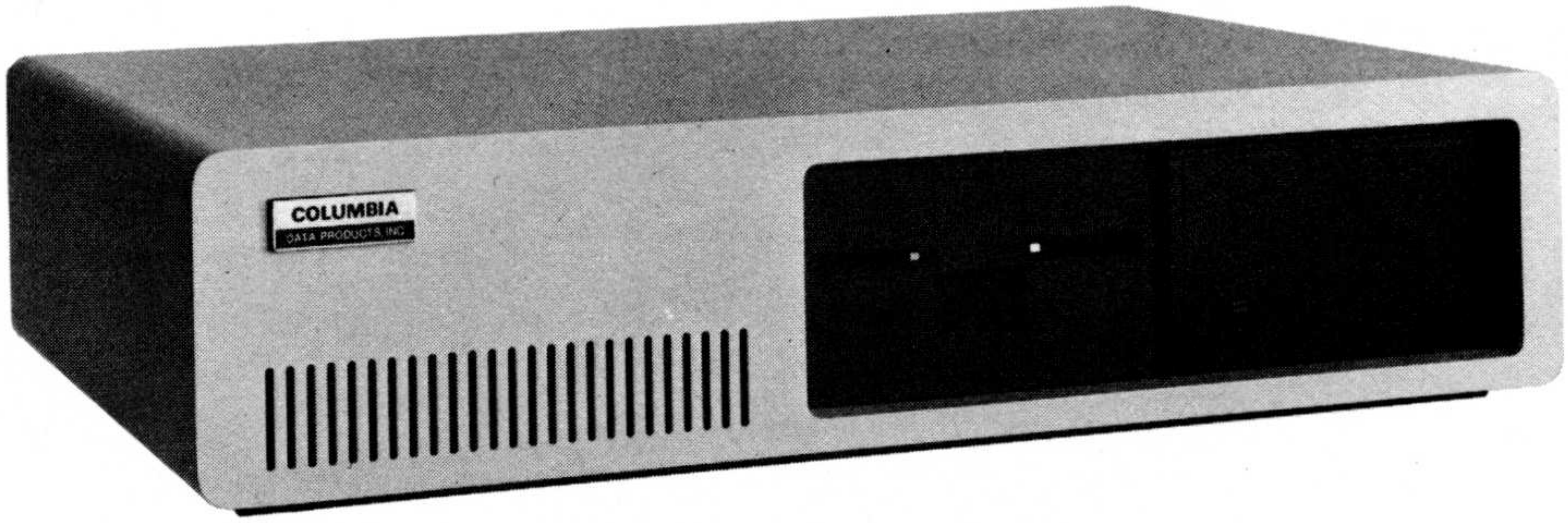
COLUMBIA

DATA PRODUCTS, INC.

MPC OPERATIONS GUIDE

MPC
OPERATIONS GUIDE

MANUAL NO. 1023
FEBRUARY 1983



COLUMBIA
DATA PRODUCTS, INC.

8990 Route 108, Columbia, MD 21045
(301) 992-3400 TWX 710-862-1891

MPC OPERATIONS GUIDE

Manual No. 1Ø23

June 1983

CAUTION

READ SECTION 2 BEFORE STARTING YOUR SYSTEM. IT IS IMPORTANT THAT YOU BACKUP YOUR OPERATING SYSTEM DISK(S) BEFORE ATTEMPTING ANY PROCEDURE. DO NOT USE YOUR DISTRIBUTION DISK(S) AS WORKING COPIES. SEE SECTION 2.

PROPRIETARY NOTICE

All information contained herein relates to the Columbia Data Products Multi-Personal Computer. The contents of this document are proprietary to Columbia Data Products, Inc., and should be so regarded by all parties provided access.

COPYRIGHT

Information presented in this manual is furnished for customer reference only and is subject to change without notice.

LIMITED WARRANTY

Columbia Data Products, Inc., warrants that this product is free from defects in workmanship and material for a period of ninety (9Ø) days from date of shipment. Columbia Data Products' sole obligation under this warranty shall be limited to correcting, without charge, at an authorized Bell & Howell Service Depot, any parts or parts thereof which shall be returned to the Dealer where the product was purchased or an authorized Bell & Howell Service Depot, transportation prepaid, and which, upon examination by Bell & Howell or Columbia Data Products, Inc. (at its option) shall be determined to be defective. The Buyer is obligated to provide warranty validation at the time the equipment is returned. If Bell & Howell (or Columbia Data Products, Inc.) determines that the product is not defective under the terms of this warranty, the Buyer shall pay the cost of all transportation and the repairs at the prevailing repair rates. Columbia Data Products' supplied component parts and Bell & Howell's repairs are warranted for thirty (3Ø) days from date of repair.

This warranty does not apply to any Columbia Data Products equipment that has been subject to neglect, misuse, improper installation and maintenance, or accident.

THIS WARRANTY SUPERSEDES AND IS IN LIEU OF ALL OTHER WARRANTIES EXPRESSED OR IMPLIED, INCLUDING THE WARRANTY OF MERCHANTABILITY AND FITNESS FOR INTENDED PURPOSE. COLUMBIA DATA PRODUCTS, INC., SHALL NOT BE LIABLE FOR LOSS OR DAMAGES OTHER THAN AS SPECIFICALLY STATED ABOVE, WHETHER ORDINARY, EXEMPLARY, SPECIAL, OR CONSEQUENTIAL, CAUSED EITHER DIRECTLY OR INDIRECTLY, BY USE OF THIS PRODUCT. NO OTHER WARRANTY OR LIABILITY MAY BE ASSUMED UNLESS EXPRESSLY AUTHORIZED IN WRITING BY COLUMBIA DATA PRODUCTS, INC.

*Warranty period for the End User will be effective ninety (9Ø) days from receipt of shipment upon proper validation (Bill of Sale and Warranty Card on file at CDP).

*Warranty period for Dealer will be effective ninety (9Ø) days from shipment upon proper validation (CDP/Distributor).

RETURN MATERIAL AUTHORIZATION

In the event it should become necessary to return this product to Columbia Data Products for service or repair, it will be necessary to first contact the Columbia Data Products Service Department to obtain an RMA number and shipping instructions. For this information, telephone (301) 992-3400.

PLEASE, first direct all queries and problems to your dealer.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION 1: INTRODUCTION AND PRODUCT DESCRIPTION

1.1 General

The Columbia Data Products (CDP) Multi-Personal Computer (MPC) is a disk-based processing and storage unit. Whether for use in the office or at home, this MPC will be capable of fulfilling all your personal needs from word processing to business applications. Utilizing 5 1/4-inch diskettes, the MPC is equipped with two 5 1/4-inch double sided drives and the capability to add a 1ØM 5 1/4-inch Winchester hard disk drive for additional data storage.

Available operating system software includes either single-user MS-DOS or CP/M-86, or multi-user, multi-tasking MP/M-86. This operating system software provides users with a host of compatible software packages for both personal and business applications.

The standard CDP 16-bit MPC hardware configuration provides 128K RAM with parity, two RS-232 serial ports, Centronics parallel printer port, a Direct Memory Access (DMA) controller, Winchester Controller interface tone generator/ speaker, keyboard interface, and eight IBM-PC compatible expansion slots.

IBM is a trademark of International Business Machines.
CP/M, CP/M-86, and MP/M are trademarks of Digital Research.
MS-DOS is a trademark of Microsoft, Inc.
Z8Ø is a trademark of Zilog, Inc.

1.2 MPC System

A. CDP 128K Memory Computer
B. Popular Key Layout, Detachable, Serial Interface Keyboard
C. Video Monitor (Color or Black and White)
D. Disk Drives (5 1/4-Inch)
   (1) Dual Floppy Drives or
   (2) One Floppy and One Hard Disk

The MPC basic unit provides the operator with a complete data processing system.

Figure 1-1. MPC System

1.3 Product Description

1.3.1 General

The MPC is a modular desktop unit measuring 22.5 inches wide by 5 inches high by 15 inches deep. Weighing approximately 25 pounds, it consists of the following basic parts: a master printed circuit board, eight expansion/peripheral slots, two independent disk drive assemblies, and a power supply. Additionally, the system is configured for a detachable keyboard and separate monitor or a dumb terminal. The external configurations such as the keyboard are customer options to suit their particular need. Connections for all external devices are located on the back panel (see Figure 1-3).

The standard configuration of the MPC contains a 16-bit 8Ø88 processor, 128K RAM with parity, dual RS-232 ports, Centronics parallel printer interface, tone generator/speaker, DMA controller, interrupt controller, floppy disk controller and Winchester hard disk interface, all on a single printed circuit board.

A detachable module, the peripheral tray, contains the power supply and either dual 5 1/4-inch floppy disks with 64Ø Kilobyte storage or a 5 1/4-inch floppy disk with a 1Ø Megabyte Winchester hard disk with controller.

In order to satisfy hardware expandability requirements, the MPC contains eight IBM-PC compatible expansion board connectors as part of the standard configuration. Several expansion boards and systems are available from Columbia Data Products. These boards and systems include 128K/256K RAM (up to 1 Megabyte expansion), additional serial ports, 8-inch floppy disk systems, Z8Ø CP/M processor expansion system, monochrome and color graphics controller, IEEE Bus controller, tape systems, and many more.

MULTI-PERSONAL COMPUTER

STANDARD FEATURES

Single Board Processor System with:

*16-Bit 8Ø88 Processor
*128K RAM with Parity
*Double Density Floppy Disk Controller
*DMA and Interrupt Controller
*Two RS-232 Serial Ports
*Centronics Parallel Printer Port
*IBM-PC Compatible Keyboard Port
*Winchester Hard Disk Interface
*Peripheral Module with Dual 5 1/4-Inch Floppy Disk
*Programmable Tone Generator/Speaker
*Eight IBM-PC Compatible Expansion Board Slots

OPTIONS - Available from Your Dealer

+ *128K/256K RAM with Parity Expansion Boards (Expandable to 1 Megabyte RAM Capacity)
+ *Single RS-232 and Dual RS-232/RS-422 Expansion Boards
+ *8Ø87 Arithmetic Co-Processor
+ *PROM Expansion Boards
+ *MODEM Board
+ *CRT Controller
+ *8-Inch Floppy Disk Controller Expansion Board with External Floppy Disk System
+ *Expansion Winchester Hard Disk Systems
+ *Z8Ø Soft-Card Processor Systems with CP/M 8Ø Compatibility
++ *IBM-PC or Custom Keyboards
++ *CRT Controllers, Black and White or Color

++ *CRT Monitors
+ *IEEE Bus Controller
++ *Cartridge and 1/2-Inch Tape Systems
++ *All IBM-PC Compatible Expansion Boards
++ *External Printers and CRT Terminals

+ Available from Columbia Data Products
++ Available from IBM and Other Vendors

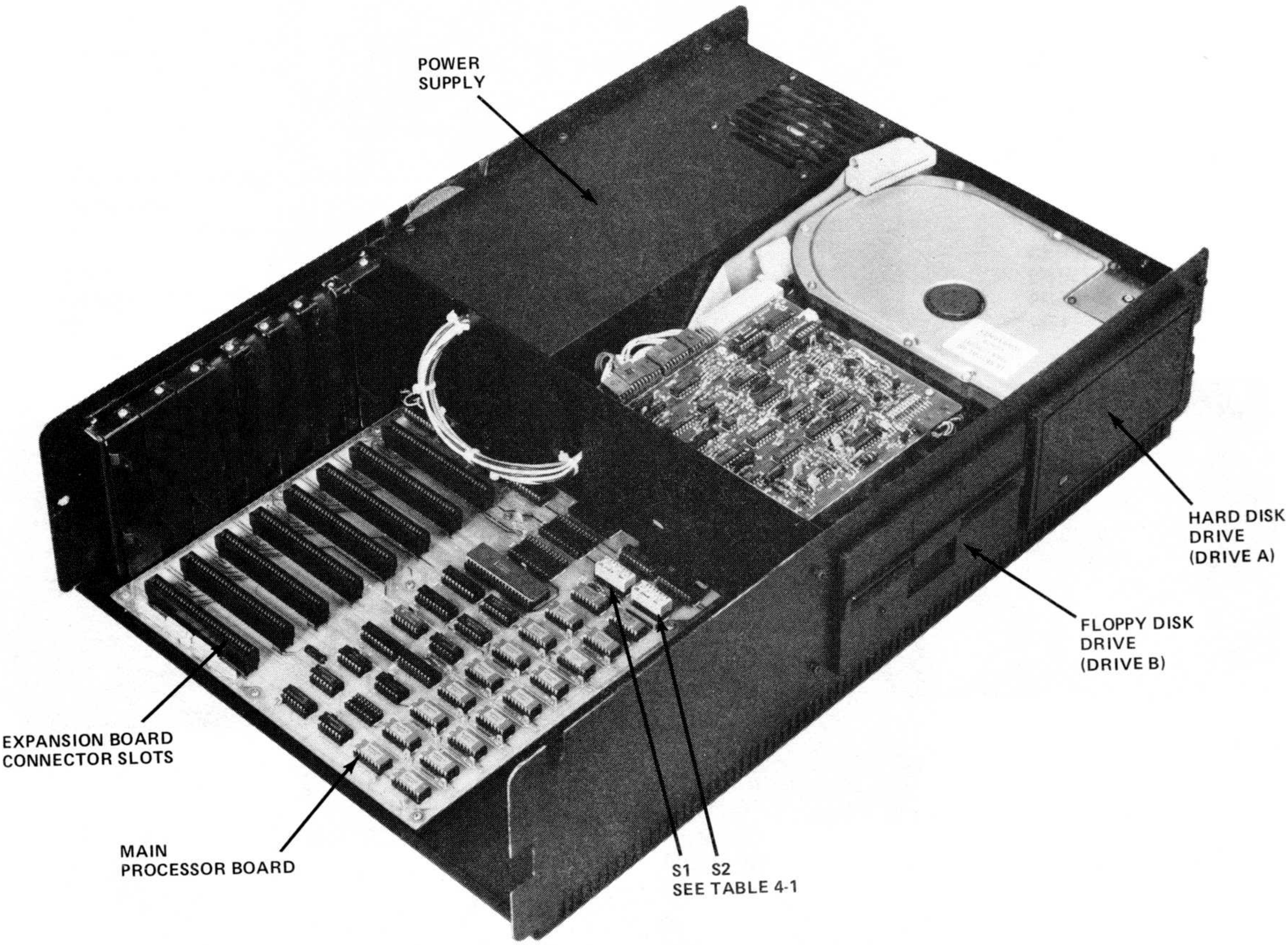


Figure 1-2. Unit Modules

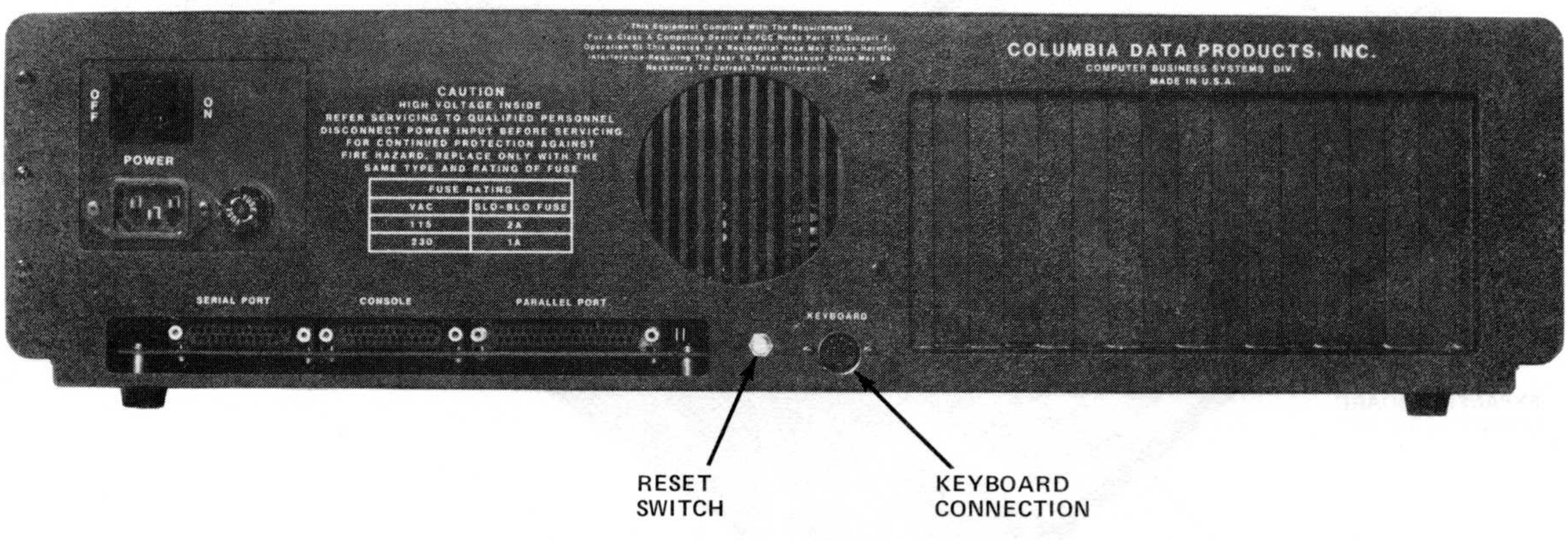


Figure 1-3. Unit Back Panel

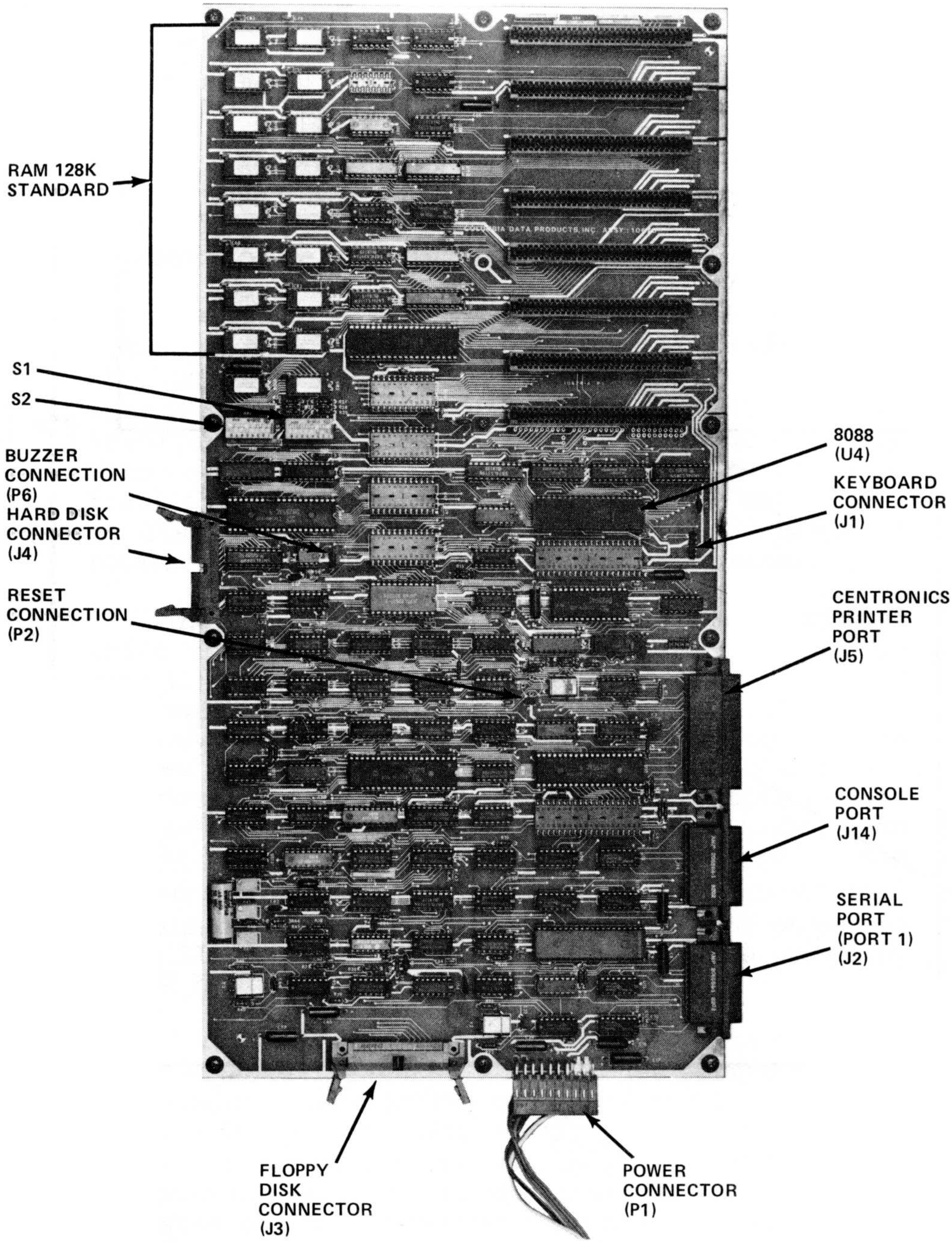


Figure 1-4. Main System PC Board

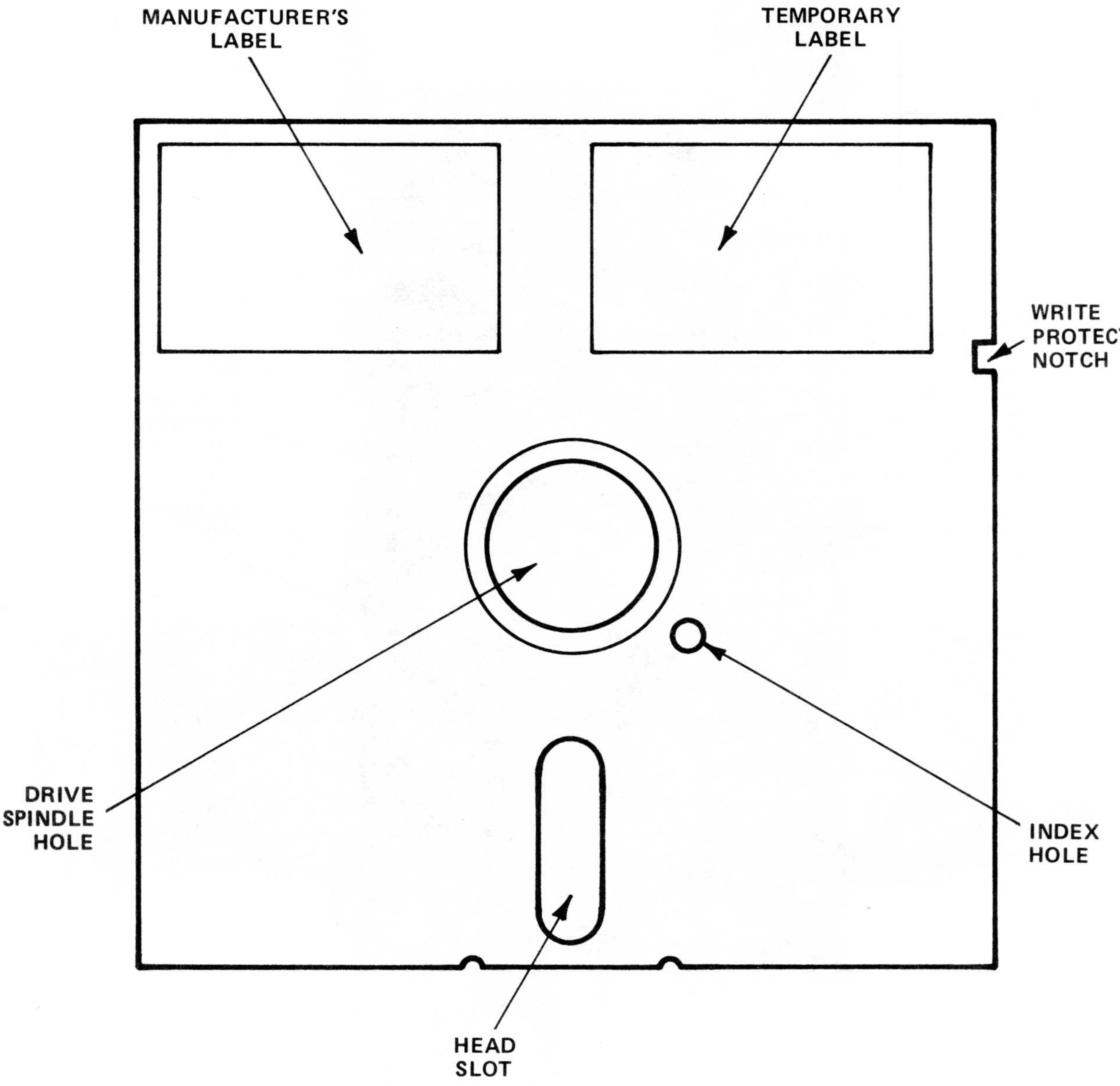


Figure 1-5. 5 1/4-Inch Floppy Diskette

The main memory on the processor board includes both RAM and EPROM. The 128K RAM portion of the memory, seen in the upper left corner of Figure 1-4, consists of 2 banks of 18 metal oxide semiconductor (MOS) integrated circuits. Four 4K EPROMs are located adjacent to the 8Ø88 in Figure 1-4.

The standard MPC system is comprised of double-sided (4Ø data track per side), double density, 32ØK disk drives to handle soft-sectored diskettes. In the standard format (double density), data can be transferred to and from memory at a rate of 25Ø kilobits per second; the average access time track-to-track is 5 msec for 48 TPI. The average capability of disk drive access time is 22Ø msec. Diskette rotational speed is 3ØØ revolutions per minute.

The disk drive modules are individually mounted units that are secured to the chassis for maximum stability during operation. Each assembly consists of read/write and control electronics, a drive mechanism, a read/write head, and a track-positioning device. The drives feature a special diskette clamping/registration design that eliminates diskette damage due to misregistration and makes possible more than 3Ø,ØØØ interchanges with each diskette. Also, the read/write head in each drive is of a single-element ceramic design that can provide media life exceeding 3.5 million passes per track and a head life exceeding 2Ø,ØØØ hours.

A typical jacketed diskette is shown in Figure 1-5. The jacket measures 5 1/4-inches square, with a drive-spindle hole in the center, a small index hole, and a write protect notch at the upper right (covered = protected). A temporary identification label for the user can be placed in the upper right-hand corner, opposite the manufacturer's permanent label. Inside the jacket is the diskette, which consists of sectors and tracks for storing data. At the start of the operation, the first sector is located via the index hole as the diskette rotates in the jacket and the read/write function is performed through the head slot by the drive controller. Information on the diskette can be protected from accidental erasure by activation of the write protect feature.

2.1 General

This section of the User's Guide will enable the user of the MPC to proceed step-by-step through the various programs. Each procedure is presented in such a manner as to allow a beginner in microcomputer operation to proceed step-by-step through each operation and build confidence as they gain experience.

