NEXT  PC
2E00 0100
-S2DF5
2DF5 45 05 <cr>  ( SYSGEN adresses 2DF5-2DF6 correspond
                   to BIOS addresses FFF5-FFF6. )
2DF6 0D 34 <cr>
2DF7 ©C

A>SYSGEN

( Use SYSGEN to copy the operating system from memory to drive B. )
```

3.8 Implementation of the IOBYTE

The IOBYTE, at location 03h in memory, is implemented for the mapping of logical to physical devices. The mapping is performed by splitting the IOBYTE into four distinct fields called the CONSOLE, READER, PUNCH, and LIST fields, as shown below.



Whenever a reference to a logical device is made for I/O, the BIOS first looks at the appropriate bit field in the IOBYTE, and then uses one of the four physical devices service routines. The IOBYTE can be either examined or modified via the STAT command.

Examples:

```
STAT DEV:          (list the current IOBYTE status)
STAT LST:=LPT:     (this assigns the LIST to LPT:
                    i.e., the parallel port.)
```

The following table show the physical devices associated with each logical device. The default assignments are also shown.

Logical device name	IOBYTE value	Physical device name	DSC 3/4 device name
CON:	00	TTY:	port 0 (default)
	01	CRT:	port 2
	10	BAT:	port 1
	11	UC1:	port 3
RDR:	00	TTY:	port 0
	01	PTR:	port 3 (default)
	10	UR1:	port 1
	11	UR2:	port 2
PUN:	00	TTY:	port 0
	01	PTP:	port 3 (default)
	10	UP1:	port 1
	11	UP2:	port 2
LST:	00	TTY:	port 0
	01	CRT:	port 2 (default)
	10	LPT:	parallel port 2
	11	UL1:	network

3.9 Standard Diskette Contents

*PIP	[command]	file copy utility
*STAT	[filemask]	file status utility
*DDT	[filename]	CP/M dynamic debugger
*ASM	filename	CP/M assembler
*ED	[filename]	CP/M editor
*SUBMIT	[filename]	submit utility
*SYSGEN		system generation utility
LOAD	filename	convert HEX to COM file
DUMP	filename	display a binary file
SETTIME		set the time-of-day
TIME		display the time-of-day
ALLOC		list/modify the Disk Allocation Table
ASSIGN	diskname device	assign device to A,B,C, or D
SETBAUD	portname rate	set baud rate of a port
FDCOPY		disk copy utility
FORMAT		disk format utility
DIRHARD		list the hard disk partitions
READO	[filename trk sec numsec]	read from the hard disk to a file
WRUNO	[filename trk sec]	write a file onto the hard disk
PROM.ASM		listing of the 1K x 8 prom program
HARDCTRL		hard disk controller load program
H30ASINT		hard disk OASIS initialization program
HDTEST3S		hard disk test program for Shugart drives
HDTEST3M		hard disk test program for Memorex drives
USERS		maintain USER + CONFIGURATION TABLES

(*) The commands preceded by an asterisk are described in 'Introduction to CP/M Features and Facilities' and other manuals. All other commands are described in sections 3.3 . Names in brackets following the file name are optional entries. This means that successful execution of the program can be made with no command line.

4.1 Hardware Overview

A block diagram showing the major components of the DSC/3 and DSC/4 is supplied on the following page. The major support chips used are the Zilog DMA, SIO, PIO, and CTC chips, and the NEC765 floppy controller chip.

DMA — Performs high speed data transfer between an I/O device and memory.