When unpacking the MPC, exercise care in its handling. As with any quality electronic system, damage can result from careless treatment. If software has been provided with the unit, do not bend or fold the diskettes contained in the shipment, permanent damage may result. For information on the proper handling of diskettes, see Section 2.4 of this manual or the back of a diskette jacket.

To unpack the MPC, first remove all diskettes and software manuals as well as the power cord and/or connector cables. Next, remove the unit from the shipping container by placing one hand in front and one in back of the unit, and lift straight up. Place the unit on a clean, flat, uncrowded surface and examine it for possible damage. If any is apparent, notify the carrier that delivered the unit and CDP Customer Service Department. If no damage can be found, proceed to the setup phase.

2.2 Setup

The first step in setting up the MPC is to place it on a flat, clean surface within a few feet of an electrical outlet with compatible voltage. (The unit is available in 11Ø-volt 6ØHz and 22Ø-volt 5ØHz. Check the identification plate on the back of the unit for the power rating.) Remove any tape from the disk drive doors on the front panel, open the doors by pressing the top of the door latch, remove any protective inserts, and close the doors by pushing the latch down until they click into place. Next, check to see that the ON/OFF switch on the back panel of the unit is in the OFF position. Once this is done, plug the power cord into the applicable receptacle on the back panel of the unit.

NOTE: DO NOT TURN ON UNIT UNTIL INSTRUCTED AT THE END OF THIS SECTION.

The MPC is easily transportable, allowing it to be set-up on any suitable flat surface. The following set-up procedure should be adhered to:

A. Connect the appropriate end of the power cord into the back of the MPC. DO NOT plug the opposite end into a power outlet at this time.

B. Set the Video Monitor on top of the MPC unit, and connect the video cable to the back of the Video Monitor, and to the video board connection on the back of the MPC (Figure 1-3).

C. Connect the keyboard to the MPC using the coiled stretch cord (similar to a telephone cord). The telephone-like connector inserts into the bottom of the keyboard and the six prong connector inserts into the back of the MPC (Figure 1-3).

D. Remove any inserts from the floppy disk drives.

E. Prior to pluging directly into a well grounded 3-prong outlet, check the back of your MPC for the proper voltage the machine is set for (example: Model 1600-1/110, the 110 indicates the proper voltage). (See Figure 2-1.) Make sure the ON/OFF switch (Figure 1-3) is in the OFF position; plug the power cord from the MPC into the appropriate outlet (110VAC/60 Hz or 220VAC/50Hz).

F. Save the shipping box and packing materials. They are a good way to ship or move your MPC.

Note: If the MPC is used with a serial display terminal not provided by CDP, a reversing cable must be used. If so, plug one end into the serial port and the other end into the console port at the back of the MPC (see Figure 1-3). A "Reversing" cable is a standard RS-232 cable with certain pins reversed. See Table A-6. Your dealer will be able to assist you in obtaining this cable.

Note: If changes are made to the memory configuration, type of console used, disk drives on line, or boards added which change the configuration ordered, see Appendix F before proceeding.

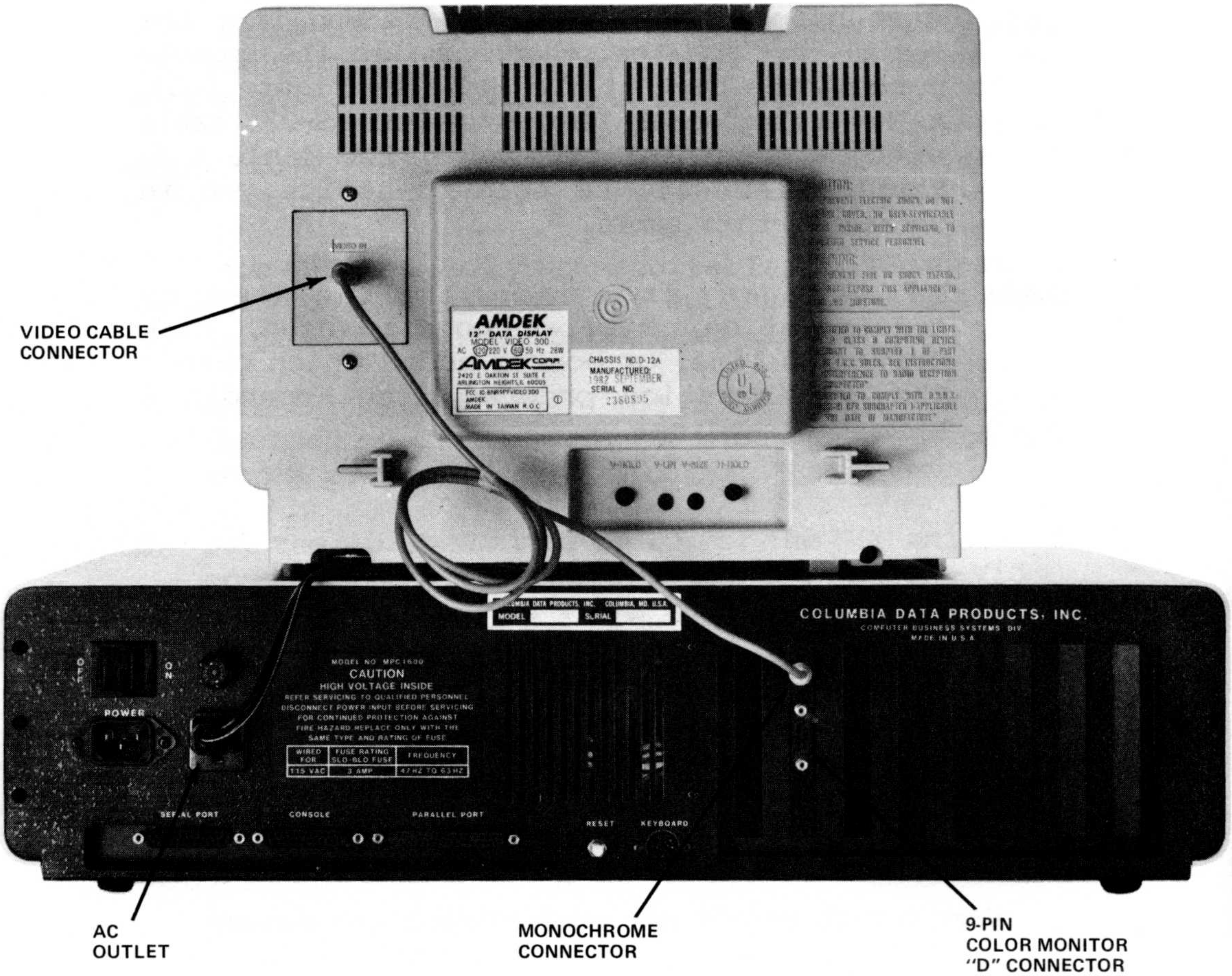


Figure 2-1. MPC System Connections

2.3 Operation

All aspects of operation of the MPC are performed at the terminal or keyboard. At the terminal or keyboard, the operator makes all inputs to the system in order to utilize it. At the same time, attention must be paid to the terminals screen for information and/or instructions being provided by the computer. In addition to the terminal and keyboard, a printer may also be connected to the unit for hard-copy information.

An operating system disk which contains one of the following is provided with each MPC.

1. MS-DOS ver 1.25
   MS-BASIC-86
   MS-GW BASIC

2. CP/M-86

2.4 Diskette Handling

Proper maintenance of diskettes is vital for trouble-free operation. It is recommended that all instructions on the diskette envelopes be carefully read and followed. When handling diskettes, never touch the recording surface (Figure 1-5). Always handle diskettes on the jacket area. After using a diskette, place it back into its protective envelope immediately. Exposed diskettes can easily be damaged if they come into contact with smoke, dust, debris, or other environmental hazards. Also, avoid bending, folding, storing in direct sunlight, or placing next to magnets or magnetized objects.

When writing on the diskette label, do not use a ball-point pen, pencil, or other hard marker, as an impression can be made on the diskette causing damage. Use only a soft, felt-tip pen. Also, never try to erase information put on a label. The resulting debris can lodge between the diskette and its jacket causing damage to the diskette.

2.4.1 Diskette Loading

To load a diskette into the disk drive, ensure that it is face up, that is, with the identification label up and the write protect notch to the left (Figure 2-2). Open the disk drive door by pulling the latch up at the center of the drive. Insert the disk as indicated in Figure 2-2 as far into the drive as it will go without forcing it and close the latch by pushing it down until it "clicks" into place. When the drive door clicks into place, the disk is centered in the drive and the Read/Write head engages the disk. When a machine command is entered at the keyboard activating the disk drive and/or at the initial booting of the program, the drive activity light will be lighted. The activity light will also be lighted during data transfer periods. Wait until this light goes out indicating that the drive has stopped before opening the door and removing the diskette.

Note: It is a good practice to insert diskettes AFTER the MPC is turned ON and to remove BEFORE the MPC is turned OFF.

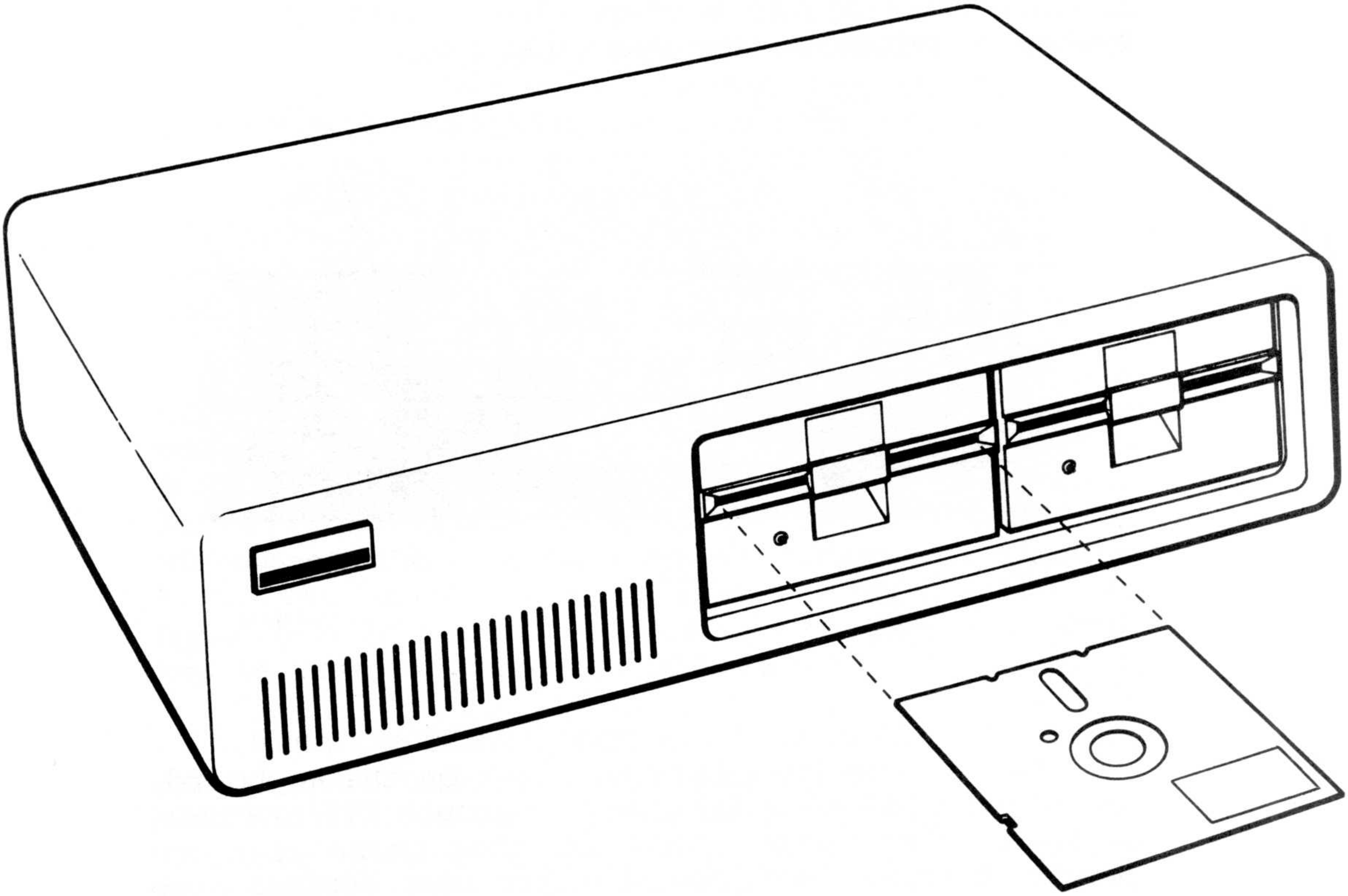


Figure 2-2. Diskette Loading

2.5 Keyboard

The keyboard for the MPC is configured essentially like a typewriter. The cream-colored keys in the center of the keyboard are the basic "typewriter" or data entry keys. The keys to the left and right of the basic keyboard are special function keys and will be described separately.

The MPC uses international symbols for TAB, SHIFT, BACKSPACE, AND ENTER (see Figure 2-3). The MPC can be reset from the keyboard by pressing the DEL key while holding down the ALT key and the CTRL key. This reset allows you to restart the current operating system or reboot another operating system.

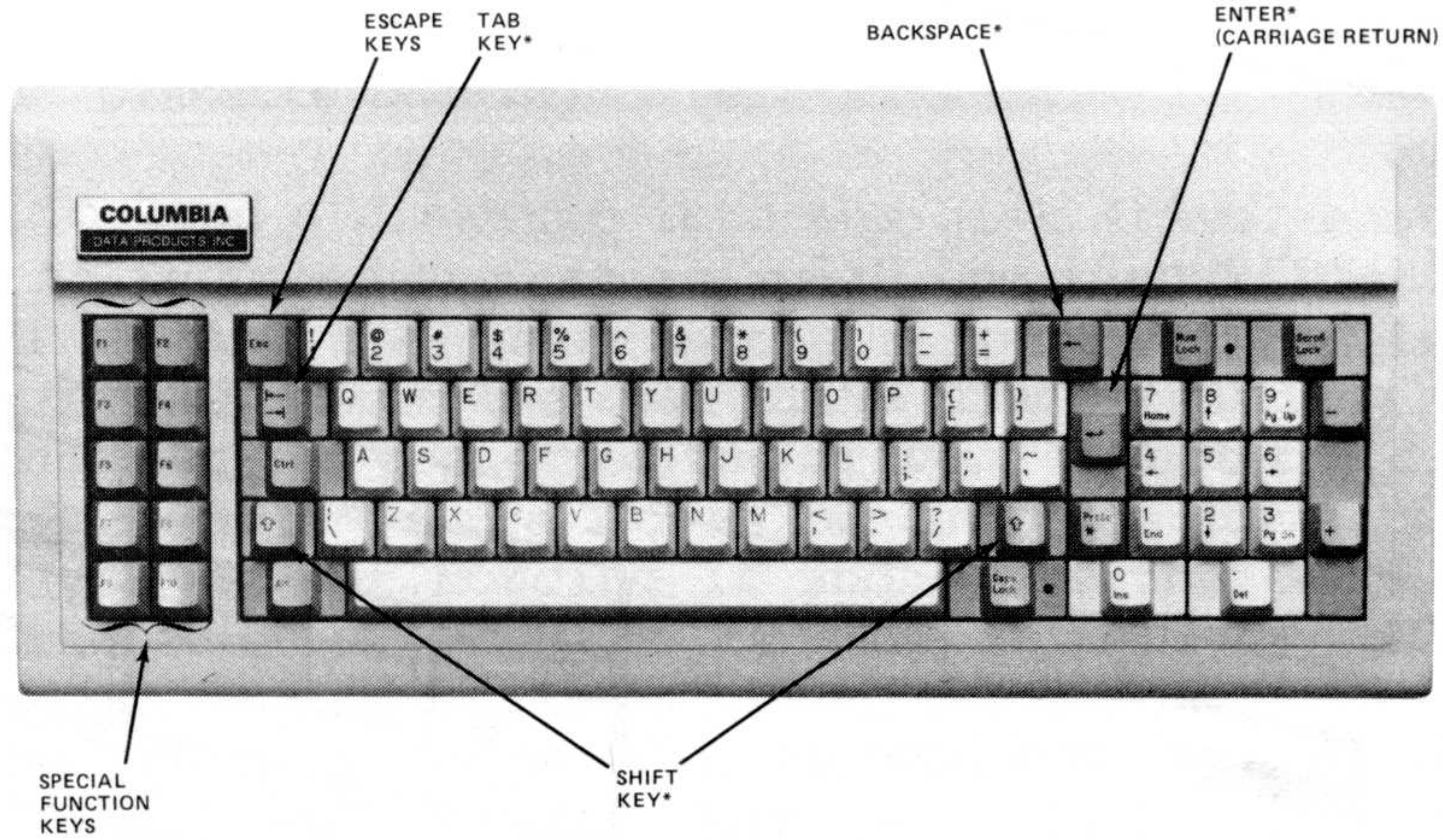


Figure 2-3. Keyboard

Proceeding from left to right on the keyboard, the block of 1Ø keys labeled F1 through F1Ø are user defined by software. That is, that these keys can provide special functions that the user desires such as designating one key to enter a specified number of characters when pressed. For a description of the Function Keys, see Section 4.3.7 for CP/M-86 and Section 3.3 for MS-DOS.

The "ESC" (escape) key (for CP/M-86 not for MS-DOS) allows the operator to clean the current line of data input. Inadvertent pressing of this key will eliminate or erase the current line of text that an operator is working on and the prompt question as well. Until the operator becomes aware of this function, it can cause confusion. If the ESC (escape) key is inadvertently pressed, the only harm done is that the current line is erased. No harm is done to the program or other data that has been entered, and the current line has to be re-typed.

The "CTRL" key, located at the left of the basic keyboard, is used in conjunction with other keys on the keyboard. When using the CTRL key, the CTRL key must be held down while the second key is pressed.

The ALT (Alternate) key enters ASCII character codes directly from the keyboard. The ALT key is held down while the three ASCII digits are typed on the numeric keypad.

The NUM LOCK (Number Lock) key on the right side of the keyboard is used to engage or disengage the numeric keys in this keypad. When the keypad is engaged, the red dot on the right side of the NUM LOCK key will be lighted. When the numeric keypad is disengaged, the keys are used for cursor control which moves the cursor up, down, right or left, and to the upper left corner of the video screen when the "HOME" key is pressed.

When the SCROLL LOCK key is engaged, the cursor control keys (arrows on the numeric keypad keys) will shift the screen text up, down, left or right but will not have any effect on the cursor. This key is only used in certain operating systems.

When the PRTSC (Print Screen) key is used in conjunction with the CTRL key (remember that the CTRL key must be held down when using it with another key), the screen text will be output to a printer.

The CAPS LOCK key, when engaged, will cause all key strokes in the Basic Keyboard portion to appear in upper case or capitalized. Also, when the CAPS LOCK key is engaged the red dot on the right side of the key will be lighted. Additionally, when the CAPS LOCK key is engaged, the shift keys, if pressed, will cause a key to print in lower case. It should be remembered that the shift keys provide an opposite function to the CAPS LOCK key on the Basic Keyboard.

2.6 Monochrome/Color/Graphics Board

Graphics adaptation for the MPC is obtained through incorporation of CDP's compatible graphics board. The color graphic board has a 62-pin connector that can be inserted into any slot available or made available. Board switching is not necessary.

2.7 Peripherals

There are two basic peripherals for the MPC: the monochrome display and the printer. (Refer to Appendix F for information on interfacing printers or other devices.) The monochrome display or CRT (cathode-ray tube) provides rapid communication between the Multi-Personal Computer and the operator. The Multi-Personal Computer uses a green-colored phosphor (P-31) display that presents a black and green image.

2.7.1 Monochrome Display Features

High Resolution Display
Non-Glare Display Screen
Low Power Consumption
120/220V Interchangeable Power Transformer
Light and Compact - Easy to Carry

The back of the monochrome display is well laid out providing easy access to the power cord and video input/output connection as well as the fine tune adjustment controls.

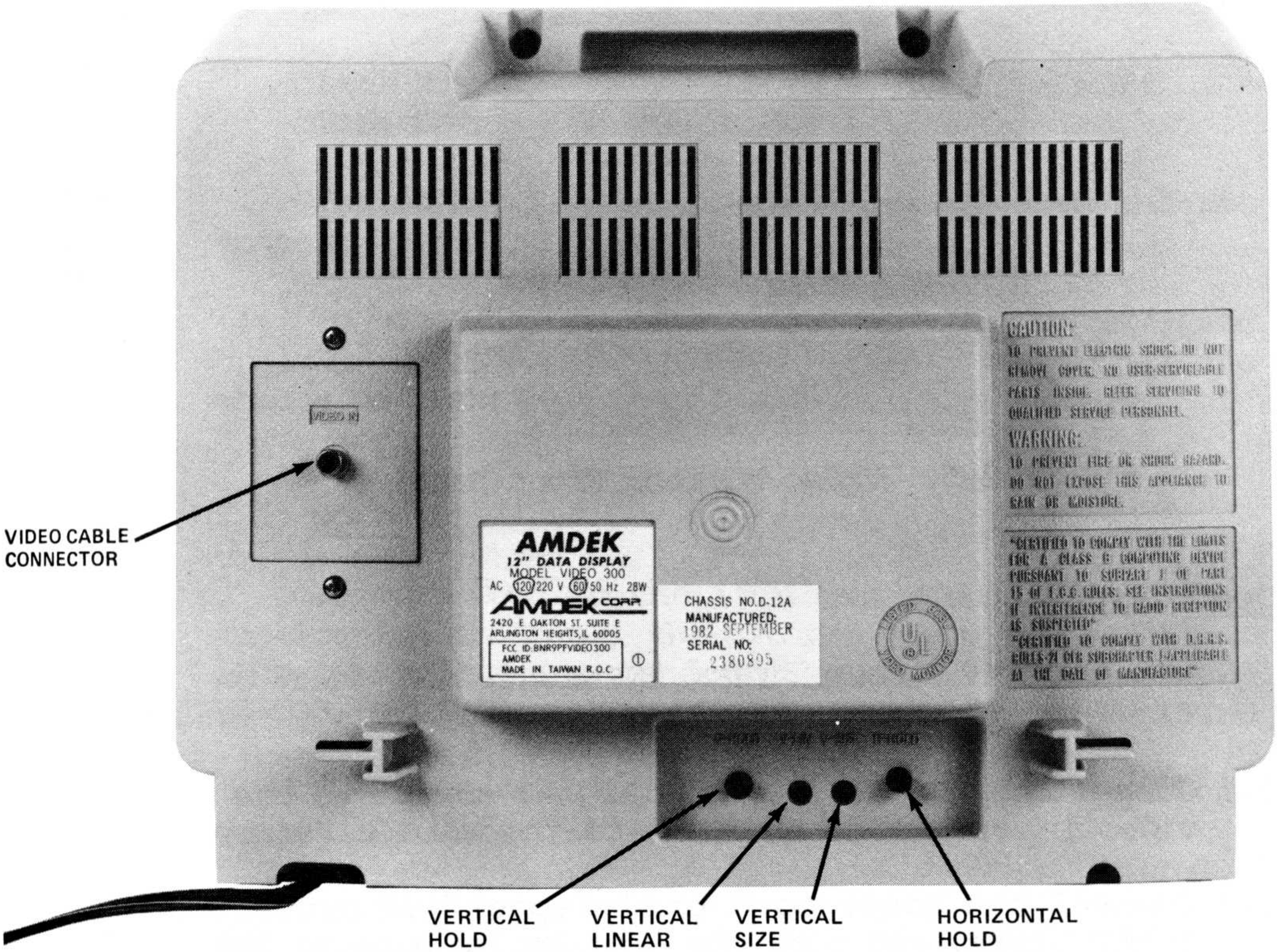


Figure 2-4. Video Monitor Back Panel

WARNING

High voltage is present inside the unit. DO NOT remove the back cover of the cabinet. If the monitor does not function properly, contact your local dealer or qualified service personnel.

FRONT PANEL CONTROLS

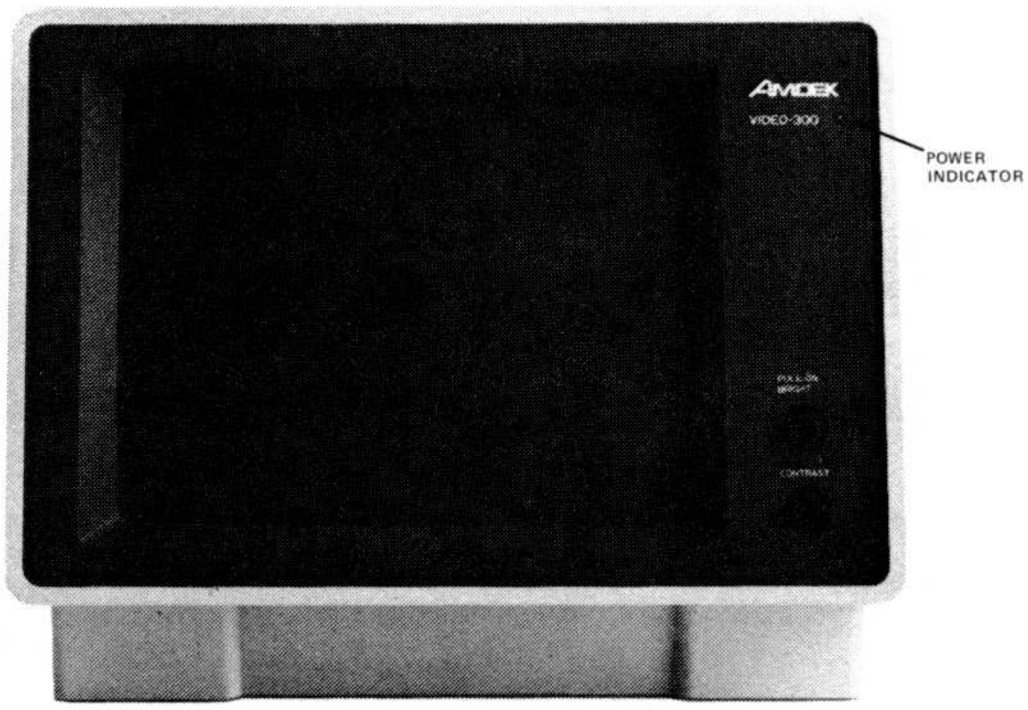


Figure 2-5. Video Monitor Front Panel

Operation

(FOLLOW THIS PROCEDURE WHEN INSTRUCTED TO TURN ON UNIT)

1. Connect the cable from the signal source to the video input jack on the back of the unit.

2. Connect the power cord to the specified outlet (120VAC/60Hz) and turn the unit on by pulling the ON switch on the front of the unit.

3. Adjust the CONTRAST and BRIGHTNESS to bring the screen image to a comfortable position that is pleasing to the eyes.

4. To adjust the picture to the center of the screen, use H-HOLD and V-HOLD controls at the back of the unit.

CAUTION

Do not use or store the monochrome display in areas with high humidity, high temperature (direct sunlight), or areas that are extremely dusty.

Do not cover the ventilation slits during operation.

Clean the display screen only with a soft cloth moistened with alcohol.

The monochrome display is manufactured and set to operate at 120VAC/60Hz current.

To convert the monochrome display unit to operate at 220VAC/50Hz, it is recommended that the change be made by qualified service personnel.

2.8 Software

In addition to the implementation of numerous application programs, the MPC can execute a variety of high-level languages and 8088 assemblers under CP/M, MS-DOS, or MP/M.

The disk operating system(s) provided by CDP are compatible with the IBM Personal Computer.

2.9 System Startup

The MPC system has 16K of read only memory to enable it to do power-up/reset, diagnostic testing, operating system boot, and monitor functions. Each of these functions are discussed in the following paragraphs.

POWER-UP/RESET:

Each time that the system is powered on or reset via the reset switch at the back of the unit, the boot ROM is enabled and the initialization routine is run.

Note: If an external hard disk peripheral is attached to the MPC, it must be powered on before the MPC is powered on.

The initialization routine configures all of the I/O devices on the main system board and tests RAM. If an error occurs during the testing of RAM, the following audible indications will be given:

> Note: Audible indicators are minimal error flags available to software at this time because no stack has been assigned and no terminal device has been located for use. This audible indicator routine can be invoked whenever stack area is suspect, to insure error reporting.

AUDIBLE INDICATORS:

2 beeps = Parity Error
3 beeps = I/O Channel Error
4 beeps = ROM Error
5 beeps = RAM Data Error

If all of the testing is accomplished, the system will respond with a single 1/2 second tone and continue according to the input output media attached.

If a dumb terminal is used, another tone will sound and the system will produce another 1/2 second tone then wait until an ASCII period (.) is typed at the terminal. The system uses the ASCII character to determine the baud-rate of the terminal device. If an ASCII (.) is not received in 5 seconds, the system will default to 192ØØ baud. If a keyboard and monitor is attached, then no baud rate determination is required. (See Appendix F for information regarding the dumb terminal as a console.)

The system will then present a decision message to the console which requires an operator response. The message reads:

TEST MEMORY (Y/N)?"

Type "N" to proceed to system bootstrap or "Y" to test memory prior to booting.

Note: Type [ESC] here to activate the ROM monitor for system testing. See Appendix E for a description of the ROM monitor commands.

If a response is not made within five (5) seconds, the MPC will automatically enter the system bootstrap sequence detailed below.

2.9.1 Memory Test

If the memory test is requested, a test sequence begins that conducts a walking ones and zeros pattern on 4K blocks of memory. A console message describes the segment being tested.

For instance:

SEGMENT#Ø1ØØ - Means testing is being conducted on the 4 K block beginning at absolute address Ø1ØØØ HEX.

Errors encountered during this sequence of tests will be reported as in the "T" test of the monitor (see Appendix E for details).

2.9.2 Operating System Bootstrap

At the completion of the optional memory test, the system bootstrap will begin. Insert a bootable floppy disk into the left-hand floppy disk drive if the system is to be booted from a floppy disk. If the system contains a hard disk which is to be used for the boot, leave the left-hand floppy disk door open (or remove the disk). The following message will be displayed on the console:

```
M.P.C. ROM/BIOS (VERSION X.XX)
SYSTEM BOOTSTRAP
```

Note: X.XX is the version number of the installed read only memory.

After printing this message, there will be a short delay while the MPC determines the disk configuration of the system. Next, the left-hand floppy disk drive indicator will light. If there is a bootable disk inserted, the operating system will boot. If there is no floppy disk, the hard disk system will boot after a slight delay.

When the boot sector has been read from the disk, the operating system will print a sign-on message and then assume control of the system.

Backing Up System Disk

The system disk provided with your MPC is very valuable and care should be taken that it is not damaged or rendered unusable. It should be used once to produce a back-up copy of your operating system and then stored in a safe place to be used again only if your back-up copies are damaged or destroyed.

Make sure that there is a write protect tab over the side notch on your system diskette. If one is not present, place one over the notch before proceeding. This will protect your diskette from accidental writes to it which may destroy your operating system.

Place this original copy of the operating system in Drive A (left drive) and a new blank diskette in Drive B (right drive).

If you are using CP/M 86 disk operating system:

You must first Format the blank disk in drive B by entering the following command when the system is booted up and the A> appears on your monitor.

FORMAT (press return key↵)

The MPC will respond with:

Do you wish to create a bootable disk? (Y/N) Type Y

Your MPC will instruct you to place the disk to be formatted in B where you have already placed the new disk and will ask you to confirm when ready by striking any key. Strike any key and the format will proceed. When this operation is completed, your monitor will display:

Format done.
A>

This has prepared your disk and placed a minimal CP/M system on it. To copy all of the system utilities on your CP/M system disk, enter the command:

COPYDISK (press return key↵)

The MPC will respond with:

CP/M-86 Full Disk Copy Utility
Version 2.Ø

Enter Source Drive (A-D)? Type A
Destination Disk Drive (A-D)? Type B
Copying Disk A: to Disk B
Is this what you want to do (Y/N)? Type Y

If you are using the MS-DOS operating system:

You must first Format the blank disk in Drive B by entering the following command when the system is booted up and the A: appears on your monitor.

FORMAT B:/S

When the /S is used during FORMAT, hidden system files are copied to the disk after it has been formatted. When the MPC informs you that the new disk is formatted, type:

COPY A:*.* B:

USE THE COPY OF THE SYSTEM DISK FROM NOW ON

2.9.3 Failure to Boot Up

If the system bootstrap routine is unable to boot the operating system, one of two messages will be displayed on the console:

1. UNABLE TO READ BOOT

   This indicates that no boot sector could be read because no disk was inserted or the disk is not readable. You may wish to try again with the same disk. If this fails, you must use another disk.

2. NO BOOT PRESENT ON DISK

   This indicates that the boot sector was read, but the disk does not contain an operating system or is not bootable.

In either case, the system bootstrap will then request:

PLEASE INSERT A BOOTABLE FLOPPY DISK
THEN TYPE [RETURN]

If the floppy boot failed and you then decide to boot the hard disk, just remove the floppy disk and type [RETURN]. Conversely, if the hard disk boot failed and you decide to boot a floppy disk, insert the disk and type [RETURN].

Note: If you type [ESC] in response to the above request, you will activate the ROM monitor. See Appendix E for a description of the ROM monitor commands.

For additional hard disk information, see Appendix C.

TURNING ON THE MPC

Now that you have read this section and understand the operation of the MPC:

* Turn the unit on via the ON/OFF switch on the back of the unit - an audible tone will be heard.

* Turn the monochrome display unit on, and adjust as necessary (see Section 2.7.1 and Figure 2-4 and 2-5).

* Insert the desired disk into drive A (left drive), and press the door closure shut until it "clicks" into place.

* The program will automatically boot from ROM, providing initial start-up information and the A> (A: with MS-DOS) (prompt). At this time, BACK UP YOUR DISK!

Note: If you are using the MS-DOS Operating System, refer to Section 3. If you are using the CP/M-86 Operating System, refer to Section 4.

SECTION 3: MS-DOS OPERATING SYSTEM

3.1 General

The MS-DOS disk operating system is one of Microsoft's family of operating systems for 8Ø86 and 8Ø88 microprocessors. It provides a simple but powerful interface between the user and a computer system's resources.

See Section 2.9.2 for instructions on bootstrapping the system. When MS-DOS is booted, it will display the following:

MS-DOS version 1.25
Copyright 1981, 82 Microsoft, Inc.

Copyright 1982, Columbia Data Products
Version 2.Ø5

Command v. 1.17
Current date is TUE 1-Ø1-198Ø (constant)
Enter new date: (D/M/Y - two digit) <CR>
Current time is Ø:ØØ:ØØ:ØØ (constant)
Enter new time: (H/M/S/tenths of a second) <CR>

A:

Note: The new date and time is entered into RAM; when the system is reset (BOOTED), the RAM is cleared and reset requiring a new date and time to be entered. To circumvent the data and time, enter [RETURN] for each request.

The utilities for the MPC are listed in the directory (Table 3-1 DIR) with an explanation of each provided in the Microsoft MS-DOS Disk Operating System manual. Listed here are the utilities necessary for operation of the MPC.

At this point of introduction, you have a basic familiarity with the workings of the MPC. The next area on the disk to be looked at is the directory (DIR). The directory contains a listing of the utilities found in the MPC. The explanation for each of these utilities is contained in the Microsoft MS-DOS Disk Operating System manual included in your shipment provided MS-DOS was ordered. The DIR is similar to that listed in Table 3-1.

To produce a copy of the DIR on the video monitor, simply type in the following at the A: (prompt).

A:DIR ↵

Carriage Return (Key Symbol)
Utility Requested
Prompt

Table 3-1. DIR (MS-DOS)

A:DIR		Size of File In Bytes
COMMAND	COM	5120
DEBUG	COM	6003
EDLIN	COM	2432
CHKDSK	COM	1720
SYS	COM	605
EXE2BIN	EXE	1280
LINK	EXE	41856
MASM	EXE	67968
FILCOM	COM	8320
FORMAT	COM	3684
CREF	EXE	13824
LIB	EXE	32128
LOADBOOT	EXE	2432

Table 3-1. (Continued)

BUILDHD	DAT	1152
HDBOOT	SYS	512
	15 FILE(S)	

A:

To find out how much usable space is available to perform any programming, periodically check the space available on the disk. This can be accomplished as follows:

A: CHKDSK ↵

Carriage Return Key (Key Symbol)

Utility from DIR (Entered by Operator)

Prompt

At the carriage return, a video monitor display similar to the following will be displayed:

Table 3-2. MS-DOS Disk Access Space

```
A:CHKDSK

  322560  bytes total disk space
    9216  bytes in two hidden files
  196608  bytes in (15) user files
  116736  bytes available on disk

  131072  bytes total memory
  114832  bytes free

A:
```

3.1.1 FORMAT (Floppy Disk)

The FORMAT utility is used to place the basic operating format from one disk onto another. This operation can be performed according to the following procedure:

A: FORMAT ↵

Carriage Return

Utility

Prompt

Columbia Data Format ver. 1.20
Double Sided Disk
Insert new diskette for drive A:
and strike any key when ready

NOTE: Insert disk and press any key at this time.

RESPONSE: Formatting . . .

322560 bytes total disk space
322560 bytes available on disk

FORMAT another (Y/N)? N

Operator Input

A:

To format another disk, press Y and follow the same procedure.

During the FORMAT Procedure, an internal switch is engaged by use of the vergule (/); failure to utilize the "internal switch (/)" will NOT allow the desired procedure to be enabled.

FORMAT Example:

SW1 = Format single sided disk

Example: FORMAT [/S] [/1] [/H]
<DRIVE>
Where <DRIVE> = A:, B:, C:, D:, or E:
/S = Copy System
/H = Hard Disk
/1 = Single Sided

SW2 = /1 = Format single sided disk (default is double sided)

SW3 = /H = Format hard disk

A: FORMAT B:/S ↵

Carriage Return Key Function
System Tracks for MS-DOS
Internal Switch
Disk Drive
Utility
Prompt

Columbia Data Format ver.1.10
Double Sided Disk
Insert new diskette for drive B:
and strike any key when ready

Formatting . . .   System transferred

When Formatting is finished
When Formatting is taking place

322560 bytes total disk space
13824 bytes used by system
308736 bytes available on disk

FORMAT another (Y/N)?

To format another disk, press Y (yes). To discontinue formatting, press N (no). If, for some reason, the "Y" key is pressed instead of the "N" when formatting is to be discontinued, press and hold the CTRL key down and press the "C" key. This function will place you back to the A: prompt for initial input. (If the "Y" key is pressed instead of the "N" key, the disk will still get formatted if the second pass has already started.)

CAUTION

> A hard disk format should not be interrupted. If this happens, the system must be returned to Columbia Data Products for a "PRIMARY HARD DISK FORMAT."

3.1.2 COPY (File Transferring)

To transfer a file from one disk to another (drive A to drive B), the following procedure is utilized:

A: COPY BUILDHD.DAT B: ↵

- ↵ — Carriage Return
- B: — Drive copied to
- BUILDHD.DAT — File to be copied
- COPY — Command function
- A: — Prompt

3.1.3 ERASE or DEL

The ERASE command, though not listed in the DIR (Directory), is contained under the Operating System File as a resident command. To utilize the ERASE command to eliminate (erase) a file, proceed as follows:

A: ERASE (File to be erased) <FILESPEC>
A: DEL <FILESPEC>

After erasing a file, always check the DIR to ensure that the file has been erased.

A: DIR   or
A: DIR B: if the file was on drive B DIR

Additional files on this disk can be checked and utilized in conjunction with the Microsoft manuals.

3.1.4 DEBUG

DEBUG is a debugging program used to provide a controlled testing environment for binary and executable object files. DEBUG eliminates the need to reassemble a program to see if a problem has been fixed by a minor change. It allows you to alter the contents of a file or the contents of a CPU register, and then to immediately reexecute a program to check the validity of the changes.

Example: A: DEBUG
load data from a disk

```
>LØ 1 Ø 1
          Amount of sectors
          to load
        Sector Ø (starting
        sector)
      Disk 1(A)
   Memory location Ø
  Load
```

```
>DØ
    Dump

>EØ
    Examine

>Q  <CR>
    Quit

A:
```

See MS-DOS Operating Manual for more information.

3.1.5 LINK

LINK is a relocatable linker designed to link together separately produced modules of 8Ø86 object code. LINK is also used to link object files and object libraries to create executable .COM (File EXE2BIN is needed to do this) and .EXE files.

```
Example:  A: LINK
          Microsoft Object Linker V1.1Ø
          (c) Copyright 1981 by Microsoft Inc.

          Object Modules [<FILESPEC>.obj]:
          Run File [NUL]:
          List File [NUL]:
          Libraries [NUL]:
```

3.1.6 MASM

Assembling with MACRO-86 requires two types of commands:

Method 1 MASM

```
Object Modules [<FILESPEC> .obj.]:
Run File    [NUL]:
List File   [NUL]:
Libraries   [NUL]:
```

Method 2 MASM (source), (object), (listing), (cross-ref) [/switch...]

(Refer to Chapter 5 of the macro assembler manual in the Microsoft MS-DOS Disk Operating System.)

3.1.7 Installing MS-DOS

The MS-DOS Operating System ORIGINAL diskette should not be usedin the daily operation of the MPC. It should be used for producing the first MASTER diskette, then stored in a safe place in case all copies made from it become defective or lost.

DO NOT REMOVE THE WRITE-PROTECT STICKER FROM THIS MASTER DISK OR ATTEMPT TO MODIFY IT UNLESS INSTRUCTED TO DO SO.

This diskette should be copied to a MASTER diskette using the following procedure:

A. On 16ØØ-1 (dual-floppy) System:

1) Turn power on.

2) Insert ORIGINAL diskette in left-hand floppy drive (A:). MS-DOS will boot automatically.

3) Type (return) twice in response to DATE and TIME questions.

4) Type FORMAT B:/S, then type (return).

5) Insert a new diskette without a write-protect tab in the right-hand drive (B:) as requested on the screen, then type (return). The diskette will be formatted and the system will be transferred to the new diskette.

6) Respond with N to the "Format Another?" question, then type (return).

7) Type COPY *.* B: then type (return). All files will be copied to the new diskette. When complete, an "A:" prompt will appear.

8) Remove the ORIGINAL diskette from drive A: and store it in a safe place.

9) Move the MASTER diskette from drive B: to drive A:.

10) If you do not have a Serial Printer (connected to the Serial Port on the computer) and do not desire the RAM disk to be installed (256K memory board is required for the RAM disk to work), you are now ready to use your MS-DOS Master Diskette. To make additional back-ups proceed directly to Step 12.

11) To install the Serial Printer and/or the RAM disk on your system, type one of the following with the write-enabled MASTER diskette in drive A:

DEFINE 16ØØ-1S for serial printer only

or

DEFINE 16ØØ-1R for RAM disk only

or

DEFINE 16ØØ-1SR for both of the above

Type (return). When installation is done, "Function Complete" will be displayed.

12) The MASTER diskette is now ready to generate additional SYSTEM diskettes as required. Use the MASTER diskette in the above Steps 4 through 8. It is a wise practice to apply a write-protect sticker to the MASTER diskette at this time. Do not copy files to the MASTER diskette.

B. On 16ØØ-2, 16ØØ-3, 16ØØ-4 (hard disk) Systems:

1) Install computer as instructed in the MPC Operations Guide.

2) Turn power on.

3) Insert ORIGINAL diskette in floppy drive. MS-DOS will boot automatically.

4) Type (return) twice in response to DATE and TIME questions. An (A:) prompt will then be displayed.

5) Remove the ORIGINAL diskette from the floppy drive, remove and retain its write-protect sticker, then re-insert the diskette.

6) Type one of the following, depending on the model of the computer (see decal on rear of computer for model no.)

   DEFINE 16ØØ-2 (5 Megabyte Hard Disk)

   or

   DEFINE 16ØØ-3 (7.5 Megabyte Hard Disk)

   or

   DEFINE 16ØØ-4 (1Ø Megabyte Hard Disk)

   Type (return). System will display "Function Complete" when done.

7) Remove the diskette from the floppy drive, replace the write-protect sticker, then re-insert the diskette.

8) Hold down CTRL, ALT and DEL keys together to re-boot system. Type (return) twice in

response to DATE and TIME questions. An (A:) will then be displayed.

9) If the hard disk contains files which are not backed up on a floppy diskette, and you wish to retain them on the hard disk, type SYS B: and (return), then proceed directly to Step 11. If the computer is brand new, perform Step 1Ø first.

1Ø) Type FORMAT B:/H/S, then type (return). When ready, strike any key. Hard disk will require up to 1Ø minutes to format. Indicator on hard disk drive will remain lit during this time. Type (N) in response to "Format Another?" question when format is done.

11) Type COPY *.* B:, then type (return). When copy is complete, an (A:) will be displayed. If you do not have a Serial Printer (connected to the Serial Port on the computer) and do not desire the RAM disk to be installed (256K memory board is required for the RAM disk to work), do the following:

12) Type one of the following, depending on the model of the computer as determined in Step 6 above:

DEFINE 16ØØ-2 B:

or

DEFINE 16ØØ-3 B:

or

DEFINE 16ØØ-4 B:

You are now ready to use your MS-DOS system.

13) If you <u>do</u> wish to install a Serial Printer and/or a RAM disk, determine the model number of the MPC. (Refer to Step 6 above.) To install Serial Printer or RAM disk on your system, add suffix "S" (Serial Printer) or "R" (RAM disk) or "SR" (both) to the model number as in the following example:

DEFINE 16ØØ-2R

Remove the write-protect sticker from the diskette. Run the DEFINE program twice; once specifying drive "A:" and once specifying drive "B:". For example, on a 16ØØ-2 computer with RAM disk to be installed, type

<u>DEFINE 16ØØ-2R A:</u>, and type (<u>return</u>)

then type

<u>DEFINE 16ØØ-2R B:</u>, and type (<u>return</u>)

Place the write-protect sticker on the diskette.

14) A hard disk will not create a bootable floppy disk.

<u>NOTE:</u> All configurations set the Serial Port to 96ØØ baud, no parity, 2 stop bits, 7 data bits and expanded tabs. If this is not acceptable, the DEFINE program must be used in the menu-driven mode to further modify the system.

3.1.8 DEFINE
(ver. 3.ØØ for Columbia Data's MS-DOS Operating System)

The utility **DEFINE.COM** permits the user to configure the physical characteristics of the Columbia Data 16ØØ MPC. **DEFINE** may be run whenever new hardware is added or changes are required in the previous installations.

The program allows the user to modify the system configuration bytes in the boot block, which define the following aspects of the operating system:

(1) Disk drive capacities and locations
(2) Serial Port baud rate and byte format
(3) Existance of RAM disk
(4) Expansion of tabs to spaces on Serial Port
(5) Redirect PRN data to Serial Port

The new configuration specified will be saved in the boot block of the desired disk and will take effect when the system is booted from that disk.

The MS-DOS distribution disk is configured for the 16ØØ-1 Dual Floppy System. Baud rate is set to 96ØØ, serial format is no parity, 2 stop and 7 data bits. Tabs are expanded, there is no RAM disk, and the parallel printer is assumed. If you want to change the standard configuration to correspond with your system, execute DEFINE.COM to modify the boot block on the system disk.

DEFINE allows two modes of operation. It supports the **COMMAND** mode, where a previously defined system configuration may be installed on a specified disk by entering the configuration name and drive letter on the invoking line. (Some standard configurations are stored in the companion file **DEFINE.DAT**, which is also on the distribution disk.) The use of this command mode of **DEFINE** is also described in Section 3.1.7, Installing MS-DOS. **DEFINE** also supports the **MENU-DRIVEN** mode which permits the user to "custom tailor" a system to his specific needs.

PROGRAM OPERATION

The **COMMAND** mode is a direct way to define a system which has been previously configured and named. The **DEFINE** program requires the following syntax when used in the **COMMAND** mode:

A:define [configuration] [drive]

where [configuration] = a legal configuration name
[drive] = drive containing system to be DEFINED

The configurations supported by this release are:

1600-1	1600-2	1600-3	1600-4
1600-1S	1600-2S	1600-3S	1600-4S
1600-1R	1600-2R	1600-3R	1600-4R
1600-1SR	1600-2SR	1600-3SR	1600-4SR

where the model number of the computer (with no optional disk peripherals added) may contain suffixes "S" to enable redirection of PRN data to Serial Port (AUX) and "R" to enable Ram Disk as drive F:. Note that suffix "SR" is legal, but "RS" is not legal.

All configurations set the Serial Port to 9600 baud, no parity, 2 stop bits, 7 data bits and expanded tabs. If this is not acceptable, the **DEFINE** program must be run in the **MENU-DRIVEN** mode in order to modify the system. Once the new system configuration is defined, it can be named and added to the **DEFINE.DAT** file so that this configuration can be defined in the **COMMAND** mode.

An example:

A:define 1600-2s b:

This will modify the system on drive B: for the Model 16ØØ-2 computer with PRN data to be redirected to the Serial Port and the Ram Disk not installed. If drive B: is a floppy disk, the disk thus produced will boot any 16ØØ-2 computer (the computer which **DEFINE** is run on does not have to be a 16ØØ-2 in this case). If drive B is a hard disk, the computer must be a 16ØØ-2. Note that the companion file, **DEFINE.DAT**, must be present on the currently logged disk.

To use the newly-DEFINED system, it must be booted from the disk which has been modified.

To use the **MENU-DRIVEN** mode to define a new system configuration, type:

A:define

The program will display a description of its function. Type [RETURN] to continue.

The program will then ask for the standard model of your computer, if you are uncertain, see the decal on back of your computer. A list of all 16ØØ models is listed on screen. Select the letter corresponding to your model.

(a)	16ØØ-1	Dual Floppy System (MPC and VPC Models)
(b)	16ØØ-2	Floppy + 5 Megabyte Hard Disk
(c)	16ØØ-3	Floppy + 7.5 Megabyte Hard Disk
(d)	16ØØ-4	Floppy + 1Ø Megabyte Hard Disk

NOTE: Hard Disk drives are only used on the MPC

Enter Selection (a/b/c/d): a

If there are no disk expansion options added to your computer, enter "N" to the "Have any disk expansion options been added to the computer? (Y/N):" question. Otherwise, type "Y", then enter the total number of 1641 units (8 inch floppy drive) and the expansion hard disk model (if any) for your system. An example:

Have any disk expansion options been added to the computer? (Y/N): Y

How many 1641 (8 inch floppy drives) are there?

a) None
b) One
c) Two

Enter Selection (a/b/c): b

What is the model of expansion hard disk?

a) No expansion disk
b) 1642 (5 inch, 5 Megabyte Drive)
c) 1643 (5 inch, 10 Megabyte Drive)
d) 1644 (8 inch, 10 Megabyte Drive)
e) 1645 (8 inch, 20 Megabyte Drive)
f) 1646 (8 inch, 40 Megabyte Drive)

Enter selection (a/b/c/d/e/f): a

Next, enter the drive where the system is to be modified.

Example:

Where is system to be modified?

a) On Drive A:
b) On Drive B:

Enter selection (a/b): a

NOTE: if you have a 1600-1 system with a hard disk expansion, system to be modified can be on drive C:, the hard disk.

After all the information of your system is specified, the main menu will be displayed. Nothing has been changed on the system disk at this point.

```
--------------- MAIN MENU -------------

  a) Define a new system configuration

  b) Name the current configuration

  c) Exit DEFINE program
```

Select Function (a/b/c): a

You must select the (a) option to change the configuration ofyour system. DEFINE will display your current system and ask which aspect of the system you want to change.

Example:

```
THE SYSTEM CONFIGURATION IS:

 Drive assignments are:

   ...... IN COMPUTER ........     ....... EXPANSION
    floppy          floppy                 8" floppy
      A:              B:                      D:

 Baud rate is 9600
 Serial format is NO PARITY, 2 STOP , and 7 DATA bits.
 Tabs are EXPANDED to spaces for Serial Port
 PRN data is PASSED to Parallel Port.
 There is NO RAM DISK.
```

WHICH ASPECT OF SYSTEM DO YOU WANT TO CHANGE?

a) Baud Rate of Serial Port
b) Byte Format of Serial Port
c) Tabs set every 8 columns on Serial Port (on/off)
d) Redirect PRN data to Serial Port (on/off)
e) Ram disk (on/off)
f) All (or many) of the above
g) No changes ... Update system disk only
h) Return to MAIN MENU

Enter Selection (a/b/c/d/e/f/g/h): f

a) and b) Baud Rate and Byte Format defines the characteristics of the serial device, if your system is configured for a parallel printer, there is no need to change them.

Select the appropriate configuration for your serial printer by selecting option (a) and/or (b) in the configuration menu.

c) If you want tabs set every 8 columns on the Serial Port, set tabs to on.

d) PRN data can be sent to the Parallel or Serial Port. If you want to use the Serial Printer, then redirect PRN data to the Serial Port.

e) If a RAM disk is added to your system, select this option to set the RAM disk.

f) If you want to change many of the above aspects, this choice permits you to specify each aspect in sequence.

g) This function is provided to update the system disk (when no changes in a, b, c and d in the above aspects are necessary), but the model number and disk expansion options (specified earlier) have not been configured on the disk. (for example, the disk initially is defined for a basic 1600-1 system, but must be defined for a 1600-1 with 8" floppy).

h) Goes directly to the main menu without updating the system disk.

Example: Select 4800 baud, no parity, one stop and 7 data bits, Pass tabs to the Serial Port, redirect PRN data to Serial Port and install a RAM disk.