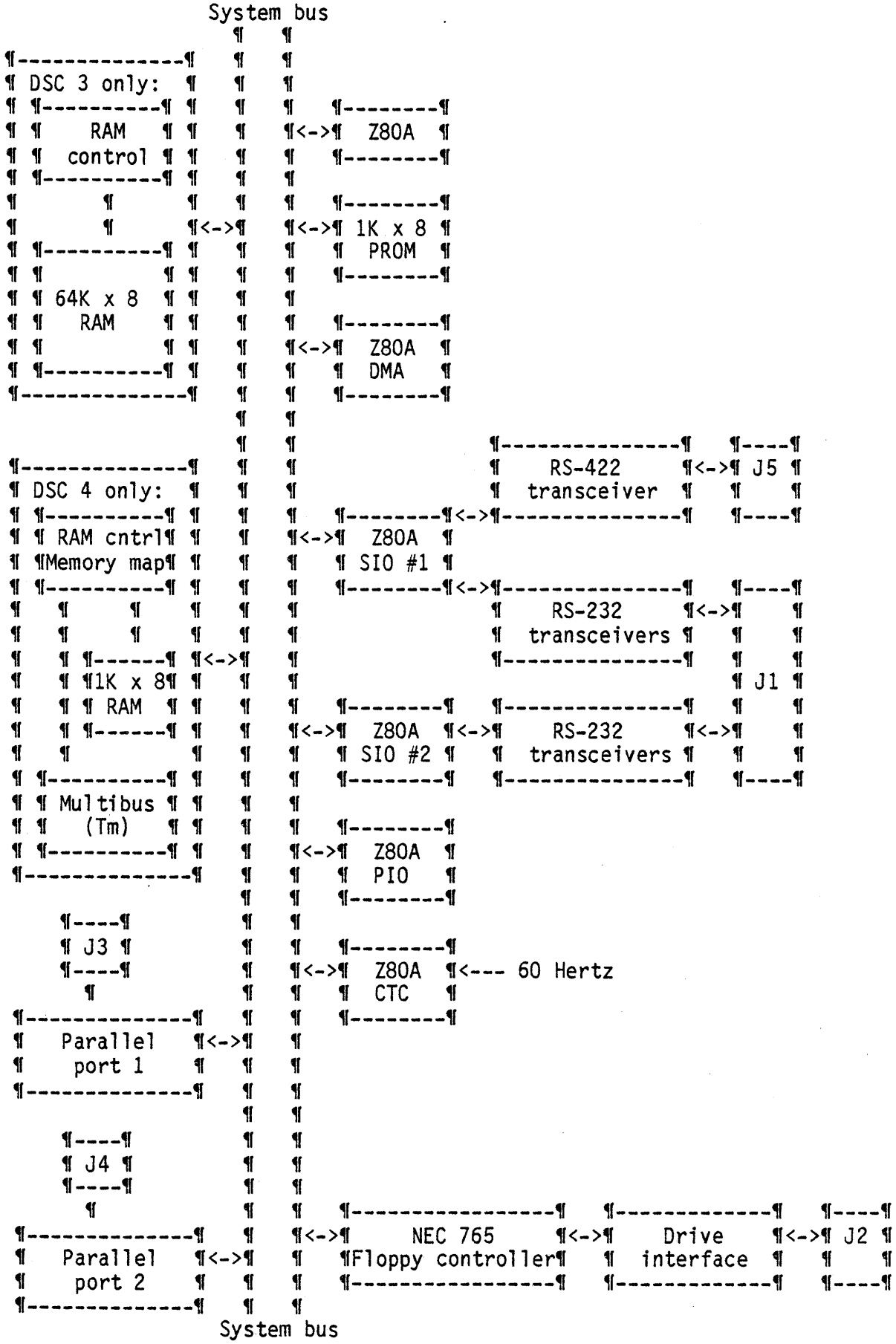
SIO — Each SIO chip controls up to 2 RS-232 ports. Also, channel B of SIO #1 can control an RS-422 port.

CTC — Generates baud rates for the I/O ports, and generates real-time interrupts at line frequency.

PIO — Provides control signals for I/O devices.

NEC765 — Controls up to 4 floppy disks.