In the current system, Baud rate is 9600

Select desired baud rate:

a) 19,200 baud
b) 150 baud
c) 300 baud
d) 600 baud
e) 1200 baud
f) 2400 baud
g) 4800 baud
h) 9600 baud

Enter baud rate (a/b/c/d/e/f/g/h): g

In the current system, Serial format is NO PARITY, 1 STOP and 8 DATA bits.

Select one of the following:

a) No parity, 1 stop bit, 7 data bits
b) No parity, 2 stop bit, 7 data bits
c) No parity, 1 stop bits, 8 data bits
d) No parity, 2 stop bits, 8 data bits
e) Odd parity, 1 stop bit, 7 data bits
f) Odd parity, 2 stop bit, 7 data bits

g) Odd parity, 1 stop bits, 8 data bits
h) Odd parity, 2 stop bits, 8 data bits
i) Even parity, 1 stop bit, 7 data bits
j) Even parity, 2 stop bit, 7 data bits
k) Even parity, 1 stop bits, 8 data bits
l) Even parity, 2 stop bits, 8 data bits

Select the desired format (a/b/c/d/e/f/g/h/i/j/k/l): a

In the current system, Tabs are expanded to spaces for Serial Port.

Do you want tabs to be expanded to spaces (set every 8 columns) on the Serial Printer (Y/N): N

In the current system, PRN data is PASSED to Parallel Port.

Do you want to redirect PRN data to Serial Port (Y/N): Y

In the current system, There is NO RAM DISK

Do you want RAM disk to be installed (Y/N): Y

After all the changes are made, DEFINE will display the new configuration for your system. If this is what you want, answer "Y" to the question "Do you accept the new configuration (Y/N):", otherwise DEFINE will return to the main menu without changing the boot block.

NOTE: This is the only point in the menu driven **DEFINE** program from which a new configuration can be written to disk.

If you want to save your new configuration for later use then select option (b) in the main menu to name your configuration (name should be up to 25 characters). Then next time, use the COMMAND mode to directly define this configuration on other disks.

NOTE: In hard disk systems, this program will read the boot from HDBOOT.SYS if the hard disk does not contain a legal boot sector. The main menu provides a manual loadboot function in case the hard disk contains a boot which must be revised (due to a new version of HDBOOT.SYS).

To activate a system which has been previously configured and named, the COMMAND mode is a direct way to define the system. This program requires the following syntax when used in the COMMAND mode:

```
A:define [configuration] [drive]
```

where [configuration] = a legal configuration name
[drive] = drive containing system to be DEFINED

The configurations supported by this release are:

16ØØ-1	16ØØ-2	16ØØ-3	16ØØ-4
16ØØ-1S	16ØØ-2S	16ØØ-3S	16ØØ-4S
16ØØ-1R	16ØØ-2R	16ØØ-3R	16ØØ-4R
16ØØ-1SR	16ØØ-2SR	16ØØ-3SR	16ØØ-4SR

where the model number of the computer (with no optional disk peripherals added) may contain suffixes "S" to enable redirection of PRN data to Serial Port (AUX) and "R" to enable Ram Disk as drive F:. Note that suffix "SR" is legal, but "RS" is not legal.

All configurations set the Serial Port to 96ØØ baud, no parity, 2 stop bits, 7 data bits and expanded tabs. If this is not acceptable, the **DEFINE** program must be run in the **MENU-DRIVEN** mode in order to modify the system. Once the new system configuration is defined, it can be named and added to the **DEFINE.DAT** file so that this configuration can be defined in the **COMMAND** mode.

An example:

A:define 16ØØ-2s b:

This will modify the system on drive B: for the Model 16ØØ-2 computer with PRN data to be redirected to the Serial Port and the Ram Disk not installed. If drive B: is a floppy disk, the disk thus produced will boot any 16ØØ-2 computer (the computer which **DEFINE** is run on does not have to be a 16ØØ-2 in this case). If drive B: is a hard disk, the computer must be a 16ØØ-2. Note that the companion file, **DEFINE.DAT**, must be present on the currently logged disk.

To use the newly-DEFINED system, it must be booted from the disk which has been modified.

3.2 Intraline Commands*

Intraline commands include the special editing functions and the control character functions: Only the special editing functions are discussed here. (See Section 2.3.1.2 in the MS-DOS Manual for more information on the control character functions.)

The special editing commands have been assigned to the Function keys that make the best use of a specific terminal keyboard. Therefore, each command is identified by a functional name rather than by a specific key, and each is configurable to a particular keyboard key code. A code has been given to each command for ease of reference during the examples which demonstrate the function. (For an application on a specific terminal, the codes should be replaced by the names of the specific terminal keys.) Table 3-3 summarizes the commands, codes, and functions.

Table 3-3. Special Editing Commands

Command	Code	Function	Key Sequence Assigned
Copy one Character	<C1>	Copy one character from the template to the new line	F1
Copy up to character	<CM>	Copy all characters from the template to the new line up to the character specified	F3
Copy template	<CT>	Copy all remaining characters in the template to the new line	F5

*Reprinted by permission of Microsoft, Inc.

Skip one character	<S1>	Do not copy (skip over) a character in the template	F2
Skip up to Character	<SM>	Do not copy (skip over) the characters in the template up to the character specified	F4
Quit input	<QI>	Void the current input; leave the template unchanged	F6
Insert mode	<INS>	Enter insert mode	F9
Replace mode	<REP>	Exit insert mode (toggle from insert); this is the default	F1Ø
New template	<NT>	Make the new line the new template	F7
Backspace	<CL-X>	Backspace 1 character (same as CTRL-H)	F8

3.3 MS-BASIC

To run MS-BASIC, bootstrap the MS-DOS system (see 3.1) then enter:

```
A:BASIC
```

MS-BASIC will then respond:

```
BASIC-86 Rev. 5.21
[86-DOS version]
Copyright 1977-1981 (c) by Microsoft
Created: 12 Feb-82
62367 Bytes free
OK
```

At this point, MS-BASIC is initialized.

3.3.1 Initialization*

The procedure for initialization will vary with different implementations of BASIC-8Ø**. Check the appropriate appendix at the back of the MS-BASIC manual to determine how BASIC-8Ø is initialized with your operating system.

3.3.2 Modes of Operation*

When BASIC-8Ø is initialized, it types the prompt "OK". "OK" means BASIC-8Ø is at command level, that is, it is ready to accept commands. At this point, BASIC-8Ø may be used in either of two modes: the direct mode or the indirect mode.

In the direct mode, BASIC statements and commands are preceded by line numbers. They are executed as they are entered. Results of arithmetic and logical

*Reprinted by permission of Microsoft, Inc.
**BASIC 86 is an updated version of BASIC 8Ø

operations may be displayed immediately and stored for later use, but the instructions themselves are lost after execution. This mode is useful for debugging and for using BASIC as a "calculator" for quick computations that do not require a complete program.

The indirect mode is the mode used for entering programs. Program lines are preceded by line numbers and are stored in memory. The program stored in memory is executed by entering the RUN command.

3.3.3 Line Format*

Program lines in a BASIC program have the following format (square brackets indicate optional):

nnnnn BASIC statement [:Basic statement...]
<carriage return>

At the programmer's option, more than one BASIC statement may be placed on a line, but each statement on a line must be separated from the last by a colon.

A BASIC program line always begins with a line number, ends with a carriage return, and may contain a maximum of:

72 characters in 8K BASIC-80
255 characters in Extended and Disk BASIC-80

In Extended and Disk versions, it is possible to extend a logical line over more than one physical line by use of the terminal's <line feed> key. <Line feed> lets you continue typing a logical line on the next physical line without entering a <carriage return>. (In the 8K version, <line feed> has no effect.)

*Reprinted by permission of Microsoft, Inc.

3.3.3.1 Line Numbers*

Every BASIC program line begins with a line number. Line numbers indicate the order in which the program lines are stored in memory and are also used as references when branching and editing. Line numbers must be in the range Ø to 65529. In the Extended and Disk versions, a period (.) may be used in EDIT, LIST, AUTO, and DELETE commands to refer to the current line.

3.3.4 Character Set*

The BASIC-8Ø character set is comprised of alphabetic characters, numeric character, and special characters.

The alphabetic characters in BASIC-8Ø are the upper case and lower case letters of the alphabet.

The numeric characters in BASIC-8Ø are the digits Ø through 9.

The following special characters and terminal keys are recognized by BASIC-8Ø:

Character	Name
	Blank
=	Equal sign or assignment symbol
+	Plus sign
-	Minus sign
*	Asterisk or multiplication symbol
/	Slash or division symbol
^	Up arrow or exponentiation symbol
(	Left parenthesis
)	Right parenthesis
%	Percent

*Reprinted by permission of Microsoft, Inc.

#	Number (or pound) sign
$	Dollar sign
!	Exclamation point
[	Left bracket
]	Right bracket
,	Comma
.	Period or decimal point
'	Single quotation mark (apostrophe)
;	Semicolon
:	Colon
&	Ampersand
?	Question Mark
<	Less than
>	Greater then
\	Backslash or integer division symbol
@	At-sign
_	Underscore
<rubout>	Deletes last character typed
<escape>	Escapes Edit Mode subcommands See Section 2.16 in the Microsoft BASIC Manual
<tab>	Moves print position to next tab stop. Tab stops are every eight columns.
<line feed>	Moves to next physical line
<carriage return>	Terminates input of a line

3.3.4.1 Control Characters*

The following control characters are in BASIC-8Ø:

Control-A Enters Edit Mode on the line being typed.

Control-C Interrupts program execution and returns to BASIC-8Ø command level.

Control-G Rings the bell at the terminal.

*Reprinted by permission of Microsoft, Inc.

Control-H	Backspace. Deletes the last character typed.
Control-I	Tab. Tab stops are every eight columns.
Control-O	Halts program output while execution continues. A second Control-O restarts output.
Control-R	Retypes the line that is currently being typed.
Control-S	Suspends program execution.
Control-Q	Resumes program execution after a Control-S.
Control-U	Deletes the line that is currently being typed.

NOTE: For complete information concerning the MS-BASIC program, refer to the Microsoft MS-BASIC Manual included in your MPC shipment.

SECTION 4: CP/M-86 OPERATING SYSTEM

4.1 General

The CP/M-86 program is similar in operation to the MS-DOS program in that it has a DIR (Directory), an A> (prompt), and utilities available for operator use.

4.2 CP/M-86 System Bootstrap

Use the CP/M-86 system disk to bootstrap the system. Please read Section 4.4.1 if your system contains a hard disk.

When the CP/M-86 Operating System is booted, it will display the following:

```
Columbia Data Products, Inc.
Boot Version 1.Ø
July 2, 1982
CP/M-86 v1.1 for the Columbia Data Products
Multi-Personal Computer
Copyright (c) 1982, Columbia Data Products, Inc.
CDP version 1.5, October 17, 1982
Hardware Configured:
Memory (Kb):        128
32Ø Kb floppy disks:  2

A>
```

4.3 DIRectory

At the A> (prompt), type in:

```
A> DIR ↵
|  |   └─Carriage Return
|  |     (Function Key)
|  └─Utility
└─Prompt
```

Upon pressing the carriage return key (↵), a DIR (Directory) listing similar to the following will appear on the video monitor:

Table 4-1. DIR (CP/M-86 Dual Floppy Drive)

```
A> DIR
A:DDT86     CMD:FORMAT   CMD:SUBMIT   CMD:STAT CMD
A:GENCMD    CMD:ASM86    CMD:ED       SMD:PIP  CMD
A:COPYDISK  CMD:BIBMPC   CMD:BCDPMPC  CMD:READ  ME
A:FUNCTION  CMD:LMCMD    CMD:LOADER   CMD:8Ø87 LIB
A:HELP      CMD:GENDEF   CMD:SPEED    CMD:TOD  CMD
A:RANDOM    A86:SINGLES  LIB:DEBLOCK  LIB:CPM
A:REDIRECT  CMD:COPYDISK CMD:
```

Table 4-2. DIR (CP/M-86 Hard Disk Drive)

```
A> DIR
A:DDT86     CMD:FORMAT   CMD:SUBMIT   CMD:STAT      CMD
A:GENCMD    CMD:ASM86    CMD:ED       SMD:PIP       CMD
A:COPYDISK  CMD:BIBMPC   CMD:BCDPMPC  CMD:READ      ME
A:FUNCTION  CMD:LMCMD    CMD:LOADER   CMD:8Ø87      LIB
A:HELP      CMD:GENDEF   CMD:SPEED    CMD:TOD       CMD
A:RANDOM    A86:SINGLES  LIB:DEBLOCK  LIB:CPM
A:REDIRECT  CMD:COPYDISK CMD:BLDSYS86 CMD:HDLOADER  CMD
```

4.3.1 CP/M-86 Transient Utility Commands*

A program that executes a Transient Utility command comes into memory only when you request it. Section 5 in the Digital Research CP/M-86 User's Guide provides operating details for the standard CP/M-86 Utilties listed in the table below.

Table 4-3. CP/M-86 Utilities

COMMAND	MEANING
ASM86	translates 8Ø86 assembly language programs into machine code form.
COPYDISK	creates a copy of a disk that can contain CP/M-86, program files, and data files.
DDT86	helps you check out your programs and interactively correct "bugs" and programming errors.
ED	lets you create and alter character files for access by various programs.
GENCMD	uses the output of ASM-86 to produce an executable command file.
HELP	displays information on how to use each CP/M-86 command.
PIP	combines and copies files.
STAT	lets you examine and alter file and disk status, and assign physical I/O devices to CP/M-86 logical devices.

*Reprinted by permission of Digital Research

SUBMIT	sends a file of commands to CP/M-86 for execution.
TOD	sets and displays the system date and time.

4.3.2 FORMAT

FORMAT is a utility program that is invoked by typing "FORMAT" at the keyboard in response to the CP/M (A>) prompt. Its function is to format disks into data sectors. The data sectors are organized into tracks. In addition to the organizing the disk, the FORMAT utility program verifies that the format was actually recorded on the disk correctly by writing a pattern to each sector and then reading each sector from the disk.

WARNING

Any data previously placed on the disk being formatted is destroyed.

Floppy Disk Format

The FORMAT program (utility) is used to place the operating format on the first two tracks of the disk according to the following:

Type in <CR>

A>DIR

DIR is the acceptable input for the directory which gives a listing of the supplied utilities on the disk. From the DIR, choose the utility to be worked with; in this case FORMAT. At the A> (prompt), type FORMAT and depress the enter key. The following will appear on the display monitor:

A> FORMAT <CR>

FORMAT Utility v1.1

Do you wish to create a bootable disk? (Y/N) N

Place disk to be formatted in B: (Right Drive)
Confirm when ready (Press any key)

At this time, the "B" drive will become active and the drive activity light will be lighted. When the FORMAT procedure is completed, the following will be displayed:

Format done.

A>

To FORMAT a bootable disk, proceed as follows:

A> FORMAT

Format Utility v1.1

Do you wish to create a bootable disk? (Y/N) Y
Place disk to be formatted in B: (Right Drive)
Confirm when ready (Press any key)

At this time, the "B" drive will become active (which is audible) and the drive activity light will be lighted. When the FORMAT procedure is completed, the following will be displayed:

Format done.

A>

4.3.3 <u>COPYDISK</u>

The COPYDISK utility allows the copying of the entire contents of one disk to another. To utilize this function or utility, proceed as follows:

A> COPYDISK

CP/M-86 Full Disk Copy Utility
Version 2.Ø

Enter Source Drive (A-D)? <u>A</u>

Destination Disk Drive (A-D)? <u>B</u>

Copying Disk A: to Disk B:
Is this what you want to do (Y/N)?

WARNING

All information on the disk being copied to will be erased.

NOTE: A. If, at this point, you decide not to copy the disk or the wrong disk drive is entered for Destination Disk Drive, enter N and carriage return, and the COPYDISK function will abort and display "Copy aborted" on the video monitor. The monitor will then display:

Copy another disk (Y/N)? <u>N</u> (See NOTE:B)
COPY program exiting

A>

B. If at this point you decide to continue with copying the disk in drive A to drive B, enter Y, a carriage return, and the information on disk A will copy to disk B. The following procedure will take place:

Copying Disk A: to Disk B:
Is this what you want to do (Y/N)? Y
Copy Started
Reading Track (Decrements from 79-Ø)
Writing Track (Decrements from 79-Ø)
Verifying Track (Decrements from 79-Ø)
Copy Completed
Copy another disk (Y/N)? N (operator option)
COPY program exiting

A>

4.3.4 ERA (Erase)

The ERA command is a built-in command which does not show up on the DIR (Directory). This command can be used to remove one or more files from the directory of a disk. The ERA command should be used with care since all files that satisfy the file specification to be deleted (erased) will be removed from the disk directory. To utilize this command, proceed as follows:

A: ERA (File) ↵

- ↵ — Carriage Return (Function Key)
- (File) — File to be Erased from the DIR
- ERA — Erase command
- A: — Disk Drive Prompt

4.3.5 PIP

More formally, PIP stands for "Peripheral Interchange Program" which simply means that you can copy a file from one disk to another. However, PIP can be used for a number of other tasks such as renaming a file, combining files, creating a file from input, or transferring data from an input to an output device.

To transfer a file from one disk drive to another, proceed as follows:

A> PIP B:=A: FORMAT.CMD ↵

Carriage Return Key Function

File being transferred from drive A to drive B

Copy from drive A to drive B

Command

Disk Drive Prompt

NOTE: Refer to the CP/M-86 documentation supplied with your MPC (if the CP/M-86 operating system was ordered) for a comprehensive discussion of PIP.

4.3.6 HELP

The utility "HELP" will display a modified directory of utilities and a step-by-step walk through of each. To exit the HELP Utility, use CTRL C; depress the CTRL key and then the "C" key.

4.3.7 FUNCTION

This utility allows the user to program a line of instructions that will be called up when the appropriate Function Key is selected.

To invoke this utility, type the word FUNCTION at the system prompt. After clearing the screen, the new screen will appear with the cursor located next to the box containing F1. The instructions for each action are displayed only when the previous action is complete, and error messages are displayed then removed after a short time allowed for reading.

An example, defining key F3.

A> FUNCTION <CR>

At this point, pressing the down arrow once positions the cursor at F3. Now press END. The old instructions are replaced and the input line is displayed. Now, type STAT A:*.* (lower case is OK), press END again, the screen clears, and the key is defined.

Any time a key is pressed that is defined as "not allowed," an error message is displayed in the lower right area of the screen. Also, if characters have to be removed, use the left arrow to backspace over them. If the maximum characters are approached, 18 characters - one being the automatically appended <CR>, the error message will be displayed. In this case, use the left arrow to remove the characters up to the beginning of the line if required.

4.3.8 REDIRECT

The program REDIRECT allows you to assign the logical devices to one or more physical devices. For example, to assign the list device to a serial printer after CP/M is booted (the parallel lst device is assigned at boot time).

CP/M-86 defines four logical devices CON:, AXI:, AXO:, and LST:. REDIRECT uses these devices when assigning physical devices. The table below lists each physical device and the type of operation it performs, and in quotes, the name of the connector.

Keyboard	Input	
Screen	Output	
Serial Port #Ø	Input and Output	"SERIAL PORT"
Serial Port #1	Input and Output	"CONSOLE PORT"
Printer Port #Ø	Output	"PARALLEL PORT"
Printer Port #1	Output	"SERIAL PORT"
Printer Port #2	Output	"CONSOLE PORT"
Dummy/Device	Output	

A logical device can obtain input from one source, but a logical output device can be directed to one or all physical output devices. For instance, input can be received by the system from the keyboard then, if there are two video terminals attached to the serial I/O ports, data can be sent to both.

```
A> REDIRECT
I/O Redirection Utility v1.Ø September 29, 1982
Copyright (c) 1982 Columbia Data Products, Inc.


                         TYPES:

           Console          Input/Output (1)
           List             Output       (2)
           Auxilliary       Input/Output (3)


SELECT    2


                      Output Devices:

               Screen                 (1)
               Serial Port #Ø         (2)
               Serial Port #1         (3)
               Printer #Ø             (4)
               Printer #1             (5)
               Printer #2             (6)
               Dummy Device           (7)


SELECT    5


Input [    ] Output [Printer #1] Device [LIST]
OK?  <Y or N>  Y
More?  <Y or N>  N

A>
```

Figure 4-1. REDIRECT

4.3.9 SPEED

The SPEED program allows you to set or change the serial Port #Ø and serial Port #1 communication attributes. All allowable attributes are contained in the program menus (Figure 4-2). The table below shows the attributes set at boot time.

Baud Rate	96ØØ
Parity	none
Stop Bits	1
Word Length	8 Bits

```
A>SPEED

Serial Port Utility v1.Ø September 28, 1982
Copyright (c) 1982 Columbia Data Products, Inc.


                      Baud Rate:
                      192ØØ  (1)
                       96ØØ  (2)
                       48ØØ  (3)
                       24ØØ  (4)
                       12ØØ  (5)
                        6ØØ  (6)
                        3ØØ  (7)
                        15Ø  (8)


SELECT   2


                 Parity:   none  (1)
                           even  (2)
                            odd  (3)


SELECT   2

                 Stop Bits:  one  (1)
                             two  (2)


SELECT   2

              Word Length:  7 bits  (1)
                            8 bits  (2)
```

Figure 4-2. SPEED Select Options

```
SELECT   2

                  Serial Port:  Ø  (1)
                                1  (2)

SELECT   2


Baud [96ØØ] Parity [even] Stop Bits [2] Word [8 bits]
Port #1 [1]
OK? <Y or N>  Y
Do the other port?  <Y or N>  N
A>
```

Figure 4-2. SPEED Select Options (continued)

4.4 Hard Disk Formatting

In the event it becomes necessary to build or rebuild hard disk software information, utilize the following procedure.

Table 4-4. Instructions for Building/Rebuilding Hard Disk Software

1. Reset the unit by pressing the reset button on the back panel.

2. Insert the distribution floppy labeled "CP/M-86" into the floppy drive A.

3. Type "FORMAT" on the keyboard, and the disk activity light on the floppy drive should light indicating the drive is in operation. The following question/operator response will appear on the CRT.

   FORMAT UTILITY v1.1A [Floppy/Winchester]
   Do you want Hard Disk Format [Y or N]
   Do you really want to do this [Y or N]

   When the format procedure is finished, the following will appear:

   FORMAT DONE

The format program takes 7 to 15 minutes depending on the capacity of the disk and destroys any data previously recorded on the hard disk. When the formatting is finished, FORMAT DONE will print on the terminal screen.

4. Put the CP/M-86 system tracks on the hard disk. To do this, type BLDSYS86 <CR>. The CP/M prompt (A>) will display when this function is complete.

5. At this time, RESET the unit again to boot CP/M-86 from the hard disk.

6. When CP/M-86 is finished booting from the hard disk, copy the CP/M-86 disk programs to the hard disk by typing the following response to the A> prompt.

   B:PIP A:=B:*.*[VR]

   Note: Though "B" is represented, the floppy may be "C" "D", etc., depending upon the system configuration.

7. After all the distribution files have been transferred to the hard disk, CP/M-86 should boot-up from the hard disk.

4.4.1 Hard Disk Boot

Booting from the hard disk is automatic when the system contains a hard disk drive.

NOTE: Do not boot from the floppy disk unless it is necessary to build/rebuild hard disk software. The CP/M floppy disk is provided only for hard disk rebuild and recovery. When it is necessary to boot from the floppy disk for hard disk rebuild, use utilities BLDSYS86 and HDLOADER and refer to Table 4-4.

4.5 Escape Sequences

Escape Code Sequences are not available from the Keyboard directly; they are, however, available under program control in whatever programming language is being used, such as FORTRAN, COBOL, BASIC, OR CBASIC. Escape Sequences can be designated to control functions such as change display colors, and to program the function keys (F1 through F1Ø) and the cursor keypad keys. The ASCII escape character "Hex 1B" allows the programmer to use this function to perform certain steps. To indicate which function is to be performed, (for example: "ESC a" Set Console Mode) the key used to activate that function ("a") must follow directly after the escape character 1B Hex (ESC). Escape character sequences are sent directly to the console output by the operating system.

Table 4-5. CP/M-86 Escape Codes

ESC a	Set Console Mode
ESC b	Set Foreground Color
ESC c	Set Background Color
ESC d	Redirect Console Input
ESC e	Redirect Console Output
ESC f	Redirect Auxiliary Output
ESC g	Redirect Auxiliary Input
ESC h	Redirect List Output
ESC i	Enable/Disable Transparent Mode
ESC j	Save Cursor Position
ESC k	Restore Cursor Position
ESC l	Enable/Disable Console Status Mode
ESC A	Cursor Up
ESC B	Cursor Down
ESC C	Cursor Forward
ESC D	Cursor Backward (Non-destructive)
ESC E	Clear Screen
ESC H	Home Cursor
ESC K	Clear to End of Line
ESC Y	Position Cursor
ESC /	Set Color Palette
ESC ?	Get Time, Date, Background Message
ESC :	Program Function/Cursor Keys
ESC ESC	Display Escape Graphics Character

The following paragraphs will give the operator a brief description of the escape functions provided in the MPC.

A. ESCa Set Console Mode
Escape "a" sets the Console Mode which selects the monochrome or color display. The Console Mode also selects the number of columns and rows on the CRT screen. The ESCa sequence must be followed by a one digit number from 0 to 7 which selects a mode according to the following table:

0 - 40 X 25 Black and White
1 - 40 X 25 Color
2 - 80 X 25 Black and White
3 - 80 X 25 Color
4 - 320 X 200 Color Graphics
5 - 320 X 200 Monochrome Graphics
6 - 640 X 200 Monochrome Graphics
7 - 80 X 25 Monochrome

B. ESCb Set Foreground Color
Escape "b" sets the Foreground Color which displays each character. The ESCb sequence must be followed by a selection character which determines the color as indicated below:

Control Byte Bit Pattern

```
7 6 5 4 3 2 1 0
        | | | └─Blue
        | | └───Green
        | └─────Red
        └───────Hi-intensity
```

C. ESCc Set Background Color
Escape "c" selects the Background Color. This function can also be utilized to cause characters to blink off and on. As with ESCb, ESCc must be followed by a selection character which determines the color as indicated below:

```
7 6 5 4 3 2 1 0
        | | | |___ Blue
        | | |_____ Green
        | |_______ Red
        |_________ Blink
```

Blink can be used with the following colors:

h - black	k - blue + green	n - red + green
i - blue	l - red	o - red + green + blue
j - green	m - red + blue	

D. ESCd,e,f,g,h Redirect Commands
The escape letters (d,e,f,g,and h) followed by two bytes, redirect the I/O between physical and logical devices

Escape Function
d - Console Input
e - Console Output
f - Auxiliary Input
g - Auxiliary Output
h - List Output

The most significant bit recognized by CP/M-86 should be set to one to prevent any fluctuation between tab and carriage return character values.

When using I/O redirection, remember that only a single Input Source can be specified while several output sources can be specified. Additionally, a set of bits is viewed as two bytes when trying to comprehend the workings of I/O redirection.

BYTE 1		BYTE 2		Physical Device
Binary	Hex	Binary	Hex	
10000001	81H	10000000	80H	Keyboard
10000010	82H	10000000	80H	Screen
10000100	84H	10000000	80H	Serial Port 0
10001000	88H	10000000	80H	Serial Port 1
10010000	90H	10000000	80H	Printer 0 (Parallel)
10100000	A0H	10000000	80H	Printer 1 (0)
11000000	C0H	10000000	80H	Printer 2 (1)
10000000	80H	10000001	81H	Light Pen
10000000	80H	10000010	82H	Reserved for Cassette I/O
10000000	80H	10000100	84H	Reserved for Game Card I/O
10000000	80H	10001000	88H	Dummy Device

E. ESCi Enable/Disable Transparent Mode

Escape "i" is used to enable/disable the transparent mode. Enabling the transparent mode will output to the CRT monitor the following:

Carriage Return
Backspace
Line Feed
Bell

Disabling the transparent mode will cause the above functions to occur. The transparent mode must be enablied when you want the special symbols to appear on the CRT monitor that are assigned to these functions.

Values for the least significant bit following the Escape "i" are as follows:

0 - Disable
1 - Enable

F. ESCj Save Cursor Position
Escape "j" saves the present cursor position.

G. ESCk Restore Cursor Position
Escape "k" returns the cursor to a previously saved (ESCj) position.

H. ESCl Enable/Disable Console Status Mode
Escape "l" causes the special feature in the console status to be enabled/disabled whether the console status reports any keyboard entry characters waiting. When enabled logical input is waiting for keyboard input. When disabled, the console status reports keyboard as well as internal character strings.

I. ESC A Cursor Up
Escape "A" will move the cursor up one line. Escape "A" will not affect the cursor if it is already on the top line.

J. ESC B Cursor Down
Escape "B" will move the cursor down one line. Escape "B" will not affect the cursor if it is already on the bottom line.

K. ESC C Cursor Forward
Escape "C" moves the cursor one position to the right but will not move the cursor off the screen.

L. ESC D Cursor Backward
Escape "D" moves the cursor one position to the left but will not move it off the screen. When moving the cursor to the left, existing characters will not be erased.

M. ESC E Clear Screen
Escape "E" will clear the screen and move the cursor to the upper left portion of the screen.

N. ESC H Home Cursor
Escape "H" moves the cursor to the upper left corner of the screen but DOES NOT clear the screen.

O. ESC K Clear to End of Line
Escape "K" will clear the line of text from the cursor position to the end of the line.

P. ESC Y Position Cursor
Escape "Y" in conjunction with numerical characters following the ESC "Y" dictate the row and column for cursor placement. The first number (Ø to23) indicates the row, the second number (Ø to 79) indicates the column. To prevent row and column confusion, 2Ø Hex has been added to each value.

Q. ESC / Set Color Palette
Escape "/" in conjunction with a following character will set the monitor display color palette.

R. ESC ? Get Time, Date, Background Message
The Escape "?" will cause a string of characters to be placed into the console input stream exactly as follows:

MM/DD/YY,HH:MM:SS,...blanks...

S. ESC: Program Function Keys
Escape ":" programs the ten function keys (F1-F1Ø) and the cursor control keys on the keyboard (see Figure 2-3).

Example:

ESC: Key Input ØØH

Hex Ø Terminates Line

Arbitrary Characters

a. Up to 18 characters for Function Keys

b. Up to 4 characters for Cursor Control Keys

Function/Cursor Control Key to be programmed (Table 4-5)

Escape Command

Table 4-6. ESC : Default Settings

Key Identifiers	Default Settings
; - F1	dir <cr>
< - F2	dir b:<cr>
= - F3	stat <cr>
> - F4	stat b:<cr>
? - F5	pip <cr>
@ - F6	pip b:=a:*.*[v]
A - F7	stat *.*<cr>
B - F8	stat b:*.*<cr>
C - F9	(not programmed)
D - F10	(not programmed)
G - Home	Esc H (Home)
H - Up Arrow	Esc A (Cursor Up)
I - Page Up	(not programmed)
K - Left Arrow	ESC D (cursor left)
M - Right Arrow	ESC C (cursor right)
O - End	'END'
P - Down Arrow	ESC B (cursor down)
Q - Page Down	(not programmed)
R - Ins	(not programmed)
S - Del	DEL (ASCII delete)

<cr> = Enter Key

4.6 Light Pen

The Light Pen acts as a probe in conjunction with the user developed software. It can be used for designing electrical circuits, mechanical design, architectural design, etc.

The Light Pen is activated by pressing the tip of the Pen against the desired area on the monitor screen. The activation of the Pen by pressing against the CRT in conjunction with the user's operational program allows the user to perform functions as dictated by the program.

The Light Pen can be used for a number of applications limited only by the programmer.

The Light Pen connects to J3 on the CDP video board. When connecting the Light Pen to the video board, be sure the unit is turned OFF.

WARNING

Electrical Hazard.

Remove the two mounting screws at the back of the unit (Figure 1-3) and slide the cover forward about half way, leaving enough room to connect the Light Pen to the video board.

If there is no data waiting when input is requested from the Light Pen by the program, the message

"Waiting for Light Pen Input"

will appear on the CRT monitor. When data is received, the data line is reset and the data from the Light Pen is placed into the Console status stream.

Format for console input from the Light Pen is as follows:

Byte 1 - Character Row + 2ØH
Byte 2 - Character Column + 2ØH
Byte 3 - Dot Row + 2ØH
Byte 4 - Most Significant 6 Bits of Dot Column + 2ØH
Byte 5 - Least Significant 6 Bits of Dot Column + 2ØH
Byte 6 - Terminator, always ØØ(ASCII NUL)

Misinterpretation of characters can be avoided by the driver converting all binary values to graphic ASCII characters. This can be accomplished by adding 2ØH to each byte.

SECTION 5: THEORY OF OPERATION

5.1 General

Operation of the MPC is centered on the master printed circuit board, which defines the unit's capability and governs the activities of the computer. The key component of the main system board is the 4.77 MHz 8Ø88 processor, which is interconnected to all other components of the main board.

5.2 8Ø88 Processor*

The 8Ø88 Processor is designed with an 8-bit external data path to memory and I/O. The 8Ø88 is contained in a standard 4Ø-pin dual in-line package and operates from a single +5V power source.

The 8Ø88 Processor has a complement of eight 16-bit general registers. The general registers are subdivided into two sets of four registers each: the data registers and the pointer and index registers. The data registers are unique in that their upper and lower halfs are separately addressable. This means that each data register can be used interchangeably as a 16-bit register, or as two 8-bit registers.

The 8Ø88 Processor can accommodate up to 1,Ø48,576 bytes of memory in both minimum and maximum mode. From a storage point of view, the 8Ø88 memory spaces are organized as identical arrays of 8-bit bytes. Instructions, byte data, and word data may be freely stored at any byte address without regard for alignment thereby saving memory space by allowing code to be densely packed in memory.

The 8Ø88 Processor always accesses memory in bytes. Word operands are accessed in two bus cycles regardless of their alignment. Instructions also are fetched one byte at a time. Alignment of word operands does not affect the performance of the 8Ø88.

The 8Ø88 is designed for the multiprocessing environment. It has built-in features that help solve coordination problems that have discouraged multiprocessing system development in the past.

5.3 Main System Board

The main system board, measuring 9 3/4-inches by 2Ø-inches, fits horizontally on the base of the unit. The system board contains all the necessary connectors for attaching the desired peripheral equipment. External connectors are PORT 1 (Serial Port) and the Console Port, which are 25 pin connectors, and the Centronics Port which is a 37 pin connector (see Figure 1-3). There are also eight 62-pin expansion board connector slots for additional configurations (see Figure 1-2).

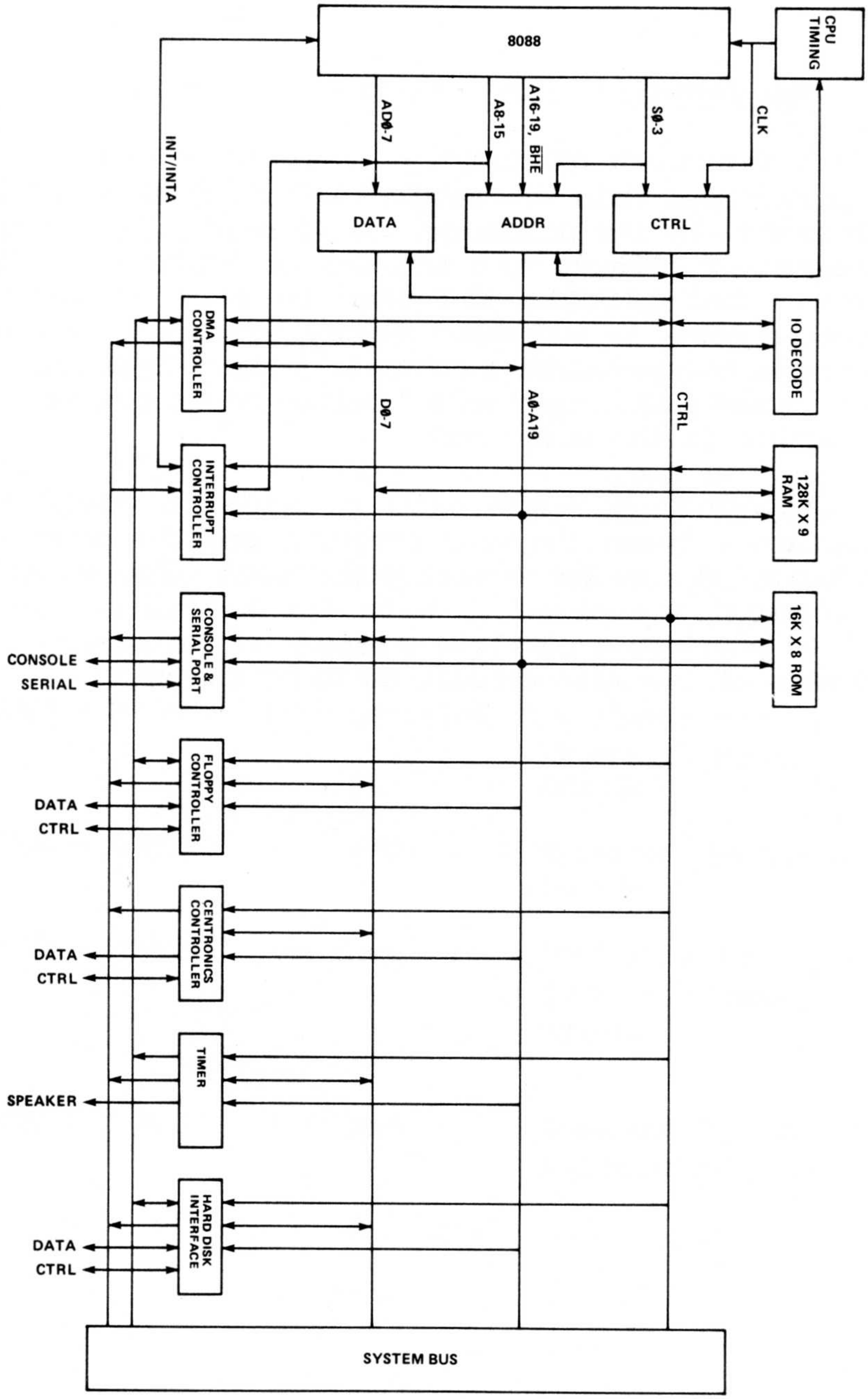


Figure 5-1. MPC System Block Diagram

5.4 Main Memory

The function of memory is to store data and instructions. Main memory is that which is directly addressable by the processor. In the MPC, this memory consists of 128K of RAM and 16K of EPROM. RAM is memory that permits the user to read and write computer words in addressed memory locations, whereas EPROM is a version of read-only memory that can be programmed and erased on a limited basis but is not accessible in the write mode.

Although the ROM BIOS provides a complete interface between the user programs and the computer hardware, it may be necessary for some user programs to communicate directly with the hardware. This section provides a complete summary of the memory and I/O maps of the main circuit board of the MPC.

Table 5-1. Memory Map, Main Circuit Board

Address Range (Hex)	Memory Type	Function
00000 - 00400	RAM	Interrupt vectors for 8088 CPU
00400 - 0047F	RAM	Reserved for ROM BIOS Variables
00480 - 0050F	RAM	Reserved for ROM Monitor Variables
00510 - 1FFFF	RAM	User Program/Operating System Area
20000 - AFFFF	Not Installed	Available for Expansion Memory Boards
B0000 - BFFFF	I/O	Reserved for Video Boards
C0000 - EFFFF	Not Installed	Available for Expansion Memory Boards
FC000 - FFFFF	ROM	Reserved for ROM BIOS and Monitor Code

Table 5-2. I/O Map, Main Circuit Board

Port (Hex)	Type	Function	Bit Definitions
INTEL 8237 DMA Controller:			
ØØØ	I/O	#Ø Base & Current Address (Refresh)	
ØØ1	I/O	#Ø Base & Current Word Count	
ØØ2	I/O	#1 Base & Current Address	
ØØ3	I/O	#1 Base & Current Word Count	
ØØ4	I/O	#2 Base & Current Address (Floppy Disk)	
ØØ5	I/O	#2 Base & Current Word Count	
ØØ6	I/O	#3 Base & Current Address (Note 6)	
ØØ7	I/O	#3 Base & Current Word Count	
ØØ8	Input	Status Register	Note 1
ØØ8	Output	Command Register	Note 1
ØØ9	Output	Request Register	Note 1
ØØA	Output	Mask Register	Note 1
ØØB	Output	Mode Register	Note 1
ØØC	Output	Clear Byte Pointer Flip/Flop	Note 1
ØØD	Input	Temporary Register	Note 1
ØØD	Output	Master Clear	Note 1
ØØF	Output	Write All Mask Register Bits	Note 1
Ø1Ø-Ø1F	---	Not Available	

Table 5-2. (Continued)

Port (Hex)	Type	Function	Bit Definitions
INTEL 8259 Interrupt Controller:			
Ø2Ø	I/O	Registers Requiring AØ=Ø	Notes 1,12
Ø21	I/O	Registers Requiring AØ=1	Notes 1,12
Ø22-Ø3F	---	Not Available	
INTEL 8253 Triple Timer Circit:			
Ø4Ø	I/O	System Tick Count Register	
Ø41	I/O	Memory Refresh Count Register	
Ø42	I/O	Tone Generator Count Register	
Ø43	Output	Mode Register	Note 1
Ø44-Ø5F	---	Not Available	
INTEL 8255 Parallel I/O Circuit:			
Ø6Ø	Input	Configuration Switches (1=OFF)	BØ=Bank 2 SW 1
			B1=Bank 2 SW 2
			B2=Bank 2 SW 3
			B3=Bank 2 SW 4
			B4=Bank 2 SW 5
			B5=Bank 2 SW 6
			B6=Bank 2 SW 7
			B7=Bank 2 SW 8
Ø61	Output	MPC Control	BØ=Tone Enable
			B1=Speaker Enable

Table 5-2. (Continued)

Port (Hex)	Type	Function	Bit Definitions
			B2=Not Used B3=Not Used B4=Parity Disable B5=I/O CHK Disable B6=Keyboard Enable B7=Note 2
Ø62	Input	Keyboard Data/ Switches Bit Definitions Shown Apply To "Switches" Case (Port Ø61, Bit 7=1)	BØ=Bank 1 SW 1 B1=Bank 1 SW 2 B2=Bank 1 SW 2 B3=Bank 1 SW 2 B4=Undefined B5=Undefined B6=I/O CHK Error B7=Parity Error
Ø63	Output	Control Register	Note 1
Ø64-Ø8Ø	---	Not Available	

Miscellaneous Ports:

Port (Hex)	Type	Function	Bit Definitions
Ø81	Output	#2 DMA Page	Note 3
Ø82	Output	#3 DMA Page	Note 3
Ø83-Ø9F	---	Not Available	
ØAØ	Output	NMI Enable	B7=NMI Enable
ØA1-ØBF	---	Not Available	

Table 5-2. (Continued)

Port (Hex)	Type	Function	Bit Definitions
INTEL 8255 Parallel I/O Circuit:			
ØCØ	Input	Printer, Hard Disk Status	BØ=Note 4 B1=Disk Not Ready B2=Note 5 B3=Not Used B4=Not Used B5=Not Used B6=Printer Busy B7=PRNTR Not Fault
ØC1	Output	Printer Data Port	
ØC2	I/O	Printer Status, Misc. Control	BØ=IRQ5 B1=Not Used B2=PRNTR Not ACK (In) B3=Not Used B4=Disable IRQ5 B5=PRNTR Not Strobe B6=Note 6 B7=Note 6
ØC3	Output	Control Register	Note 1
ØD4-ØDF	---	Not Available	
Hard Disk Data Port:			
ØEØ	I/O	Hard Disk Data Port (Direction is Controlled By Hard Disk Controller and Is Indicated On Port ØCØ Bit Ø.)	
ØE1-ØFF	---	Not Available	

Table 5-2. (Continued)

Port (Hex)	Type	Function	Bit Definitions
National INS 825Ø Rear Panel Console Port:			
2F8	Input	Receiver Buffer Register	
2F8	Output	Transmitter Holding Register	
2F9	Output	Interrupt Enable Register	Note 1
2FA	Input	Interrupt Ident. Register	Note 1
2FB	Output	Line Control Register	Note 1
2FC	Output	MODEM Control Register	Note 1
2FD	Input	Line Status Register	Note 1
2FE	Input	MODEM Status Register	Note 1
2FF	Output	Divisor Latch (1.8432 Mhz)	
INTEL 8272 Floppy Disk Controller:			
3F2	Output	Write Control Register This Port is Not Part of the 8272 Device.	BØ=Note 7 B1=Note 7 B2=Not Reset FDC B3=Enable DMA B4=DRV Ø Motor On B5=DRV 1 Motor On B6=DRV 2 Motor On B7=DRV 3 Motor On
3F4	Input	Main Status Register	Note 1
3F5	I/O	Data Register	Note 1

Table 5-2. (Continued)

Port (Hex)	Type	Function	Bit Definitions
National INS 825Ø Rear Panel Serial Port:			
3F8	Input	Receiver Buffer Register	
3F8	Output	Transmitter Holding Register	
3F9	Output	Interrupt Enable Register	Note 1
3FA	Input	Interrupt Ident. Register	Note 1
3FB	Output	Line Control Register	Note 1
3FC	Output	MODEM Control Register	Note 1
3FD	Input	Line Status Register	Note 1
3FE	Input	MODEM Status Regster	Note 1
3FF	Output	Divisor Latch (1.8432 Mhz)	

Memory Map, Color Graphics Video Board:

Address Range (Hex)	Memory Type	Function
B8ØØØ - BBFFF	RAM	Video Display Buffer

5.5 Direct Memory Access (DMA) Controller

The DMA controller is located on the system board, allowing high speed transfers of program and data bytes between memory and I/O peripherals. The DMA controller has 4 independent channels, allowing the system to perform up to 4 block transfers concurrently. In the MPC, Channel Ø is used to perform dynamic RAM refreshing. This incurs an overhead of approximately 7% of the system processing time. Channels 1-3 are available for use by the system. Channel 2 is used intermittently for floppy disk DMA transfers and Channel 3 is used for transfers to and from the Winchester controller for hard disk systems.

Three modes of operation are possible with the Direct Memory Access circuit: the byte-at-a-time mode in which control is returned to the processor after each one-byte cycle; the burst mode in which operation continues as long as the ports are ready; the continuous mode in which the processor does not regain control until the operation is completed.

5.6 Color Graphics Video Board Programming Reference

This section provides information required for programming the color graphics video board. It can be used in conjunction with the I/O port map for this board.

5.6.1 Initialization

Reset the video enable bit of the mode register (Port 3D8, Bit 3). Next, program the CRT controller chip (Ports 3D4 and 3D5) for the desired operation. Finally, program the mode and color registers (Ports 3D8 and 3D9).

5.6.2 Mode Register Programming

Video Mode	Mode Byte
40 X 25 Alpha B & W	2CH
40 X 25 Alpha Color	28H
80 X 25 Alpha B & W	2DH
80 X 25 Alpha Color	29H
320 X 200 Graphics B & W	0EH
320 X 200 Graphics Color	0AH
640 X 200 Graphics B & W	1EH

Note: To control high intensity background using the blinking attribute in alpha modes, clear Bit 5 of the mode byte. Since Bit 3 of all mode bytes is set, the video will be enabled when the mode is programmed.

5.6.3 Color Register Programming

Bit 0 - Selects blue as a border in 40 X 25 alpha mode or as a background color in 320 X 200 color graphics mode.

Bit 1 - Selects green as a border color in 40 X 25 alpha mode or as a background color in 320 X 200 color graphics mode.

Bit 2 - Selects red as a border color in 40 X 25 alpha mode or as a background color in 320 X 200 color graphics mode.

Bit 3 - Intensifies the border color in 40 X 25 alpha mode or background color in 320 X 200 color graphics mode.

The above bits define the following colors:

ØØØØ = Black	Ø11Ø = Brown	11ØØ = Lt Red
ØØØ1 = Blue	Ø111 = Lt Gray	11Ø1 = Lt Magenta
ØØ1Ø = Green	1ØØØ = Dk Gray	111Ø = Yellow
ØØ11 = Cyan	1ØØ1 = Lt Blue	1111 = White
Ø1ØØ = Red	1Ø1Ø = Lt Green	
Ø1Ø1 = Magenta	1Ø11 = Lt Cyan	

Bit 4 - Intensifies the background colors in the alpha mode.

Bit 5 - Selects the pallette for 32Ø X 2ØØ color graphics mode. The pallette defines the color produced by each value of the pixel (C1, CØ) for that point of the screen.

If Bit 5 = Ø

C1	CØ	Color
Ø	Ø	Background Color
Ø	1	Cyan
1	Ø	Magenta
1	1	White

If Bit 5 = 1

C1	CØ	Color
Ø	Ø	Background Color
Ø	1	Green
1	Ø	Red
1	1	Brown

5.6.4 Status Register Interpretation

The status register (Port 3DA) of the color graphics video board is used for video buffer access timing and light pen testing.

Bit Ø - This bit is high when the video display is blanked during a sync interval. In 8Ø X 25 alpha mode, the video buffer should not be accessed by the CPU unless this bit is high or the access will produce interference in the form of a short horizontal line on the display.

Bit 1 - Indicates that a rising edge from the light pen switch has set the light pen trigger. This status may be cleared or set by outputting any data byte to Ports 3DB or 3BC respectively.

Bit 2 - If Ø, indicates the light pen switch is on. Switch is not debounced or latched.

Bit 3 - This signal is high during the vertical sync interval of the display.

5.6.5 Video Buffer Organization

The video buffer consists of 16K bytes of RAM located at B8ØØØ-BBFFF. The format of data in the memory is different for each type of display mode.

8Ø X 25 Alpha Modes

Each character position is defined by two consecutive bytes in the video buffer. The first byte is the ASCII character code, and the second byte is the attribute. The screen holds 2ØØØ characters which requires 4,ØØØ bytes. Consecutive locations in the buffer correspond to adjacent characters on a line with the first character in the buffer located at the upper left corner of the screen. The 256 character patterns are generated by a 2K byte ROM on the video board. It defines the character as a 7 X 7 pattern in an 8 X 8 box. The attribute byte is defined as follows:

Bit Ø - Blue Character	Bit 4 - Blue Background
Bit 1 - Green Character	Bit 5 - Green Background
Bit 2 - Red Character	Bit 6 - Red Background
Bit 3 - Intensify Character	Bit 7 - Blinking Character

Bit 7 can be redefined to intensify background by setting Bit 5 of the mode register. For black and white modes, select black by turning all colors off and white by turning all colors on.

4Ø X 25 Alpha Modes

These modes are similar to the 8Ø X 25 alpha modes except that the screen only holds 1,ØØØ characters and therefore requires 2,ØØØ bytes.

32Ø X 2ØØ Graphics

This mode uses 16,ØØØ bytes of memory, organized a 4 pixels horizontally per byte. The screen is divided into two buffers, the first holding data for the even scan lines (Ø, 2, 4, . . . , 198) at addresses B8ØØØ-B9F3F and the second holding data for the odd scan lines (1, 3, 5, . . . , 199) at addresses BAØØØ-BBF3F. Each pixel is defined by 2 bits, with the top left corner of the screen being B7 and B6 of the first byte. Each line consists of 32Ø pixels, which is defined by 8Ø consecutive bytes in the buffer. The color of the displayed pixel is selected from the current pallette by the value of the bits (C1, CØ; where C1 is the higher order bit). See the description of Bit 5 in the color register programming for a definition of the pallettes. The black and white mode is similar to color, except that the color burst is disabled.

64Ø X 2ØØ Black and White Graphics

This mode is similar to 32Ø X 2ØØ color graphics mode except that each bit represents 1 pixel on the screen and there are twice as many pixels per line. Therefore, a line is represented by 8Ø consecutive bytes in the buffer as in 32Ø X 2ØØ mode. Black is encoded as zero and white as one. The upper left corner of the screen is B7 of the first byte of the buffer.

5.6.6 Character Generation in Graphics Modes

A table of the first 128 characters similar to the alpha mode character generator is located at address FFØØH:ØA6EH in the ROM. Each character is defined by 8 consecutive bytes, representing the LIT pixels for each row of the character from top to bottom. The most significant bit represents the left-hand dot of a row. When the data is transferred to the video memory to be displayed, it must be split into even and odd row data and written to the appropriate buffers. Any other characters which are to be displayed (including those corresponding to Codes 128 - 255 in alpha mode) must be defined by the user. These patterns will be accessed as codes 128 - 255 by the ROM's INT IØH graphics character routines if the user points to the table with interrupt vector 1FH (Appendix D).

5.6.7 Standard Initialization for 6845 CRT Controller Circuit

The CRT controller is intialized for a given display mode by outputting a string of 16 bytes to its registers Ø - 15 respectively. For each byte to be output, the register number must first be output to the address register (Port 3D4), followed by the data byte to the desired register (Port 3D5). The strings used by the ROM for the 4Ø X 25 alpha, 8Ø X 25 alpha and graphics modes are given in the description of INT 1ØH (see Appendix D.4).