HARDWARE BLOCK DIAGRAM



4.2 Memory Management

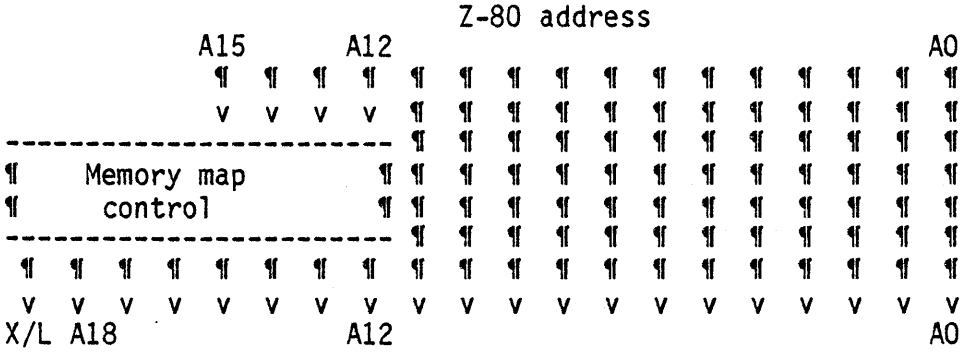
DSC/3

The DSC/3 has a 1K PROM and 64 Kbytes of dynamic RAM. When the computer is powered on or reset, the 1K PROM and the upper 48K of RAM are accessible. To 'replace' the 1K PROM with the lower 16K of RAM, an 'IN 2' instruction must be executed. Normally, this is done as part of the bootstrap procedure and is totally transparent to the user. Once the PROM has been 'mapped out', it cannot be re-enabled until the machine is powered off or the reset switch is pressed.

DSC/4

The DSC/4 has a 1K PROM and 1K static RAM on the CPU board. Each additional memory board has 128 Kbytes of RAM. When the computer is powered on or reset, only the 1K PROM and the 1K RAM are accessible. All 'even' memory pages (0xxx, 2xxx, 4xxx, ...) map to the PROM, while all 'odd' pages (1xxx, 3xxx, 5xxx, ...) map to external memory.

Since the Z-80 allows only 16-bit addresses, a mapping scheme is required to access more than 64 Kbytes. The DSC/4 has 16 8-bit memory map registers. The upper 4 bits of each 16-bit Z-80 address select one of the 16 memory map registers. This scheme is illustrated below:



Physical address

The X/L bit determines whether the physical address is in 'local' or 'external' memory. To set a memory map register, the Z-80 OUTP instruction must be used. This instruction outputs a byte to the port given by register C. The upper 4 bits of the B register select the map register to be set. For example, B = 60H selects register 6. The C register should equal 3. The output byte is written into the selected register.

Each memory map register contains 8 bits. The high bit (i.e., X/L) equals 0 to select local memory, and 1 to select external memory. The next 7 bits are placed on the address bus. However, the top 3 of the 7 bits are high-true. Thus, to map Z-80 addresses 3000-3FFFH to physical addresses 2000-2FFFH, use:

```
     LXI  B,2003H    ; B = register number x 10H, <= port 3
     MVI  A,0F3H     ; A selects physical addresses 2000-2FFFH
     OUTP A
```

To set all the memory map registers at once, use the following Z-80 code:

```
     LXI  H,MEMMAP   ; point to base of map
     LXI  B,0003H
LOOP:OUTI            ; put B-1 on address bus
     MVI  A,11H
     ADD  B
     MOV  B,A        ; load next map reg no. into B
     JNZ  LOOP
     .
     .
     .
MAP: .BYTE 0FFH,  00H, 0F1H, 0F2H ; page F gets set first
     .BYTE 0F3H, 0F4H, 0F5H, 0F6H
     .BYTE 0F7H, 0F8H, 0F9H, 0FAH
     .BYTE 0FBH, 0FCH, 0FDH, 0FEH
```

To activate the memory map, execute an 'IN 0' instruction. After the map has been activated, the local memory can still be accessed by using the X/L bit.

4.3 Adjusting baud rates

In normal operation, each of the 4 serial I/O ports is initialized as follows:

Port 0 - RS-232, 31-character type-ahead, 9600 baud

Port 1 - RS-422, 500 Khz if the network option is selected
RS-232, 9600 baud otherwise

Port 2 - RS-232, 9600 baud

Port 3 - RS-232, 9600 baud

Port 0 is initialized in the PROM monitor, while ports 1, 2, and 3 are initialized in the cold-boot portion of the BIOS. Each port can run at a different baud rate, except ports 2 and 3, which must run at the same baud rate. To change the baud rate of a serial I/O port, it is necessary to reprogram the appropriate channel of the CTC chip. This is done by outputting a string of bytes to a Z-80 port corresponding to the serial I/O port.