Table 5-3. I/O Map, Color Graphics Video Board

Port (Hex)	Type	Function	Bit Definitions
Motorola 6845 CRT Controller Circuit:			
3D4	Output	Address Register	Note 1
3D5	I/O	18 Registers	Note 1
Miscellaneous Ports:			
3D8	Output	Mode Register	B0=80X25 Mode
			B1=320X200 Graphics
			B2=B&W Mode
			B3=Enable Video
			B4=640X200 B&W
			B5=Note 8
			B6=Not Used
			B7=Not Used
3D9	Output	Color Register	B0=Blue (Note 9)
			B1=Green (Note 9)
			B2=Red (Note 9)
			B3=Note 10
			B4=Alt. Back Colors
			B5=320X200 Color Set
			B6=Not Used
			B7=Not Used
3DA	Input	Status Register	B0=Video Sync
			B1=Light Pen Trig'd.
			B2=Pen Switch Off
			B3=Alpha Data
			B4=Not Used
			B5=Not Used
			B6=Not Used
			B7=Not Used

Table 5-3. (Continued)

Port(Hex)	Type	Function	Bit Definitions
3DB	Output	Clear Light Pen Latch	Note 11
3DC	Output	Set Light Pen Latch	Note 11

NOTES FOR TABLES 5-2 AND 5-3

Note 1: Please refer to component manufacturer's specifications for details.

Note 2: This bit selects keyboard data (if Ø) or switch bank and parity error bits (if 1) for input at Port Ø62.

Note 3: The low 4 bits of this port represents the high 4 bits of the memory address to be accessed by the DMA channel. The DMA cannot cross a 64K memory boundary during a transfer.

Note 4: This bit indicates direction of hard disk data port (ØEØ) when hard disk ready (Port ØCØ Bit 1) is true. If Ø, data port (ØEØ) is output.

Note 5: This bit is connected to the console port receive data signal and is used for measuring the baud rate of the terminal.

Note 6: Bits 7 and 6 on Port ØC2 select the device to be attached to DMA Channel #3. If Bits 7,6 = 1,1 Channel #3 is connected to the DREQ3 and DACK3 on the external bus. if Bits 7,6 = 1,Ø Channel #3 is connected to the hard disk port.

Note 7: Bits 1,Ø are the binary value of drive selected.

Note 8: Setting this bit in alpha modes defines Bit 7 of the attribute byte as a blinking select. Otherwise, attribute Bit 7 defines background intensity.

Note 9: These bits border color in 4Ø X 25 alpha mode and background color in 32Ø X 2ØØ color graphics mode.

Note 1Ø: This bit intensifies border or background which Bits 2,1,Ø define.

Note 11: Output any data to port to perform function.

Note 12: Refer to Appendix D.1 for a definition of the hardware interrupt vectors.

5.7 Communications Controller

Two INS825Ø Asynchronous Communications Element (ACE) chips function as a serial data input/output interface in our microcomputer system. The functional configuration of the INS825Ø is programmed by the system software.

The INS825Ø performs serial-to-parallel conversion on data characters received from a peripheral device or a MODEM and parallel-to-serial conversion on data characters received from the CPU. The CPU can read the complete status of the INS825Ø at any time during the functional operation. Status information reported includes the type and condition of the transfer operations being performed by the INS825Ø, as well as any error conditions (parity, overrun, framing, or break interrupt).

The serial controllers connect to the outside via DB-25S connectors at the back of the MPC. Pinouts for these connectors, marked "console" and "serial port" can be found in Table A-4.

In addition to providing control of asynchronous data communications, the INS825Ø includes a programmable Baud Generator that is capable of dividing the timing reference clock input by divisors of 1 to $(2^{16}-1)$ and producing a 16x clock for driving the internal transmitter logic. Provisions are also included to use this 16x clock to drive the receiver logic. Also included in the INS8250 is a complete MODEM-control capability and a processor-interrupt system that may be software tailored to the user's requirements to minimize the computing time required to handle the communications link.

5.7.1 Features

* Adds or Deletes Standard Asynchronous Communication Bit (Start, Stop, and Parity) to or from Serial Data Stream.

* Full Double Buffering Eliminates Need for Precise Synchronization.

* Independently Controlled Transmit, Receive, Line Status, and Data Set Interrupts.

* Programmable Baud Rate Generator Allows Division of Any Input Clock by 1 to (2^{16}-1) and Generates the Internal 16x Clock.

* Independent Receiver Clock Input.

* MODEM Control Functions (CTS, RTS, DSR, DTR, RI, and Carrier Detect).

* Fully Programmable Serial-Interface Characteristics
  - 5-, 6-, 7-, or 8-Bit Characters
  - Even, Odd, or No-Parity Bit Generation and Detection
  - 1-, 1 1/2-, or 2-Stop Bit Generation
  - Baud Rate Generation (DC to 56K Baud)

* False Start Bit Detection.

* Complete Status Reporting Capabilities.

* TRI-STATE TTL Drive Capabilities for Bidirectional Data Bus and Control Bus.

* Line Break Generation and Detection.

* Internal Diagnostic Capabilities
  -Loopback Controls for Communications Link Fault Isolation
  -Break, Parity, Overrun, Framing Error Simulation

* Full Prioritized Interrupt System Controls

5.7.2 INS8250 Accessible Registers

The system programmer may access or control any of the INS8250 registers summarized in Table 5-4 via the CPU. These registers are used to control INS8250 operations and to transmit and receive data.

Table 5-4. INS8250 Accessible Register Summary

Bit No.	0 DLAB=0	0 DLAB=0	1 DLAB=0	2	3	4	5	6	0 DLAB=1	1 DL
	Receiver Buffer Register (Read Only)	Transmitter Holding Register (Write Only)	Interrupt Enable Register	Interrupt Identification Register (Read Only)	Line Control Register	MODEM Control Register	Line Status Register	MODEM Status Register	Divisor Latch (LS)	Divi Lat (M
	RBR	THR	IER	IIR	LCR	MCR	LSR	MSR	DLL	DL
0	Data Bit 0*	Data Bit 0	Enable Received Data Available Interrupt (ERBFI)	"0" If Interrupt Pending	Word Length Select Bit 0 (WLS0)	Data Terminal Ready	Data Ready (DR)	Delta Clear to Send (DCTS)	Bit 0	Bit
1	Data Bit 1	Data Bit 1	Enable Transmitter Holding Register Empty Interrupt (ETBEI)	Interrupt ID Bit (0)	Word Length Select Bit 1 (WLS1)	Request to Send (RTS)	Overrun Error (OR)	Delta Data Set Ready (DDSR)	Bit 1	Bit
2	Data Bit 2	Data Bit 2	Enable Receiver Line Status Interrupt (ELSI)	Interrupt ID Bit (1)	Number of Stop Bits (STB)	Out 1	Parity Error (PE)	Trailing Edge Ring Indicator (TERI)	Bit 2	Bit
3	Data Bit 3	Data Bit 3	Enable MODEM Status Interrupt (EDSSI)	0	Parity Enable (PEN)	Out 2	Framing Error (FE)	Delta Receive Line Signal Detect (DRLSD)	Bit 3	Bit
4	Data Bit 4	Data Bit 4	0	0	Even Parity Select (EPS)	Loop	Break Interrupt (BI)	Clear to Send (CTS)	Bit 4	Bit
5	Data Bit 5	Data Bit 5	0	0	Stick Parity	0	Transmitter Holding Register Empty (THRE)	Data Set Ready (DSR)	Bit 5	Bit
6	Data Bit 6	Data Bit 6	0	0	Set Break	0	Transmitter Shift Register Empty (TSRE)	Ring Indicator (RI)	Bit 6	Bit
7	Data Bit 7	Data Bit 7	0	0	Divisor Latch Access Bit (DLAB)	0	0	Received Line Signal Detect (RLSD)	Bit 7	Bit

5.7.2.1 INS825Ø Line Control Register

The system programmer specifies the format of the asynchronous data communications exchange via the Line Control Register. In addition to controlling the format, the programmer may retrieve the contents of the Line Control Register for inspection. This feature simplifies system programming and eliminates the need for separate storage in system memory of the line characteristics. The contents of the Line Control Register are indicated in Table 5-4 and redescribed below.

Bits Ø and 1: These two bits specify the number of bits in each transmitted or received serial character. The encoding of bits Ø and 1 are as follows:

Bit 1	Bit Ø	Word Length
Ø	Ø	5 Bits
Ø	1	6 Bits
1	Ø	7 Bits
1	1	8 Bits

Bit 2: This bit specifies the number of Stop bits in each transmitted or received serial character. If bit 2 is a logic Ø, 1 Stop bit is generated or checked in the transmit or receive data, respectively. If bit 2 is a logic 1 when a 5-bit word length is selected via bits Ø and 1, 1 1/2 Stop bits are generated or checked. If bit 2 is a logic 1 when either a 6-, 7-or 8-bit word length is selected, 2 Stop bits are generated or checked.

Bit 3: This bit is the Parity Enable bit. When bit 3 is a logic 1, a Parity bit is generated (transmit data) or checked (receive data) between the last data word bit and Stop bit of the serial data. (The Parity bit is used to produce an even or odd number on 1s when the data word bits and the Parity bit are summed.)

Bit 4: This bit is the Even Parity Select bit. When bit 3 is a logic 1 and bit 4 is a logic Ø, an odd number of logic 1s is transmitted or checked in the data word bits and Parity bit. When bit 3 is a logic 1 and bit 4 is a logic 1, an even number of bits is transmitted or checked.

Bit 5: This bit is the Stick Parity bit. When bit 3 is a logic 1 and bit 5 is a logic 1, the Parity bit is transmitted and then detected by the receiver as a logic Ø if bit 4 is a logic 1 or as a logic 1 if bit 4 is a logic Ø.

Bit 6: This bit is the Set Break Control bit. When bit 6 is a logic 1, the serial output (SOUT) is forced to the Spacing (logic Ø) state and remains there regardless of other transmitter activity. The set break is disabled by setting bit 6 to a logic Ø. This feature enables the CPU to alert a terminal in a computer communications system.

Bit 7: This bit is the Divisor Latch Access Bit (DLAB). It must be set high (logic 1) to access the Divisor Latches of the Baud Rate Generator during a Read or Write operation. It must be set low (logic Ø) to access the Receiver Buffer, the Transmitter Holding Register, or the Interrupt Enable Register.

5.7.2.2 INS825Ø Programmable Baud Rate Generator

The INS825Ø contains a programmable Baud Rate Generator that is capable of taking any clock input (DC to 3.1 MHz) and dividing it by any divisor from 1 to $(2^{16}-1)$. The output frequency of the Baud Generator is 16x the Baud rate [divisor # = (frequency input) ÷ (baud rate x 16)]. Two 8-bit latches store the divisor in a 16-bit binary format. These Divisor

Latches must be loaded during initialization in order to insure desired operation of the Baud Rate Generator. Upon loading either of the Divisor Latches, a 16-bit Baud counter is immediately loaded. This prevents log counts on initial load.

Table 5-5 illustrates the use of the Baud Rate Generator with crystal frequencies of 1.8432 MHz. For baud rates of 38400 and below the error obtained is minimal. The accuracy of the desired baud rate is dependent on the crystal frequency chosen.

NOTE:

> The maximum operating frequency of the Baud Generator is 3.1 MHz. However, when using divisors of 3 and below, the maximum frequency is equal to the divisor in MHz. For example, if the divisor is 1, then the maximum frequency is 1 MHz. In no case should the data rate be greater than 56K Baud.

Table 5-5. Baud Rates Using 1.8432 MHz Crystal

Desired Baud Rate	Divisor Used to Generate 16x Clock	Percent Error Differences Between Desired & Actual
50	2304	-
75	1536	-
110	1047	0.026
134.5	857	0.058
150	768	-
300	384	-
600	192	-
1200	96	-
1800	64	-
2000	58	0.69
2400	48	-
3600	32	-
4800	24	-
7200	16	-
9600	12	-
19200	6	-
38400	3	-
56000	2	2.86

5.7.2.3 Line Status Register

This 8-bit register provides status information to the CPU concerning the data transfer. The contents of the Line Status Register are indicated in Table 5-4 and are described below.

Bit 0: This bit is the receiver Data Ready (DR) indicator. Bit 0 is set to a logic 1 whenever a complete incoming character has been received and transferred into the Receiver Buffer Register. Bit 0 may be reset to a logic 0 either by the CPU reading the data in the Receiver Buffer Register or by writing a logic 0 into it from the CPU.

Bit 1: This bit is the Overrun Error (OE) indicator. Bit 1 indicates that data in the Receiver Buffer Register was not read by the CPU before the next character was transferred into the Receiver Buffer Register, thereby destroying the previous character. The OE indicator is reset whenever the CPU reads the contents of the Line Status Register.

Bit 2: This bit is the Parity Error (PE) indicator. Bit 2 indicates that the received data character does not have the correct even or odd parity, as selected by the even-parity-select bit. The PE bit is set to a logic 1 upon detection of a parity error and is reset to a logic Ø whenever the CPU reads the contents of the Line Status Register.

Bit 3: This bit is the Framing Error (FE) indicator. Bit 3 indicates that the received character did not have a valid Stop bit. Bit 3 is set to a logic 1 whenever the Stop bit following the last data bit or parity bit is detected as a zero bit (Spacing level).

Bit 4: This bit is the Break Interrupt (BI) indicator. Bit 4 is set to a logic 1 whenever the received data input is held in the Spacing (logic Ø) state for longer than a full word transmission time (that is, the total time of Start bit + data bits + Parity + Stop Bits).

NOTE:

> Bits 1 through 4 are the error conditions that produce a Receiver Line Status interrupt whenever any of the corresponding conditions are detected.

Bit 5: This bit is the Transmitter Holding Register Empty (THRE) indicator. Bit 5 indicates that the INS825Ø is ready to accept a new character for transmission. In addition, this bit causes the INS825Ø to issue an interrupt to the CPU when the Transmit Holding Register Empty Interrupt enable is set high. The THRE bit is set to a logic 1 when a character is transferred from the Transmitter Holding Register into the Transmitter Shift Register. The bit is reset to logic Ø concurrently with the loading of the Transmitter Holding Register by the CPU.

Bit 6: This bit is the Transmitter Shift Register Empty (TSRE) indicator. Bit 6 is set to a logic 1 whenever the Transmitter Shift Register is idle. It is reset to logic Ø upon a data transfer from the Transmitter Holding Register. Bit 6 is a read-only bit.

Bit 7: This bit is permanently set to logic Ø.

Table 5-6. Interrupt Control Functions

Interrupt Identification Register				Interrupt Set and Reset Functions		
Bit 2	Bit 1	Bit Ø	Priority Level	Interrupt Type	Interrupt Source	Interrupt Reset Control
Ø	Ø	1	-	None	None	-
1	1	Ø	Highest	Receiver Line Status	Overrun Error or Parity Error or Framing Error or Break Interrupt	Reading the Line Status Register
1	Ø	Ø	Second	Received Data Available	Receiver Data Available	Reading the Receiver Buffer Register
Ø	1	Ø	Third	Trans-mitter Holding Register Empty	Trans-mitter Holding Register Empty	Reading the IIR Register (If source of interrupt) or Writing into the transmitter holding Register
Ø	Ø	Ø	Fourth	MODEM Status	Clear to Send or Data Set Ready or Ring Indi-cator or Received Line Sig-nal Detect	Reading the MODEM Status Register

5.7.2.4 Interrupt Identification Register

The INS825Ø has an on-chip interrupt capability that allows for complete flexibility in interfacing to all the popular microprocessors presently available. In order to provide minimum software overhead during data character transfers, the INS825Ø prioritizes interrupts into four levels. The four levels of interrupt conditions are as follows: Receiver Line Status (priority 1); Received Data Ready (priority 2); Transmitter Holding Register Empty (priority 3); and MODEM Status (priority 4).

Information indicating that a prioritized interrupt is pending and the type of that interrupt are stored in the Interrupt Identification Register (refer to Table 5-6). The Interrupt Identification Register (IIR) when addressed during chip-select time, freezes the highest priority interrupt pending and no other interrupts are acknowledged until the particular interrupt is serviced by the CPU. The contents of the IIR are indicated in Table 5-4 and are described below.

Bit Ø: This bit can be used in either a hardwired prioritized or polled environment to indicate whether an interrupt is pending. When bit Ø is a logic Ø, an interrupt is pending and the IIR contents may be used as a pointer to the appropriate interrupt service routine. When bit Ø is a logic 1, no interrupt is pending and polling (if used) continues.

Bits 1 and 2: These two bits of the IIR are used to identify the highest priority interrupt pending as indicated in Table 5-6.

Bits 3 through 7: These five bits of the IIR are always logic Ø.

5.7.2.5 Interrupt Enable Register

This 8-bit register enables the four types of interrupts of the INS8250 to separately activate the chip Interrupt (INTRPT) output signal. It is possible to totally disable the interrupt system by resetting bits 0 through 3 of the Interrupt Enable Register. Similarly, by setting the appropriate bits of this register to a logic 1, selected interrupts can be enabled. Disabling the interrupt system inhibits the Interrupt Identification Register and the active (high) INTRPT output from the chip. All other system functions operate in their normal manner, including the setting of the Line Status and MODEM Status Registers. The contents of the Interrupt Enable Register are indicated in Table 5-4 and are described below.

Bit 0: This bit enables the Received Data Available Interrupt when set to logic 1.

Bit 1: This bit enables the Transmitter Holding Register Empty Interrupt when set to logic 1.

Bit 2: This bit enables the Receiver Line Status Interrupt when set to logic 1.

Bit 3: This bit enables the MODEM Status Interrupt when set to logic 1.

Bits 4 through 7: These four bits are always logic 0.

5.7.2.6 MODEM Control Register

This 8-bit register controls the interface with the MODEM or data set (or a peripheral device emulating a MODEM). The contents of the MODEM Control Register are indicated in Table 5-4 and are described below.

Bit Ø: This bit controls the Data Terminal Ready (DTR) output. When bit Ø is set to logic 1, the DTR output is forced to a logic Ø. When bit Ø is reset to a logic Ø, the DTR output is forced to a logic 1.

NOTE:

The DTR output of the INS825Ø may be applied to an EIA inverting line driver (such as the DS1488) to obtain the proper polarity input at the succeeding MODEM or data set.

Bit 1: This bit controls the Request to Send (RTS) output. Bit 1 affects the RTS output in a manner identical to that described above for bit Ø.

Bit 2: This bit controls the Output 1 (OUT 1) signal which is an auxiliary user-designated output. Bit 2 affects the OUT 1 output in a manner identical to that described above for bit Ø.

Bit 3: This bit controls the Output 2 (OUT 2) signal which is an auxiliary user-designated output. Bit 3 affects the OUT 2 output in a manner identical to that described above for bit Ø.

Bit 4: This bit provides a loopback feature for diagnostic testing of the INS825Ø. When bit 4 is set to logic 1, the following occur: the transmitter Serial Output (SOUT) is set to the Marking (logic 1) state; the receiver Serial Input (SIN) is disconnected; the output of the Transmitter Shift Register is "looped back" into the Receiver Shift

Register input; the four MODEM Control inputs (CTS, DSR, RLSD, and RI) are disconnected; and the four MODEM Control outputs (DTR, RTS, OUT 1, and OUT 2) are internally connected to the four MODEM Control inputs. In the diagnostic mode, data that is transmitted is immediately received. This feature allows the processor to verify the transmit- and receive-data paths of the INS8250.

In the diagnostic mode, the receiver and transmitter interrupts are fully operational. The MODEM Control Interrupts are also operational but the interrupts' sources are now the lower four bits of the MODEM Control Register instead of the four MODEM Control inputs. The Interrupts are still controlled by the Interrupt Enable Register.

The INS8250 interrupt system can be tested by writing into the lower six bits of the Line Status Register and the lower four bits of the MODEM Status Register. Setting any of these bits to a logic 1 generates the appropriate interrupt (if enabled). The resetting of these interrupts is the same as in normal INS8250 operation. To return to normal operation, the registers must be reprogrammed for normal operation and then bit 4 of the MODEM Control Register must be reset to logic 0.

Bits 5 through 7: These bits are permanently set to logic 0.

5.7.2.7 MODEM Status Register

This 8-bit register provides the current state of the control lines from the MODEM (or peripheral device) to the CPU. In addition to this current-state information, four bits of the MODEM Status Register provide change information. These bits are set to a logic 1 whenever a control input from the MODEM changes state. They are reset to logic 0 whenever the CPU reads the MODEM Status Register.

The contents of the MODEM Status Register are indicated in Table 2 and are described below.

Bit Ø: This bit is the Delta Clear to Send (DCTS) indicator. Bit Ø indicates that the CTS input to the chip has changed state since the last time it was read by the CPU.

Bit 1: This bit is the Delta Data Set Ready (DDSR) indicator. Bit 1 indicates that the DSR input to the chip has not changed state since the last time it was read by the CPU.

Bit 2: This bit is the Trailing Edge of Ring Indicator (TERI) detector. Bit 2 indicates that the RI input to the chip has changed from an On (logic 1) to an Off (logic Ø) condition.

Bit 3: This bit is the Delta Received Line Signal Detector (DRLSD) indicator. Bit 3 indicates that the RLSD input to the chip has changed state.

NOTE:

Whenever bit Ø, 1, 2, or 3 is set to logic 1, a MODEM Status interrupt is generated,

Bit 4: This bit is the complement of the Clear to Send (CTS) input. If bit 4 (loop) of the MCR is set to a 1, this bit is equivalent to RTS in the MCR.

Bit 5: This bit is the complement of the Data Set Ready (DSR) input. If bit 4 of the MCR is set to a 1, this bit is equivalent to DTR in the MCR.

Bit 6: This bit is the complement of the Ring Indicator (RI) input. If bit 4 of the MCR is set to a 1, this bit is equivalent to OUT 1 in the MCR.

Bit 7: This bit is the complement of the Received Line Signal Detect (RLSD) input. If bit 4 of the MCR is set to a 1, this bit is equivlent to OUT 2 of the MCR.

5.8 Arithmetic Coprocessor (Option)

The optional Arithmetic processor (8Ø87) is a single chip device providing register, data types, control, and instruction capabilities at the hardware level. The 8Ø87 also executes numerous built-in transcendental functions and executes instructions as a coprocessor to a maximum mode 8Ø88.

As a coprocessor to an 8Ø88, the 8Ø87 is wired in parallel with the CPU. The CPU's status and queue status lines enable the 8Ø87 to monitor and decode instructions in synchronization with the CPU and without any CPU overhead. The 8Ø87 can process in parallel with, and independent of, the host CPU.

The 8Ø87 is internally divided into two processing elements, the control unit (CU) and the numeric execution unit (NEU). The NEU executes all numeric instructions, while the CU receives and decodes instructions, reads and writes memory operands and executes numeric processor extension (NPX) control instructions. The two elements are able to operate independently of one another, allowing the CU to maintain synchronization with the CPU while the NEU is busy processing a numeric instruction.

5.9 Programmable Peripheral Interface

Two 8255A Programmable Peripheral Interface (PPI) chips are included in the MPC to implement parallel ports. The Intel 8255A is a general purpose programmable I/O device, having 24 I/O pins which may be individually programmed in 2 groups of 12 and used in 3 major modes of operation as follows:

Mode Ø

Each group of 12 I/O pins may be programmed in sets of 4 to be input or output.

Mode 1

Each group of 12 may be programmed for 8 lines of input or output. Of the remaining 4 pins, 3 are used for handshaking and interrupt control signals.

Mode 2

One group of 12 and an additional line from the other groups is programmed as an 8-bit bidirectional bus with 5 handshake lines. The 11 lines remaining of the other group can be programmed as Mode Ø or Mode 1.

The 8255A is a very powerful tool for interfacing peripheral equipment to the microcomputer system. It represents the optimum use of available pins and is flexible enough to interface almost any I/O device without the need for additional external logic.

Each peripheral device in a microcomputer system usually has a "service routine" associated with it. The routine manages the software interface between the device and the CPU. The functional description of the 8255A is programmed by the I/O service routine and becomes an extension of the system software.

Due to the high flexibility of this device, an adequate description of its programming and operation is outside the scope of this manual. Therefore, if any interfacing is to be done with the parallel port(s) for which drivers are not provided by CDP, a full understanding of this device should be acquired by studying the 8255A specification.

5.1Ø Interrupt Controller

The programmable interrupt controller (8259), packaged in a 28-pin DIP, can handle up to eight vectored priority interrupts for the CPU. It is designed to minimize the software and real time overhead in handling multi-level priority interrupts.

The Programmable Interrupt Controller (PIC) functions as an overall manager in an Interrupt-Driver system environment. It accepts requests from the peripheral equipment, determines which of the incoming requests is of the highest importance, ascertains whether the incoming request has a higher priority value than the level currently being serviced, and issues an interrupt of the CPU based on this determination.

Each peripheral device or structure usually has a special program or "routine" that is associated with its specific functional or operational requirements; this is referred to as a "service routine." The PIC, after issuing an interrupt to the CPU, must somehow input information into the CPU that can "point" the Program Counter to the service routine associated with the requesting device. This "pointer" is an address in a vectoring table.

The 8259A is a device specifically designed for use in real time interrupt driven microcomputer systems. It manages eight levels or requests and has built-in features for expandability (up to 64 levels). It is programmed by the system's software as an I/O peripheral. A selection of priority modes is available to the programmer so that the manner in which the requests are processed by the 8259A can be configured to match system requirements. The priority modes can be changed or reconfigured dynamically at any time during the main program. This means that the complete interrupt structure can be defined as required, based on the total system environment.

5.11 Floppy Disk Drive Functional Description

The disk drive is fully self-contained, requiring no operator intervention during normal operation. The disk drive consists of a Spindle Drive system, a Head Positioning system, and a Read/Write/Erase system.

When the front latch is opened, access is provided for inserting a diskette. The diskette is positioned by guides and by the front latch. The in/out location is ensured when the diskette is seated against the backstop.

Closing the latch activates the cone/clamp system, which centers the diskette and clamps it to the drive hub. The drive hub is driven at a constant speed of 3ØØ rpm by a servo-controlled DC motor. The magnetic head is loaded into contact with the recording medium whenever the front latch is closed.

The magnetic head is positioned over the desired track by means of a stepper motor/band assembly and its associated electronics. This positioner employs a one-step rotation to cause a one-track linear movement. When a write-protected diskette is inserted into the disk drive, the Write Protect sensor disables the write electronics of the disk drive, and a Write Protect output signal is applied to the interface.

The disk drive is also supplied with the following sensor systems:

1. A Track ØØ switch that senses when the Head/Carriage assembly is positioned at Track ØØ.

2. The Index sensor, which consists of a LED light source and phototransistor, is positioned such that a digital signal is generated when an index hole is detected.

3. The Write Protect sensor disables the disk drive write electronics whenever a write-protect tab is applied to the diskette.

The disk drive uses a standard 5.25-inch (133.4mm) diskette. For programming, double-sided, double density soft sectored diskettes are used. Using single sensity can result in corrupted data storage.