Serial I/O port	Z-80 port	Baud rate	Command bytes
0	30H	9600	45H,13
1	31H	4800	45H,26
2	32H	2400	45H,52
3	32H	1800	45H,69
		1200	45H,104
		600	45H,208
		300	05H,52
		110	05H,142

For example, to set port 1 to 300 baud, execute the following Z-80 instructions:

```
MVI  A,05H
OUT  31H
MVI  A,52
OUT  31H
```


4.4 Serial I/O ports

The DSC/3 and DSC/4 have 3 RS-232 ports, and a fourth port which can be jumper-selected to be either RS-232 or RS-422 (see section 2.3 for jumper connections). There are 2 SIO chips connected to the internal bus and to edge connectors via EIA RS232 signal receivers and drivers.

Each SIO chip controls two ports. SIO #1 controls ports 0 and 1; SIO #2 controls ports 2 and 3. Port 1 can be either RS-232 or RS-422. Port 0 is usually attached to the main system console. Port 2 is usually attached to a printer, while port 3 is available for another CRT, printer, or modem. Each port can be programmed separately by using the following Z-80 port numbers:

Serial I/O port	Z-80 data port	Z-80 command port	Standard Use
0	28H	2AH	console
1	29H	2BH	network
2	20H	22H	list
3	21H	23H	reader/punch

The SIO chips are described in detail in the Appendix. For example, the following code is used in the PROM monitor to setup port 0 as an RS-232 port:

```
     LXI  H,RS232
     LXI  B,092AH
     OUTIR
     .
     .
RS232:
     .BYTE  18H              ; channel reset
     .BYTE  14H,01001100B    ; 2 stop bits, no parity
     .BYTE  03H,11000001B    ; receiver enable
     .BYTE  05H,11101010B    ; transmitter enable
     .BYTE  11H,00000000B    ; no interrupts
```

To read from port 0, the following code is then used:

```
RWAIT:
     IN   2AH        ; get port status
     BIT  0,A        ; wait until receiver is ready
     JRZ  RWAIT
     IN   28H        ; get character from port
```

To write to the port, the following code is used:

```
WWAIT:
     IN   2AH        ; get port status
     BIT  2,A        ; wait until transmitter is ready
     JRZ  WWAIT
     MOV  A,C        ; move output char to accumulator
     OUT  28H        ; output the character to the port
```

4.5 System clock and timer

All timing on the DSC3/4 is drived from one of two sources: a 16 MHz crystal controlled oscillator and a line frequency synchronized square wave.

The 16MHz oscillator is a standard crystal controlled circuit using two 74S04 inverters biased into their active region. This oscillator output is bufferred and divided by successive powers of two to develop all binary frequencies from 16 MHz to 500 KHz.

The 4 MHz clock to allthe Z80 family devices is developed using discrete transistors to provide the rapid rise and fall times, and levels sufficiently close to the supply voltages (0V and +5V) required by Z80 specifications. Particularly important is pulse width low, and high (110 nsec) and high level (Vcc -

Serial baud rates are derived using a controller/timer chip dividing a 2 MHz input by a software selected division.

A real time clock is implemented from a line frequency signal developed in the power supply from the mains voltage. This signal is squared using a Schmidt trigger (7414) and is fed to the counter timer chip to provide interrupts at a 60 (or so) Hz rate.

4.6 The Floppy Controller

The NEC765 floppy controller chip is used to control up to 4 floppy disk drives. The Z-80 communicates with the controller via its status port (10H) and data port (11H). The status port is used to determine whether the controller is ready to receive a command or to send a result. The status port must be consulted before a byte may be read or written into the data port.

The floppy controller has three basic phases: (1) a command phase; (2) an execution phase; and (3) a result phase. During the command phase, a command string is sent, byte by byte, to the controller. The controller then execute the command. At command completion, the CPU must read all result bytes from the controller before the next command can be issued. The NEC 765 controller chip is more completely described in the Appendix.

Data is transferred between memory and the FDC by DMA. The PIO chip is used to provide several control signals for the floppy controller. The following signals are supplied by the PIO, channel A:

bits <0,1> - Drive select

bit < 2 > - Head load
If this is high, then the head on the selected drive is loaded. Depending on which type of drive is used the CPU must then delay before attempting to read or write from the drive.

bit < 4 > - Small/large precomensation
This should be high if between tracks 22 and 59 in double density.

bit < 5 > - Select precompensation
This should be high if above track 22 in double density.

4.7 The DMA chip

The DMA chip can be used to transfer data between an I/O devise and memory, or from memory to memory. The DMA chip can only be used by one device at one time. The device is selected by bits 0, 1, and 2 of channel A of the PIO chip.

PIO channel A bits 2, 1, 0	Device Selected	Z80 port for selected drive
0 0 0	NEC765 floppy controller	18h
0 0 1	SIO1 channel B (i.e., RS-422 network)	29h
0 1 0	SIO2 channel A (i.e., RS-232 port 2)	20h
0 1 1	SIO2 channel B (i.e., RS-232 port 3)	21h
1 0 0	Parallel port 1, input	01h
1 0 1	Parallel port 1, output	01h
1 1 0	Parallel port 2, input	02h
1 1 1	Parallel port 2, output	02h

In the standard BIOS provided by DMS, the DMA chip is used for floppy disk accesses and network accesses. The DMA chip is decribed more completely in Appendix B.

4.8 Interrupts

Interrupts on the DSC/3 and DSC/4 can be generated by the following chips:

DMA (highest priority)
SIO #1
SIO #2
PIO
CTC (lowest priority)

The DMA chip has the highest priority. This means that an interrupt request from a lower priority device will not be serviced until the DMA interrupt has been processed. Thus, for example, the CTC chip cannot interrupt the DMA interrupt handler, even if interrupts are enabled.

Each of the above chips is capable of generating an interrupt on a wide variety of conditions. The operating system in the DSC/3 and DSC/4 uses only a small number of these choices. Other operating systems may need to use different types of interrupts, depending upon their requirements.

Digital Microsystem's BIOS uses the following interrupts:

(1) DMA transfer complete

The DMA chip issues an interrupt when it has finished transferring data to or from the floppy disk or the network.

(2) SIO1 channel A character received

To allow console type-ahead on port 0, the SIO1 channel A is programmed to interrupt when a character is received.

(3) Network first and last character received

When the first character or the last character of a network block is received, SIO1 channel B issues an interrupt.

(4) PIO channel A, bit 7 (front panel / halt)

When the front panel switch is pressed, or when a halt instructon is executed, bit 7 of PIO channel A goes high and causes an interrupt to occur.

(5) CTC channel 3 (60 hertz clock)

A 60 hertz signal is connected to CTC channel 3. The CTC is programmed to interrupt every 1/60th of a second, so that the time of day can be maintained in locations 40h to 46h.

To handle interrupts, the Z-80 interrupt mode 2 is used. When an interrupt occurs, the upper 8 bits of the address of the interrupt vector are supplied by the I register while the lower 8 bits are provided by the chip which is causing the interrupt.

For example, the following code is used to process an interrupt every 1/60th of a second:

```
;
; Program the CTC chip, channel 3
CTC3 =    33h
CTC0 =    30h
     mvi  A,0C5h     ; set the CTC counter
     out  CTC3
     mvi  A,1        ; initial value of counter
     out  CTC3
     mvi  A,CTCvect&0FFh
     out  CTC0       ; setup vector address
        .
        .
CTCvect:
     .word CTCint    ; interrupt vector
        .
        .