5.12 Floppy Disk Controller (FDC)

The uPD 765 Floppy Disk Formatter/Controller is the link between the processor and the floppy disk drives in the MPC. This controller performs all the functions necessary to read or write data to the floppy disks. Among other features, it has single/multiple disk sector reading with automatic search or entire track read capability, single/multiple disk sector writing with automatic sector search capability, automatic disk track seeking with verification, and program-selectable track-to-track stepping time.

A data separator external to the controller chip is required to assist in organizing data incoming from the disk drives. The data separator is a phase lock loop design to provide superior data recovery and reliability. Data is sent between the floppy disk controller and system memory under DMA control using a programmed interrupt driven transfer.

The Floppy Disk Formatter/Controller is a MOS LSI device, which performs the functions of a Floppy Disk Formatter/Controller in a single chip implementation. Address mark detection is internal to the FDC which supplies the phase locked loop on read electronics. The track stepping rate, head load time, and head unload time may be programmed by the user. The FDC offers many additional features such as multiple sector transfers in both read and write with a single command and full IBM compatibility in both single and double density modes.

It includes features such as:

* IBM Compatible in both single and double density recording formats.

* Multi-sector and multi-track transfer capability.

* Accommodates single and double density format of 128, 256, 512, or 1Ø24 bytes per sector.

* Drive up to 4 floppy disks.

* Data scan capability - will scan a single sector or an entire cylinder's worth of data fields, comparing on a byte by byte basis, data in the processor's memory with data read from the diskette.

* Data transfers in DMA or NON-DMA mode.

* Parallel seek operations on up to 4 drives.

* Compatible with most microprocessors including 8Ø8ØA, 8Ø85A, uPD 78Ø (Z8Ø).

* Single phase 8MHz clock.

* Write Precompensation (MFM or FM).

5.12.1 Description

The FDC is an LSI Chip, which contains the circuitry and control functions for interfacing a processor to 4 floppy disk drives. It is capable of supporting either IBM 374Ø single density format (FM) or IBM System 34 Double Density format (MFM) including double sided recording. The FDC provides control signals which simplify the design of an external phase locked loop and write precompensation circuitry. The FDC simplifies and handles most of the burdens associated with implementing a Floppy Disk Interface.

Hand-shaking signals are provided in the FDC which make DMA operation easy to incorporate with the aid of an external DMA Controller chip. The FDC will operate in either DMA or non-DMA mode. In the non-DMA mode, the FDC generates interrupts to the processor every time a data byte is available. In the DMA mode, the processor need only load the command into the FDC and all data transfers occur under control of the FDC and DMA controller.

There are 15 separate commands which the uPD765 will execute. Each of these commands require multiple 8-bit bytes to fully specify the operation which the processor wishes the FDC to perform. The following commands are available:

Read Data	Scan High or Equal	Write Deleted Data
Read ID	Scan Low or Equal	Seek
Read Deleted Scan	Specify	Recalibrate (Re-store to Track Ø)
Read a Track	Write Data	Sense Interrupt Status
Scan Equal	Formats a Track	Sense Drive Status

5.12.2 Internal Registers

The uPD765 contains two registers which may be accessed by the main system processor; a Status Register and a Data Register. The 8-bit Main Status Register contains the status information of the FDC and may be accessed at any time. The 8-bit Data Register (actually consists of several registers in a stack with only one register presented to the data bus at a time) which stores data, commands, parameters, and FDD status information. Data bytes are read out of, or written into, the Data Register in order to program or obtain the results after a particular command. The Status Register may only be read and is used to facilitate the transfer of data between the processor and uPD765.

The relationship between the Status/Data registers and the signals RD, WR, and $A_Ø$ is shown below.

Table 5-7. Status/Data Registers

AØ	RD	WR	FUNCTION
Ø	Ø	1	Read Main Status Register
Ø	1	Ø	Illegal
Ø	Ø	Ø	Illegal
1	Ø	Ø	Illegal
1	Ø	1	Read from Data Register
1	1	Ø	Write into Data Register

Table 5-8. Main Status Register Bits

The bits in the Main Status Register are defined as follows:

BIT NUMBER	NAME	SYMBOL	DESCRIPTION
$DB_Ø$	FDD Ø Busy	$D_ØB$	FDD number Ø is in the Seek mode. If any of the bits is set, FDC will not accept read or write command.
DB_1	FDD 1 Busy	D_1B	FDD number 1 is in the Seek mode. If any of the bits is set, FDC will not accept read or write command.
DB_2	FDD 2 Busy	D_2B	FDD number 2 is in the Seek mode. If any of the bits is set, FDC will not accept read or write command.
DB_3	FDD 3 Busy	D_3B	FDD number 3 is in the Seek mode. If any of the bits is set, FDC will not accept read or write command.
DB_4	FDC Busy	CB	A read or write command is in process. FDC will not accept any other command.

Table 5-8. (Continued)

DB_5	Execution Mode	EXM	This bit is set only during execution phase in non-DMA mode. When DB_5 goes low, execution phase has ended, and result phase was started. It operates only during NON-DMA mode of operation.
DB_6	Data Input/ Output	DIO	Indicates direction of data transfer between FDC and Data Register. If DIO = "1", then transfer is from Data Register to the Processor. If DIO = "Ø", then transfer is from the Processor to Data Register.
DB_7	Request for Master	RQM	Indicates Data Register is ready to send or receive data to or from the Processor. Both bits DIO and RQM should be used to perform the hand-shaking functions of "ready" and "direction" to the processor.

The DIO and RQM bits in the Status Register indicate when Data is ready and in which direction data will be transferred on the Data Bus. The max time between the last RD or WR during command or result phase and DIO and RQM getting set or reset is 12 us. For this reason, every time Main Status Register is read the CPU should wait 12 us. The max time from the trailing edge of the last RD in the result phase to when DB_4 (FDC Busy) goes low is 12 us.

5.12.3 Command Sequence

The uPD765 is capable of performing 15 different commands. Each command is initiated by a multi-byte transfer from the processor, and the result after execution of the command may also be a multi-byte transfer back to the processor. Because of this multi-byte interchange of information between uPD765 and the processor, it is convenient to consider each command as consisting of three phases:

Command Phase:	The FDC receives all information required to perform a particular operation from the processor.
Execution Phase:	The FDC performs the operation it was instructed to do.
Result Phase:	After completion of the operation, status and other housekeeping information are made available to the processor.

In the following table, Ø = logical Ø
1 = logical 1
X = don't care

Table 5-9. Command Symbol Description

SYMBOL	NAME	DESCRIPTION
$A_Ø$	Address Line Ø	$A_Ø$ controls selection of Main Status Register ($A_Ø$ = Ø) or Data Register ($A_Ø$ = 1)
C	Cylinder Number	C stands for the current/selected Cylinder (track) number Ø through 76 of the medium.
D	Data	D stands for the data pattern which is going to be written into a Sector.
D_7-$D_Ø$	Data Bus	8-bit Data Bus, where D_7 stands for a most significant bit, and $D_Ø$ stands for a least significant bit.
DTL	Data Length	When N is defined as ØØ, DTL stands for the data length which users are going to read out or write into the Sector.
EOT	End of Track	EOT stands for the final Sector number on a Cylinder. During Read or Write operation FDC will stop date transfer after a sector # equal to EOT.

Table 5-9. (Continued)

GPL	Gap Length	GPL stands for the length of Gap 3. During Read/Write commands this value determines the number of bytes the VCOs will stay low after two CRC bytes. During Format command it determines the size of Gap3.
H	Head Address	H stands for head number Ø or 1, as specified in ID field.
HD	Head	HD stands for a selected head number Ø or 1 and controls the polarity of pin 27. (H = HD in all command words.)
HLT	Head Load Time	HLT stands for the head load time in the FDD (2 to 254 ms in 2 ms increments).
HUT	Head Unload Time	HUT stands for the head unload time after a read or write operation has occurred (16 to 24Ø ms in 16 ms increments).
MF	FM or MFM Mode	If MF is low, FM mode is selected, and it it is high, MFM mode is selected.

Table 5-9. (Continued)

MT	Multi-Track	If MT is high, a multi-track operation is to be performed. If MT = 1 after finishing Read/Write operation on side Ø FDC will automatically start searching for sector 1 on side 1.
N	Number	N stands for the number of data bytes written in a Sector.
NCN	New Cylinder Number	NCN stands for a new Cylinder number which is going to be reached as a result of the Seek operation. Desired position of Head.
ND	Non-DMA Mode	ND stands for operation in the non-DMA mode.
PCN	Present Cylinder Number	PCN stands for the Cylinder number after the completion of SENSE INTERRUPT STATUS Command. Position of Head at present time.
R	Record	R stands for the Sector number which will be read or written.
R/W	Read/Write	R/W stands for either Read (R) or Write (W) signal.

Table 5-9. (Continued)

SC	Sector	SC indicates the number of Sectors per Cylinder.
SK	Skip	SK stands for Skip Deleted Data Address Mark.
SRT	Step Rate Time	SRT stands for the Stepping Rate for the FDD. (1 to 16 ms in 1 ms increments.) Stepping Rate applies to all drives. (F = 1 ms, E = 2 ms, etc.)
ST Ø ST 1 ST 2 ST 3	Status Ø Status 1 Status 2 Status 3	ST Ø-3 stand for one of four registers which store the status information after a command has been executed. This information is available during the result phase after command execution. These registers should not be confused with the main status register (selected by $A_{Ø}$ = Ø). STØ-3 may be read only after a command has been executed and contain information relevant to that particular command.
STP		During a Scan operation, if STP = 1, the data in contiguous sectors is compared byte by byte with data sent from the

Table 5-9. (Continued)

		processor (or DMA); and if STP = 2, then alternate sectors are read and compared.
USØ,US1	Unit Select	US stands for a selected drive number Ø or 1.

5.12.4 Format A Track

The Format Command allows an entire track to be formatted. After the INDEX HOLE is detected, Data is written on the diskette: Gaps, Address Marks, ID Fields and Data Fields, all per the IBM System 34 (Double Density) or System 374Ø (Single Density) Format are recorded. The particular format which will be written is controlled by the values programmed into N (number of bytes/sector), SC (sector/cylinder), GPL (Gap Length), and D (Data Pattern) which are supplied by the processor during the Command Phase. The Data Field is filled with the Byte of data stored in D. The ID Field for each sector is supplied by the processor: that is, four data requests per sector are made by the FDC for C (Cylinder Number), H (Head Number), R (Sector Number), and N (Number of Bytes/Sector). This allows the diskette to be formatted with nonsequential sector numbers, if desired.

The processor must send new values for C, H, R, and N to the uPD765 for each sector on the track. If FDC is set for DMA mode, it will issue 4 DMA requests per sector. If it is set for interrupt mode, it will issue four interrupts per sector and the processor must supply C, H, R, and N load for each sector. The contents of the R register is incremented by one after each sector is formatted, thus, the R register contains a value of R when it is read during the

Result Phase. This incrementing and formatting continues for the whole track until the FDC encounters the INDEX HOLE for the second time, whereupon, it terminates the command.

If a FAULT signal is received from the FDD at the end of a write operation, then the FDC sets the EC flag of Status Register Ø to a 1 (high) and terminates the command after setting bits 7 and 6 of Status Register Ø to Ø and 1, respectively. Also the loss of a READY signal at the beginning of a command execution phase cause bits 7 and 6 of Status Register Ø to be set to Ø and 1, respectively.

Table 5-1Ø and 5-11 show the relationship between N, SC, and GPL for various sector sizes.

Table 5-1Ø. 5 1/4-Inch Mini Floppy

FORMAT	SECTOR SIZE	N	SC	GPL1	GPL2
	128 bytes/Sector	ØØ	12	Ø7	Ø9
FM Mode	128	ØØ	1Ø	1Ø	19
	256	Ø1	Ø8	18	3Ø
	512	Ø2	Ø4	46	87
FM Mode	1Ø24	Ø3	Ø2	C8	FF
	2Ø48	Ø4	Ø1	C8	FF
	256	Ø1	12	ØA	ØC
	256	Ø1	1Ø	2Ø	32
MFM	512	Ø2	Ø8	2A	5Ø
Mode	1Ø24	Ø3	Ø4	8Ø	FØ
	2Ø48	Ø4	Ø2	C8	FF
	4Ø96	Ø5	Ø1	C8	FF

Table 5-11. 8-Inch Standard Floppy

FORMAT	SECTOR SIZE	N	SC	GPL1	GPL2	REMARKS
	128 bytes/sector	ØØ	1A(16)	Ø7(16)	1B(16)	IBM Diskette 1
FM Mode	256	Ø1	ØF(16)	ØE(16)	2A(16)	IBM Diskette 2
	512	Ø2	Ø8	1B(16)	3A(16)	
	1Ø24 bytes/sector	Ø3	Ø4	47	8A	
FM Mode	2Ø48	Ø4	Ø2	C8	FF	
	4Ø96	Ø5	Ø1	C8	FF	
	256	Ø1	1A(16)	ØE(16)	36(16)	IBM
	512	Ø2	ØF(16)	1B(16)	54(16)	Diskette 2D
	1Ø24	Ø3	Ø8	35(16)	74(16)	IBM
MFM Mode	2Ø48	Ø4	Ø4	99	FF	Diskette 2D
	4Ø96	Ø5	Ø2	C8	FF	
	8192	Ø6	Ø1	C8	FF	

Note: 1) Suggested values of GPL in Read or Write Commands to avoid splice point between data field and ID field of contiguous sections.

2) Suggested values of GPL in format comand.

3) In MFM mode, FDC can perform a read operation only with 128 bytes/sector (N = ØØ).

Table 5-12. Status Register Ø, Identification

BIT NO.	BIT NAME	BIT SYMBOL	DESCRIPTION
D_7	Interrupt Code	IC	D_7=Ø and D_6=Ø Normal termination of Command, (NT). Command was completed and properly executed.
D_6			D_7=Ø and D_6=1 Abnormal Termination of Command, (AT). Execution of Command was started but was not successfully completed.
			D_7=1 and D_6=Ø Invalid Command issue (IC). Command which was issued was never started.
			D_7=1 and D_6=1 Abnormal Termination because during command execution the ready signal from FDD changed state.
5	Seek End	SE	When the FDC completes the SEEK Command, this flag is set to 1 (high).
D_4	Equipment Check	EC	If a fault Signal is received from the FDD or if the Track Ø Signal fails to occur after 77 step pulses (Recalibrate Command) then this flag is set.

Table 5-12. (Continued)

D_3	Not Ready	NR	When the FDD is in the not-ready state and a read or write command is issued, this flag is set. If a read or write command is issued to Side 1 of a single sided drive, then this flag is set.
D_2	Head Address	HD	This flag is used to indicate the state of the head at Interrupt
D_1	Unit Select 1	US1	These flags are used to indicate a Drive Unit
$D_Ø$	Unit Select Ø	USØ	Number at Interrupt.

Table 5-13. Status Register 1 Identification

BIT NO.	NAME	SYMBOL	DESCRIPTION
D_7	End of Cylinder	EN	When the FDC tries to access a Sector beyond the final sector of a Cylinder, this flag is set.
D_6			Not used. This bit is always Ø (low).
D_5	Data Error	DE	When the FDC detects a CRC error in either the ID

Table 5-13. (Continued)

			field or the data field, this flag is set.
D_4	Over Run	OR	If the FDC is not serviced by the main-systems during data transfers, within a certain time interval, this flag is set.
D_3			Not used. This bit always Ø (low).
D_2	No Data	ND	During execution of READ DATA, WRITE DELETED DATA or SCAN Command, if the FDC cannot find the Sector specified in the IDR Register, this flag is set. During executing the READ ID Command, if the FDC cannot read the ID field without an error, then this flag is set. During the execution of the READ A Cylinder Command, if the starting sector cannot be found, then this flag is set.
D_1	Not Writable	NW	During execution of WRITE DATA, WRITE DELETED DATA or Format A Cylinder Command, if the FDC detects a write protect signal from the FDD, then this flag is set.

Table 5-13. (Continued)

D_0	Missing Address Mark	MA	If the FDC cannot detect the Data Address Mark after encountering the index hole twice, then this flag is set. If the FDC cannot detect the Data Address Mark or Deleted Data Address Mark, this flag is set. Also at the same time, the MD (Missing Address Mark in Data Field) of Status Register 2 is set.

Table 5-14. Status Register 2 Identification

BIT			DESCRIPTION
NO.	NAME	SYMBOL	
D_7			Not used. This bit is always Ø (low).
D_6	Control Mark	CM	During executing the READ DATA or SCAN Command, if the FDC encounters a Sector which contains a Deleted Data Address Mark, this flag is set.
D_5	Data Error in Data Field	DD	If the FDC detects a CRC error in the data field, then this flag is set.
D_4	Wrong Cylinder	WC	This bit is related with the ND bit, and when the

Table 5-14. (Continued)

			contents of C on the medium is different from that stored in the IDR, this flag is set.
D_3	Scan Equal Hit	SH	During execution, the SCAN Command, if the condition of "equal" is satisfied, this flag is set.
D_2	Scan Not Satisfied	SN	During executing the SCAN Command, if the FDC cannot find a Sector on the cylinder which meets the condition, then this flag is set.
D_1	Bad Cylinder	BC	This bit is related with the Cylinder ND bit, and when the content of C on the medium is different from that stored in the IDR and the content of C is FF, then this flag is set.
D_0	Missing Address Mark in Data Field	MD	When data is read from the medium, if the FDC cannot find a Data Address Mark or Deleted Data Address Mark, then this flag is set.

Table 5-15. Status Register 3 Identification

BIT NO.	NAME	SYMBOL	DESCRIPTION
D_7	Fault	FT	This bit is used to indicated the status of the Fault signal from the FDD.
D_6	Write Protected	WP	This bit issued to indicated the status of the Write Protect signal from the FDD.
D_5	Ready	RY	This bit is used to indicate the status of the Ready signal from the FDD.
D_4	Track Ø	TØ	This bit is used to indicated the status of the Track Ø signal from the FDD.
D_3	Two Side	TS	This bit is used to indicate the status of the Two Side signal from the FDD.
D_2	Head Address	HD	This bit is used to indicate the status of Side Select signal to the FDD.
D_1	Unit Select 1	US 1	This bit is used to indicate the status of the Unit Select 1 signal to the FDD.

Table 5-15. (Continued)

$D_Ø$	Unit Select Ø	US Ø	This bit is used to indicate the status of the Unit Select Ø signal to the FDD.

5.13 Winchester Disk Interface

The Winchester disk interface is an 8 bit parallel data port with 4 control lines for byte and unit synchronization. This interface connects the MPC system board to the CDP cache buffered Winchester controller.

5.14 Winchester Disk Drive

The MPC is optionally configured with one 5.25 Winchester disk drive, a Tandon 6Ø3S with 1Ø Mbyte formatted capacity. This drive is not field serviceable.

5.15 Winchester Controller

The CDP cache buffer Winchester controller is included in those systems which are purchased with hard disk options. This controller optimizes disk access times by managing 88 sector buffers. Reading a sector will cause that sector and the next 7 to be placed in memory, overwriting the least recently used buffers. Subsequent accesses to sequential sectors by the MPC are speeded up by a factor of approximately 4Ø. For example, average access time for 1 sector read is approximately 2ØØ ms/512 bytes (disk drive limitation), and average access for 8 sectors is 235 ms/4Ø96 bytes.

5.16 Parallel Printer Interface

The parallel printer interface is Centronics compatible. It has 8 parallel data lines and 4 control lines for byte synchronization and printer status. This port is located on the back of the unit in a DB-37 connector for reliability. PINOUTS for the connector are shown in Table A-5.

5.17 Timer Counter

The timer counter is a 3 channel device which performs several tasks in the MPC. Channel Ø is used to time the refreshing of the dynamic RAM. Channel 1 provides an interrupt for time functions in the MPC (real time clock, timeouts, etc.). Channel 2 is a tone generator. The count value determines the frequency of the signal sent to the speaker.

SECTION 6: TROUBLESHOOTING AND MAINTENANCE

6.1 Troubleshooting

In the course of operating the system, certain difficulties may be encountered that can be easily remedied without outside assistance. Listed below are several of the more common problems that fall into this category, their likely causes, and solutions.

Symptom	Possible Cause	Solution
The unit does not operate upon power ON	Fuse blown	Replace fuse
Improper image appears on screen	Mismatched baud rates	Set consistent baud rates
	Wrong terminal configuration	Set terminal for 7 data bits 1 start and 1 stop bit, no parity full duplex
	Incompatible terminal/computer configuration	Make new connector cable
Disk drives operate but screen response is "Insert Bootable Disk"	No system tracks on diskette	Insert system diskette

Screen displays message OPERATION FAILED	Unreadable diskette	Try another system disk (For hard disk systems try another BLDSYS 86 or MS-DOS system disk)
Screen displays message INCOMPATIBLE SYSTEM	Improper diskette to boot system	Use a diskette which is compatible with the system
Terminal does not respond to commands	Damaged diskette	Insert new diskette
	Keyboard not functioning	Try new keyboard cable
Characters echo on screen	Improper duplex setting	Set terminal for full-duplex

6.2 Unit Maintenance

The MPC system requires virtually no maintenance, thanks to its modular, solid-state design and high-reliability components. However, there are a few minor procedures that should be followed to help assure trouble-free operation. These are:

- Run a floppy head-cleaning diskette in each drive about once every 30 days, or when there are repeated read errors, to keep the mechanism in top operating order. These diskettes can be purchased from any distributor of floppy disk media.

- Do not leave disk drive doors open during shutdown. Dust will collect inside the drive mechanisms, thus reducing performance. Caution: Remove diskettes prior to shutdown or data on the diskette may be lost.

- Do not leave the MPC running needlessly after use.

- Do not use the top of the MPC as a shelf for storing miscellaneous papers, tools, or diskettes (see next subsection). The unit should stand unencumbered for maximum efficiency and safety.

- Do not transport the optional expansion hard disk unit without locking the hard disk drive with the mechanism provided (8-inch only).

6.3 Diskette Care

As indicated, proper maintenance of the diskettes is vital if trouble-free operation is to be maintained. It is recommended that all instructions on the diskette envelopes be carefully read to ensure proper care. However, there are a few general rules that apply to all diskettes and are stated here.

When handling the diskettes, never touch the recording surface. Always handle them on the jacket area. After using a diskette, place it back in its protective envelope immediately. Exposed diskettes can be easily damaged if they come into contact with smoke, dust, debris, or other environmental hazards. Also avoid bending, folding, or creasing the diskettes.

When writing on the diskette label, do not use a ball-point pen, pencil, or other hard marker, as an impression can be made on the diskette causing damage. Use only a soft felt tip pen. Never try to erase information put on a label. The resulting debris can lodge between the diskette and its jacket, causing damage.

Always keep the diskettes away from magnets or magnetized objects and from direct sunlight. Diskettes work best at temperatures between 4Ø degrees F and 12Ø degrees F, and temperatures can easily exceed the higher limit near a window in warmer climates.

APPENDIX A

SPECIFICATIONS
AND
PIN CONFIGURATIONS

Table A-1. Specification Summary

Overall System

Dimensions	
Height	5-Inches (12.5 cm)
Width	22.5-Inches (56.25 cm)
Depth	15-Inches (37.5 cm)
Weight	25 Pounds (11.36 kilos)
Power Requirements	110VAC 60Hz 220VAC 50Hz
Power Supply	+5VDC 10Amp +12VDC 3Amp -12VDC 3Amp
Environmental	
Operating Temperature	50 degrees F to 100 degrees F (10-39 degrees C)
Relative Humidity	30% to 80% non-condensing

Processor

Type	8088
Clock Frequency	4.77 MHz
Word Size	16 Bits
Address Size	20 Bits

Main Memory

Type	128 kilobyte RAM
Size	4 kilobyte EPROM

Table A-1. (Continued)

Disk Assembly

Media	5 1/4-inch floppy disk (Double-sided double density)
Storage Capacity	1Mbyte (dual floppy disk system)
Tracks	4Ø
Sectors/Track	8
Bytes/Track	4Ø96
Bytes/Sector	512
Data Transfer Rate	25Ø Kilobits/sec
Disk Rotation Speed	3ØØrpm
Access Time	
Track-to-Track	5 ms
Average	75 ms
Head Life	2Ø,ØØØ hours
Media Life	3 million passes per track

Disk Assembly 5 1/4-Inch Winchester (Optional)

Media	5 1/4-inch Winchester Hard Disk
Storage Capacity	1ØM
Disks	3 Model Dependent
Read/Write Heads	6 Model Dependent
Cylinders	135
Track Density	254 TPI
Data Transfer Rate	5.Ø Megabits/sec
Access Time	
Track-to-Track	3 msec
Head Setting	15 msec
Disk Rotation Speed	36ØØ rpm

Table A-2. Keyboard

STANDARD FEATURES:

*19mm Solid State Switches
*Capacitive Switch Technology
*Durable 2-Shot Molded Keytops
*Full N-Key Rollover
*Rigid Frame Mounting
*5 Year Switch Warranty
*Matte Finish, Sculptured Keytops
*Synchronous Serial Output
*+5 Volt Only Operation (+12 Volt Optional for Long Cables - Over 6 feet)
*Auto Repeat
*Chassis Ground
*External Reset

CONNECTOR DETAIL:

PIN	Function
1	CLK
2	DATA
3	RESET
4	GND
5	+5V

ELECTRICAL DATA:

Input Power	+5VDC @5ØØ mA. typ.
Serial Data Output	1 Start Bit, 8 Data Bits,

Table A-3. Video Monitor Specifications (AMDEK Model Video 300)

1.	Video Input	Composite Video Signal 1.0 $\pm$ 0.2Vp-p, Sync negative 0.3 Vp-p
2.	Input Impedance	75 ohm
3.	Scan Standard	Horizontal: 15750Hz Vertical: 50/60Hz
4.	Video Response	18MHz (-3dB)
5.	Deflection Linearity	Within 2%
6.	Controls:	
	Service	B+, H. width, V. Height, V. Lin, Focus, Sub-Brightness
	User	Brightness, Contrast, H. Hold, V. Hold
7.	CRT	12-Inch, 90° Deflection, P31 (Green)
8.	Semiconductors	Transistors 16 Diodes 20
9.	Power Source	AC 120 Volts/60Hz or 220 Volts/50Hz
10.	Power Consumption	28W
11.	Dimensions	14.5" (W) X 11.5" (H) X 13.7" (D)

Table A-3. (Continued)

12.	Carton Dimensions	17.48" (W) X 15.35" (H) X 17.Ø8" (D)
13.	Net Weight	17lbs.

Table A-4. Serial Pin Connectors

Console and Serial Port

Pin No.	Description		Function
1	GROUND	(GND)	
2	TRANSMIT DATA	(TXD)	OUTPUT
3	RECEIVE DATA	(RXD)	INPUT
4	REQUEST TO SEND	(RTS)	OUTPUT
5	CLEAR TO SEND	(CTS)	INPUT
6	DATA SET READY	(DSR)	INPUT
7	GROUND	(GND)	
8	DATA CARRIER DETECT	(DCD)	INPUT
20	DATA TERMINAL READY	(DTR)	OUTPUT
22	RING INDICATOR	(RI)	INPUT

Note: To provide display output on the console port, it is necessary to configure an RS232 cable with pins 2/3, 4/5 and 6/20 reversed (see Table A-6).