CTCint:
     ei         ; interrupts must be re-enabled
     <save user registers>
     <process interrupt>
     <restore registers>
     reti       ; end of interrupt handler
```

It is usually a good idea to keep interrupts enabled all the time, except for timing sensitive code and critical sections of the operating system. Each interrupt handler must be terminated by an RETI instruction. This instruction returns the CPU to the program that was executing before the interrupt occurred. It also allows the interrupting chip and lower priority chips to cause new interrupts. However, it is not desirable to use a RETI instruction when the DMA chip is programmatically reset (by sending 0C3h to port 38h). When the DMA chip is reset, it automatically allows other chips to start interrupting. However, interrupts are not re-enabled until an EI instruction is executed.

4.9 Parallel ports

Two 8-bit parallel ports are provided on the DSC/3 and DSC/4. They can be accessed by using IN and OUT instructions on Z80 ports 1 and 2. Two control bits are also provided for each parallel port. These control bits are accessible by reading bits 3,4,5, and 6 of channel A of the PIO chip. They can be used for handshaking with auxiliary I/O devices.

In the standard CP/M BIOS as supplied by DMS, port 1 is used to interface with the hard disk, while port 2 is used as a Centronics compatible printer interface.

Note: The parallel ports should not be confused with the PIO chip. The PIO chip is used to provide miscellaneous control sinals for different devices within the DSC/3 and DSC/4. It cannot be used as an interface with an auxiliary I/O device.

4.12 The 128K Byte RAM Board

The memory board used in the DSC 4 contains 128K bytes of MOS dynamic memory and may occupy any of the eight 128K blocks of address space allowed by the Multibus. The address block occupied is selected by connecting one of the eight pairs of jumper pins located between U12 and U13. Typical access time from a read command is 330 ns or better. Worst case access time is 750 ns and is only encountered when a read request occurs immediately after a refresh cycle is started. To increase the speed of processors such as the Z80 which can terminate wait cycles before requiring data to be valid the 128K memory board is normally configured to provide XACK/ typically 325 ns after a read request just as the data becomes valid. A Multibus-conforming XACK/, which typically goes low 410 ns after a read request, is available by connecting the upper two pins of the jumper between U9 and U10 and cutting the default trace between the lower two pins. The memory board does not require data to be valid at the beginning of a write command so that a write can be initiated before valid data is available. Although this does not conform to Multibus specifications, it results in shorter write cycles with some bus masters.

The 128K byte memory board 4 uses 16K bit dynamic memory chips with address multiplexing, RAS/ and CAS/ timing and Multibus slave response timing generated by Intel's 8202 dynamic RAM controller operating with a 25 MHz clock. The board supports both 8 bit and 16 bit operation conforming to Multibus standard operation:

ADR0/	BHEN/	Active data lines	Operation
high	high	DAT0/ thru DAT7/	8 bit read or write to even address
low	high	DAT0/ thru DAT7/ (also DAT8/ thru DATF/)	8 bit read or write to odd address
high	low	DAT0/ thru DATF/	16 bit read or write must be even address
low	low	DAT8/ thru DATF/	8 bit read or write to odd address w/ 16 bit master

(Last operation is not a normal Multibus operaton but is supported by the 128K byte RAM board.)

The parallel operation of upper and lower halves of the data bus is achieved by dividing the 128K bytes of memory into to 64K byte blocks each with its own data-in buffer (U23 and U18), data-out latch (U22 and U17), separate write line and separate parity checking circuitry and associated parity memory. Data at even addresses is stored in the block near the right edge of the board and data at odd addresses is stored in the block closer to the left edge of the board. When an 8 bit master reads or writes to an odd address, bi-directional tranceivers, U20 and U21 transfer data in the appropriate direction between DAT0/ thru DAT7/ and the memory block occupying the odd addresses.

During a write operation to either block of memory the data-out line of the parity memory is floating and is pulled high by a pullup resistor. Parity generator/checkers, U19 and U24, then generate parity bits for each block which are written into the corresponding parity memory. Since the data stored in the parity memory is random before a write operation, reading from an address before writing into both that address and the same address with the lowest bit complimented may result in a parity error.

A memory read operation combines the results of parity checks of both blocks from U19 and U24 and latches an error at the end of either CAS/ or MRD/ if one exists. This generates an interrupt on INTO/, pin 41 of the Multibus, the highest priority interrupt. Once a parity error is latched it will only be cleared by a read from any address in the range of either 60000H to 7FFFFH or C0000H to DFFFFH. The choice of address range used to clear the parity error condition is made by connecting the center jumper pin located between U6 and U7 to one of the adjacent pins. The center pin is connected to the upper pin by a default trace, configuring the board to respond in the lower address range. The trace must be cut if the other connection is made. The read which clears the parity error will lower XACK/ and one of the lower 8 data lines selected by the jumpers between U21 and U22. Error recovery software can use this information to tell which memory board detected the error if each board lowers the data line associated with its location in the Multibus address space. Because of this method of determining the offending memory board, system memory cannot occupy the address block which other boards use to respond to a parity error read. If this address block is required for system memory, none of the jumpers between U21 and U22 should be connected (cut the default trace) and pin 11 of U9 should be lifted. Memory boards will generate an interrupt on a parity error but other methods must be used to determine the offending memory board.

5.2 Software Symptoms

Disk-related error messages may occur occasionally. A error may be caused by a minor problem, such as a full diskette, or a more serious problem, such as a media problem or electronic failure. The discussion in this section identifies all software generated error messages, their probable causes, and any operator action that should be taken.

I/O Error nnnn — Occurs when attempting to boot up CP/M. Can be caused by any one of a number of problems. Your diskette may be damaged. The hardware may have failed. If this error persists after trying a known good diskette, then it is likely that the hardware has failed. If the error number is between 0 and 3FFH, then the error occurred while attempting to read sectors 0 and 1; if the error number is between 9000 and 90FFH, then the error occurred while attempting to read elsewhere on track 0 or 1. The controller's result string is stored beginning at location 9380H; it must be examined to determine the cause of the error.

*** SEEK error — Occurs when a floppy disk drive has failed while stepping to a new track.

*** SYNC error — The NEC765 controller chip is out of phase with the CPU chip. Indicates that the NEC chip was misprogrammed, or that interrupts have been disabled or missed.

*** DATA error — Data CRC error. This error can only occur on a disk read. It indicates that the data on the diskette has been damaged. If you type a RETURN, you may be able to recover the data. However, if the error is in the directory, the diskette may be unusable. It is possible that you may recover the data by reading the diskette on another drive, or by repeating the current operation. Re-writing the offending sector is likely to eliminate this error.

Persistent errors can be caused by diskette wear, improper diskette care, bad diskette quality, bad disk drives, bad power supply levels, or a bad disk controller.

*** ID error — ID CRC error. This error indicates the an ID field on a diskette has been damaged. The same comments apply to ID errors and DATA errors, except that rewriting the offending sector will probably not eliminate this type

of error.

*** TRAC error — The disk read/write head is positioned over the wrong track. The previous seek operation failed to position the read/write head over the proper track. To fix this problem, try re-homing the disk (a warm boot will usually do this). If the error persists, the controller or the drive is probably at fault.

*** ENDT error — Access beyond end-of-track. The DMA chip has failed to interrupt the CPU at the completion of a DMA operation. This can happen when interupts are disabled, even for a short period of time. This can also happen if the Z-80 I register is tampered with.

*** ORUN error — The floppy controller was not serviced prompty. This error may be caused by a DMA chip failure or a software failure.

*** SECT error — Sector cannot be found. This error indicates that the ID field for the current sector cannot be found on the current track. The diskette may be damaged, or the drive may have failed to step properly during the previous seek operation.

*** PROT error — The diskette is write-protected. To allow writing on the diskette, the notch on the end opposite the label must be covered.

*** DENS error — Missing address mark. The diskette is probably formatted in the wrong density. For example, this error is generated when a DSC/2 double density diskette is read on the DSC/3. It is also possible that your diskette has been damaged.

Other errors

*** INT error — Bad interrupt. An interrupt has occurred, but the wrong vector was used. This usually indicates a software error.

*** CALL error — An non-available BIOS feature has been requested. Different versions of the BIOS are needed for different hardware configurations; your BIOS is incompatible with your hardware configuration.