Table A-5. Parallel PINOUT Connections (Centronics Compatible)

Pin No.	Signal
1	- $\overline{\text{STB}}$
2-9	- DØ-D7
1Ø	- $\overline{\text{ACK}}$
11	- BUSY
32	- $\overline{\text{FAULT}}$
19,21,23,25,27	- GND

Note: A parallel connector cannot be connected to a serial printer.

Table A-6. Reversing Cable Configuration

MPC	Std. Serial Device
GND 1	1 GND
TXD 2	3 RXD
RXD 3	2 TXD
RTS 4	5 CTS
CTS 5	4 RTS
DSR 6	2Ø DTR
GND 7	7 GND
DTR 2Ø	6 DSR

APPENDIX B

MS-DOS DEFINITION OF DIRECTORY UTILITIES*

*Reprinted by permission of Microsoft, Inc.

MS-DOS DEFINITION OF DIRECTORY UTILITIES

CHKDSK	This function verifys the integrity of the directory structure. If any errors are found, the appropriate error message is displayed and corrective action is attempted. (MSDOS User's Guide Page 33)
COMMAND	This function is used to interface between the user and the underlying operating system. It allows the user to perform file management functions such as rename and delete, as well as to load and execute programs. (MSDOS User's Guide Chapter 2)
CREF	Is a crossreference utility used to create a crossreference listing from an assembly source listing. (MSCREF CrossReference Facility Manual)
DEBUG	A program used to provide a controlled testing environment for executable object files. (MSDOS User's Guide Chapter 5)
EDLIN	Performs intraline editing using the special editing keys that are also available at the MSDOS command level. (MSDOS User's Guide Chapter 4)

EXE2BIN	Used to convert .EXE files to .COM files. In general, only assembly language programs that have specifically formulated may undergo such conversions. (MSDOS User's Guide Chapter 3)
FILCOM	A file comparison program used to check for differences between files. Either text or binary files may be compared. (MSDOS User's Guide Chapter 6)
FORMAT	Is used to format disks so that they can be used with MSDOS. (MSDOS User's Guide Chapter 3, Pages 314)
HDBOOT.SYS	File loaded on the hard disk that allows the system to be booted from the hard disk.
LIB	The Microsoft MSLIB library manager used to create, maintain, and manipulate libraries of object files. (MSDOS Utility Software Package Ref Manual)
LINK	Used to link object files and object libraries to create executable .COM and .EXE files. (MSDOS Utility Software Package Ref Manual)
MASM	Microsoft's relocatable macro assembler for 8Ø86 and 8Ø88 microprocessors, MACRO86. (MSDOS Utility Software Package Ref Manual)

SYS	Used to transfer MSDOS.SYS and IO.SYS from a system disk to a formatted disk that does not contain the MS=DOS operating system on it. (MS-DOS User's Guide Chapter 3)

APPENDIX C

HARD DISK OPERATING INFORMATION

APPENDIX C: HARD DISK OPERATING INFORMATION

C.1 Introduction

This chapter explains how to operate the disk drive system via the controller. The host system communicates with the controller by sending commands and (in the case of write commands) data, and by receiving return codes and (in the case of read commands) returned data. The commands, data, and return codes are all transferred in bytes on the bidirectional bus lines data Ø-7 on the user interface. A handshake procedure (described in Section C.6) controls the transfer of the individual bytes.

C.2 Commands

The host system directs the activities of the controller by issuing commands. An individual command consists of one, two, or four bytes. The first byte (byte Ø) always identifies the command, and hence determines how many bytes of command information will be sent altogether.

C.2.1 Command Levels

There are two command levels designated levels Ø and 1. The first byte of each command string is called byte Ø. Bit 7 of byte Ø identifies the command level in binary representation.

Level Ø commands are commands for normal operation and are used in connection with logical sector addressing. Automatic remapping of bad sectors is enabled on level Ø.

Level 1 commands are used to configure the drive and for diagnostic purposes. Automatic remapping of bad sectors is not enabled in level 1.

C.2.2 Level Ø Commands (Normal Operation)

The format for a level Ø (4 byte) command is as follows:

Table C-1. Level Ø Commands

BYTE	BIT POSITION							
NUMBER	7	6	5	4	3	2	1	Ø
Ø	Ø	Ø	X	X	OP3	OP2	OP1	OPØ
1	L19	L18	L17	L16	Ø	DR2	DR1	DRØ
2	LØ7	LØ6	LØ5	LØ4	LØ3	LØ2	LØ1	LØØ
3	L15	L14	L13	L12	L11	L1Ø	LØ9	LØ8

X = Don't Care
(OP3-OPØ) = Operation Code
(DR2-DRØ) = Drive Address
(L19-LØØ) = Logical Sector Address

The following table lists the level Ø commands corresponding to particular values of the operation code:

Table C-2. Level Ø Command Values

Note	OP Code	Description
*	ØØØØ	NULL COMMAND. This command gives a ØØ return code.
*	ØØØ1	SECONDARY FORMAT. Format the drive (doing one write/read pass) but retaining the old bad sector map information. This command will not be accepted if the drive is in "format protect" mode.
	ØØ1Ø	READ SECTOR. Read a sector of data from the drive, and transfer that sector to the host.
	ØØ11	WRITE SECTOR WITH VERIFY. Transfer a sector of data from the host, write it on the disk, read the same sector, and check for CRC error.
	Ø1ØØ	ADD TO BSM. Add a sector to the bad sector map.
*	Ø1Ø1	RESET DRIVE. Do warm start reset of the controller hardware and software. Read the system sector into the controller buffer from the drive addressed in DR2-DRØ.
*	Ø11Ø	REZERO DRIVE.
*	Ø111	READ BUFFER. Transfer a sector of data from the controller buffer to the host system.

Table C-2. (Continued)

*	1000	WRITE BUFFER. Transfer a sector of data from the host system to the controller buffer.
	1001	NOT USED.
	1010	NOT USED.
	1011	WRITE SECTOR WITHOUT VERIFY. Transfer a sector of data from the host system, and write that sector on the disk. Do not verify.
	1100	FETCH SECTOR FROM DISK. Read a sector of data from the disk into the controller buffer.
	1101	STORE SECTOR ON DISK WITH VERIFY. Transfer a sector of data from the controller buffer, write it on the disk, read the same sector, and check for CRC error.
	1110	STORE SECTOR ON DISK WITHOUT VERIFY. Transfer a sector of data from the controller buffer, and write that sector on the disk. Do not verify.
*	1111	RETURN DRIVE TYPE.

Key to notes: * = A command that uses only two bytes.

As indicated in the above table, some of the commands do not involve a sector address and are issued using only two bytes from the normal four-byte sequence. When this is done, bits 4 through 7 of command byte 1 (the second command byte) are ignored by the controller.

In the case of a write command involving data transfer across the user interface, the host system sends the data immediately after sending the four command bytes.

After a command has been executed (or terminated because of an error condition), the controller sends a one-byte return code. In the case of a read command involving data transfer across the user interface, the return code is immediately followed by a sector of data sent by the controller to the host system. If, however, the return code shows a fatal error, the requested sector of data is not sent to the host. If the user desires this sector anyway and if the fatal error was of CRC-error origin, the user must immediately issue a read buffer command. The return codes are described in Section C.3. The following paragraphs give more detailed information concerning some of the commands.

Format Command

This command erases the entire disk as follows. First, the controller rezeros the disk drive (moves the head stack to cylinder zero and resets any error condition). Then the controller formats the drive in sectors with a data pattern of ØE5 (Hex). After writing the drive in this way, the controller reads all of the sectors on the drive. If an error is found, the corresponding sector is written and read a maximum of 16 times or until a valid rewrite is accomplished. If errors occur all 16 times, the corresponding sector is added to the bad sector map (BSM). After the entire reread process is completed on the drive, the BSM thus amended is written into the reserved area on the disk and becomes part of the system sector area.

During normal operation in level Ø, automatic skipping of defective sectors is performed entirely by the controller and is transparent to the host system.

The format command will not be accepted by the controller unless the drive is in format enable mode (Bit 3 of the configuration byte). If bit 3 of the configuration byte is a one, issuing a format command will cause the controller to return the "format protected" return code - (error code 9).

All drives are formatted at the factory. The user should not format any drive as part of an initial set up procedure. However, if an additional bad sector should develop (as evidenced by the appearance of hard errors, always at the same sector address), the format command could be used to ammend the BSM and thus remove the defective sector from further operations.

The user must always offload from the disk any data to be saved before issuing the format command. Remember also to put the drive in format enable mode before issuing a format command and in format protect mode before issuing regular read or write commands.

Read Sector Command

First the controller checks the sector address for validity (an appropriate error code is returned if the sector address is not valid). The cache is then checked for the presence of the desired sector. If found, it is checked for errors (unless already done), and if no errors are found, the return code and data are sent to the host. If the sector is not found in the cache, then that sector and several sectors sequentially following it are read from the disk (default number read is 8, may be changed using level 1 command, OP code 1111). If an error was found, then only the desired sector is re-read. To perform a physical read, the head stack is sent to the correct cylinder. The data is read from the selected sector (or sectors) into the appropriate cache buffers, and the return code is generated. If an error is detected, the controller automatically retries the read from the disk. When the command execution (including retries, if any) is complete, the return code is sent to the host system followed (if no fatal error has been detected) by the requested sector of data.

Write Sector With Verify Command

After the command and data have been transferred across the user interface, the controller checks the sector address for validity (an appropriate error code is returned if the sector address is invalid). If the deferred write bit is set in the configuration byte, then a "no error" return code is sent to the host to confirm the receipt of the data. This frees the host system to continue processing while the sector is physically being written by the hard disk system. If a non-fatal error occurs, it will not be reported. Fatal errors will be reported on the next command, but this will occur only in catastrophic circumstances.

To perform a physical write, the head stack is sent to the correct cylinder, and the data is written to the specified sector (the data is written to the disk from the controller buffer into which it has been transferred from the host system). The controller reads the sector just written to verify the integrity of the data. Finally, based on all available information, the controller sends a return code to the host system (unless already sent).

Write Sector Without Verify Command

This command executes in the same manner as the write sector with verify command except that the controller does not read the newly written sector.

Reset Drive Command

This command does a warm start reset of the controller hardware and software and then reads the system sector (DSB) and bad sector map (BSM) into the controller buffer and hence into the system drive status area.

C.2.3 Level 1 Commands (Diagnostic and Configuration)

The format for a level 1 command is a function of the particular command issued. However, byte Ø and bits 3-Ø of byte 1 are the same for all level 1 commands and are as follows:

Table C-3. Level 1 Commands

BYTE	BIT POSITION							
NUMBER	7	6	5	4	3	2	1	Ø
Ø	1	Ø	X	X	OP3	OP2	OP1	OPØ
1	A19	A18	A17	A16	Ø	DR2	DR1	DRØ
2	AØ7	AØ6	AØ5	AØ4	AØ3	AØ2	AØ1	AØØ
3	A15	A14	A13	A12	A11	A1Ø	AØ9	AØ8

X = Don't Care
(OP3-OPØ) = Operation Code
(DR2-DRØ) = Drive Address
(AØØ-A19) = Absolute Sector Address

The following table lists the level 1 commands corresponding to particular values of the operation code:

Table C-4. Level 1 Command Values

Note	OP Code	Description
	ØØØØ	NOT USED.
*	ØØØ1	PRIMARY FORMAT. Format the drive (doing one write/read pass). Destroy and rewrite bad sector map information.
	ØØ1Ø	DIAGNOSTIC READ. Read a sector of data from the drive at the specified absolute address. Transfer that sector to the host.
	ØØ11	DIAGNOSTIC WRITE WITH VERIFY. Transfer a sector of data from the host, write it on the disk at the specified absolute address. Read the same sector for CRC error.
	Ø1ØØ	ADD TO BSM. Add a sector to the bad sector map.
	Ø1Ø1	NOT USED.
	Ø11Ø	NOT USED.
	Ø111	READ FROM CONTROLLER DIRECTLY TO HOST.
	1ØØØ	WRITE TO CONTROLLER MEMORY DIRECTLY FROM HOST.

Table C-4. (Continued)

	1001	EXECUTE. Used in conjunction with write to controller RAM. Jumps to specific program location. These commands allow the user to alter controller functions.
*	1010	RETURN STATUS REPORT. Controller sends a string of bytes indicating status of drive.
*	1011	SEND BAD SECTOR MAP. Controller sends the bad sector map to the host.
*	1100	UNLOCK CONFIGURATION BYTE. Reset the "locked" bit in the drive configuration byte.
	1101	NOT USED.
*	1110	SET CONFIGURATION BYTE. Load a new configuration byte.
*	1111	SET READ SIZE. Set number of sectors to read in a physical disk access.

Key to notes: * = A command that uses only two bytes.

The following paragraphs describe the commands of level 1:

Primary Format Command

This command is used at the factory for the original process of formatting the drive. It is generally neither necessary nor advisable for the user to employ this command. Whereas the format command (level Ø and level 1) may add to the BSM, the primary format command destroys the existing BSM and writes a new one.

Prior to making the write/read pass, the primary format process loads the drive status block (DSB) with default values. It also clears the bad sector map (BSM).

There are two ways of adding new entries to the BSM. One is by using the level Ø format command. The other is by using the add to BSM command. The format command has the advantage of being a very simple command to issue. The add to BSM command has the advantage that it can be used to skip sectors that produce errors on a very intermittent basis. Care must be taken, however, to be sure that the proper addressing method is used to specify which sector is to be remapped. For details, see the discussion of the add to BSM command below.

Diagnostic Read Command

This command is similar to the level Ø read sector command except the sector address is an absolute address rather than a logical address. The absolute addresses begin with the first physical sector on the disk whereas the logical addresses begin with the first user sector. Thus the absolute address of a given sector is equal to 22 (Hex) plus the logical address of the same sector.

The diagnostic read command may be used to read the system sector. The system sector is written in the first 23 absolute sectors. When the diagnostic read command is given, a specific one of these system sectors may be read.

Diagnostic Write With Verify Command

This command is similar to the level Ø write sector with verify command except for the addressing differences as detailed in the above discussion of the "diagnostic read" command.

Add to BSM Command

This command is used to add a sector to the bad sector map. This command may be executed as a level Ø or level 1 command. If the level 1 command is given, the user must send the absolute address of the sector to be added. The format of the level 1 command is the same as that of the level Ø command.

Typically, evidence of a bad sector comes in the form of CRC errors occuring repeatedly at the same sector address. These errors can be logged and an add to BSM command can be issued to add the offending sector to the BSM. It is important, however, that the command level and sector remapping mode be the same when the add to BSM command is issued, as it was when the errors were detected. Remapping is enabled or disabled according to bit 4 of the configuration byte.

The bad sector map is stored to on a first come first served basis. All sector addresses are first checked to assure they are not already being remapped via the remapping facility. If the BSM is full, the function is terminated with a return code of fatal + BSM FULL error.

Read from Controller Memory Command

This command is used to read values presently in the controller memory and communicate them to the host system. The memory data is extracted from the controller on a byte-at-a-time basis. Each host command address is a single 8 bit byte of data represented by the address in command bytes 2 and 3 (the third and fourth bytes in a four byte command string). This address is in low byte/high byte format.

After fetching the data from memory, the return code will be sent via the user interface followed immediately by the 8 bits of controller memory data.

Write to Controller Memory Command

This command is used to modify controller memory locations with data from the host system. The addressing scheme is the same as the read from controller RAM command above and the byte-at-a-time restriction still remains. The host value to be written into controller memory follows the four byte command sequence from the host system. The return code response will be zero if the data is placed successfully and fatal + write protected (1ØØØ11Ø1) if unsuccessful. Unsuccessful functioning of this command can be a result of attempting to write into ROM occupied memory space or due to RAM-memory malfunctioning.

Execute Command

This command is used to vector controller CPU program execution to routines that are downloaded via the "write to controller memory" command. This command uses the same addressing convention as the two preceding commands. Also, the command will send a return code of zero prior to the vector being taken.

The routine must include a sequence to satisfy the all control functions, and to return to the command reception software for controller functioning to continue.

Return Status Report Command

This command sends a return code followed by a string of 6 bytes: (1) version of controller firmware, (2) drive type, (3) configuration byte, (4) number of sectors read in physical access, (5 and 6) number of remapped sectors (low byte, high byte).

Send Bad Sector Map

This command sends a return code followed by N groups of 3-byte sector addresses (where N is the number of remapped sectors). Each 3-byte address corresponds to the absolute address of a remapped (bad) sector (in high , low, middle byte order).

Set Read Size

This command accepts a byte from 1 through 2Ø (Hex) which determines how many sequential sectors are read whenever a physical disk read is done. This value defaults to 8 on a newly primary formatted drive.

Set Configuration Byte Command

This command is used to load the configuration byte with values sent by the user and is operative only when the locked bit (bit 7) of the configuration byte is equal to Ø. The following table shows the bit definitions (and default values) for the configuration byte:

Table C-5. Configuration Byte Bit Definitions

Bit #	Bit Name	Default
7	Locked	Ø
6	Deferred Write	Ø
5	Reserved	1
4	Sector Remapping Enabled	1
3	Format Protect	Ø
2	Retries Enabled	1
1	Write Protect	Ø
Ø	Reserved	1

If deferred write = 1, the return code will be sent immediately after the write data is received. Non-fatal errors will not be reported. Fatal errors will be reported on the next command.

If sector remapping = 1, the controller automatically remaps the bad sectors by adding them to the BSM.

If format protect = 1, the add skip, format, primary format, and load interlace table commands are disabled.

If retries enabled = 1, retries are enabled during all seek, read, and write verify operations.

If write protect = 1, the entire drive is write protected.

C.3 Return Codes

As explained above, the controller sends a one byte return code when a command execution has been completed (or terminated because of an error condition). The format for the return code is as follows:

Table C-6. Return Code Format

BIT POSITION

7	6	5	4	3	2	1	Ø
F	V	R	S	ER3	ER2	ER1	ERØ

F = Fatal error (command not successfully completed)

V = Verify error (error detected during the verify portion of a write sector with verify or a store sector on disk with verify command).

R = Retry (retry was attempted - if R=1 and F=Ø, the retry was successful.

S = Seek error (error detected during seek check).

ER3-ERØ = Error code (see table below).

If all eight bits of the return code are Ø, no error occurred.

The following table lists the conditions corresponding to particular values of the error code (ER3-ERØ):

Table C-7. Error Code Values

Note	Error Code	Description
	ØØØØ	BAD SECTOR MAP IS FULL.
	ØØØ1	READ/WRITE FAULT.
*	ØØ1Ø	SERVO FAULT DURING SEEK.
*	ØØ11	HEADER MISMATCH (CYLINDER OR HEAD).
	Ø1ØØ	UNIMPLEMENTED OR ILLEGAL COMMAND.
*	Ø1Ø1	SERVO FAULT DURING REZERO.
*	Ø11Ø	DISK TRANSFER (READ/WRITE) TIMEOUT.
	Ø111	SYSTEM SECTOR BAD.
*	1ØØØ	SERVO FAULT DURING WRITE.
	1ØØ1	FORMAT PROTECTED (FORMAT ISSUED WITHOUT FORMAT ENABLE)
*	1Ø1Ø	BAD SECTOR MAP ENTRY MADE.
*	1Ø11	DATA CRC ERROR DURING READ.
*	11ØØ	HEADER MISMATCH (SECTOR).
	11Ø1	WRITE PROTECTED (WRITE COMMAND WITH WRITE PROTECT).

Table C-7. (Continued)

111Ø	ILLEGAL ADDRESS
1111	DRIVE DOES NOT ACKNOWLEDGE THE COMMAND.

Key to Notes: * = Retries are attempted on these error conditions.

The following table lists the possible error conditions that may occur during execution of the level Ø commands. For compactness, HEX notation is used:

Table C-8. Level Ø Command Error Conditions

COMMAND OP CODE	POSSIBLE ERROR CODE Ø	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Ø	X															X
1	X		X	X	X	X	X		X	X	X	X	X	X		X
2	X	X	X	X		X	X				X	X	X		X	X
3	X	X	X	X		X	X		X		X	X	X	X	X	X
4	X	X	X		X											X
5	X					X	X	X			X	X	X			X
6	X					X										X
7	X															X
8	X															X
9					X											X
A					X											X
B	X	X	X	X		X	X		X				X	X	X	X
C	X	X	X	X		X	X				X	X	X		X	X
D	X	X	X	X		X	X		X		X	X	X	X	X	X
E	X	X	X	X		X	X		X				X	X	X	X
F	NO RETURN CODE															

Notes: Commands 1 and 4 return error code 4 if the BSM is full.

Command F does not produce a return code as such. However, a byte is returned in place of the return code which depicts the drive type according to this code:

Ø1 =	SHUGART 8" SA1ØØ4 SYSTEM	8.5 MBYTE
Ø2 =	TANDON 5.25" TM6Ø2 SYSTEM	5.Ø MBYTE
Ø3 =	TANDON 5.25" TM6Ø3 SYSTEM	1Ø.Ø MBYTE
Ø4 =	QUANTUM 8" Q2Ø1Ø SYSTEM	8.5 MBYTE
Ø5 =	QUANTUM 8" Q2Ø2Ø SYSTEM	17.5 MBYTE
Ø6 =	TANDON 5.25" TM5Ø3 SYSTEM	15.Ø MBYTE
Ø7 =	QUANTUM 8" Q2Ø4Ø SYSTEM	35.3 MBYTE

Ø8-ØF = EXPANSION ASSIGNMENTS

C.4 Start-Up Procedures

C.4.1 Power Up

There is no power-up sequencing required on the part of the user. When power is applied to the drive, the drive will operate normally when up to speed.

C.4.2 Initialization

There are no initialization procedures required after power up. On a one time basis, the configuration byte should be set if the factory programmed default value is not acceptable to the user. The factory installed value for this byte is 35 (Hex). Also, the set read size may be used to select the number of sectors to read in a physical access, also on a one time basis. The default value is 8.

C.5 Normal Operation

C.5.1 Addressing Methods

The drive address is set up by hardwiring on the disk drive electronics. The user specifies the address of the drive selected by sending the drive address on the data bus lines during the second byte (byte 1) of the command.

Within the selected drive, an individual sector can be addressed. The user has a choice of logical or absolute addressing. For logical sector addressing, level Ø commands are used, and for absolute sector addressing, level 1 commands are used.

C.5.2 Data Transfer Method

Data is transferred as byte parallel data in an asynchronous manner using a handshake process as explained in Section C.5.3 and C.5.4.

Because the controller has enough storage for an entire sector of data, there are no minimum data rate requirements that need to be met by the host system.

C.5.3 Issuing Commands to the Controller

The formats for the various commands are discussed in Section C.2. When the user issues a command to the controller, the following sequence of events takes place on the user interface:

1. The controller indicates its readiness to accept a command by holding READY and -BUS DIR both high.

2. The user places command byte Ø on the data Ø-7 lines, and then generates a negative pulse on the -STROBE line.

3. The controller accepts the data and generates a negative pulse on the READY line.

Steps 2 and 3 are repeated for each command byte in turn, and then (in case of a write command involving data transfer across the user interface) steps 2 and 3 are repeated for each byte of data as well. After the last command (or data) byte has been transferred, READY remains low during command execution.

When the controller is ready to send the return code, the controller lowers -BUS DIR (READY is already low). Next, the controller places the return code on the data Ø-7 lines and then raises READY (BUS DIR remains low). The user responds by generating a negative pulse on the -STROBE line. The rising edge of this pulse indicates to the controller that the return code has been gated into the host system. The controller then lowers READY.

If there is no data to be returned to the host system, the controller raises -BUS DIR and then raises READY indicating to the host that the user interface is available for another operation. If, however, the command is a read command involving such a data transfer, the following sequence occurs for each byte of data to be returned:

1. The controller lowers READY (-BUS DIR is already low).

2. The controller places the byte of data on the data Ø-7 lines and then raises READY (-BUS DIR remains low).

3. The user generates a negative pulse on the -STROBE line. The rising edge of this pulse

indicates to the controller that the byte of data has been gated into the host system. The controller responds by lowering READY.

When all the data has been transferred, the controller raises BUS DIR and then raises READY indicating to the host system that the user interface is available for another operation.

C.5.4 Reading/Writing Data

Writing data to the disk drive system is done by issuing a write sector with verify or write sector without verify command followed immediately by the sector of data to be written. The controller will then respond as described in Section A.5.3. There is no difference (with regard to the handshaking procedure on the user interface) between command bytes and data bytes.

C.5.5 Interrupts

There are no specifically designated interrupt lines on the user interface. The falling edge of -BUS DIR can be used as an interrupt to signal the host system that the selected drive has completed its assigned operation and a return code is available.

C.6 Shut-Down Procedures

C.6.1 Normal

Power may be shut off whenever a command is not in process. No special power-down sequence is required.

C.6.2 Power Fail

If power should fail during a command execution, only the current command would be affected. The user could retry that command when power is restored. Aside from that, there are no adverse affects and no danger to data integrity resulting from a power failure.

APPENDIX D

ROM BIOS INTERFACE SPECIFICATION

APPENDIX D. ROM BIOS INTERFACE SPECIFICATION

D.1 General

The MPC contains a read only memory basic input output system (ROM BIOS) which permits the user program to access most system devices without regard for the physical requirements of the device. Each BIOS function is called by a pre-defined software interrupt to which the necessary parameters are passed through 8Ø88 registers. For interrupts which perform multiple functions, register AH passes the function number. In general, any register (except AX and FLAGS) which does not return a result, will be returned unchanged.

The numeric values used in the following discussion are in decimal unless followed by letter "H" in which case they are hexadecimal.

Table D-1. Summary of Reserved Interrupts

Interrupt	Function	Type
INT Ø1H	Single Step ISR	Hardware
INT Ø2H	Nonmaskable ISR	Hardware
INT Ø3H	Breakpoint Trap	Software
INT Ø5H	Print Screen	Software
INT Ø8H	Timer Tick ISR	Hardware
INT Ø9H	Keyboard ISR	Hardware
INT ØBH	Dumb Terminal ISR	Hardware
INT ØEH	Floppy Disk Controller	Hardware
INT 1ØH	Video Interface	Software
INT 11H	Equipment Report	Software
INT 12H	Memory Size Report	Software
INT 13H	Diskette I/O	Software
INT 14H	Serial Communications	Software
INT 16H	Keyboard I/O	Software
INT 17H	Parallel Printer	Software
INT 19H	System Bootstrap	Software

Table D-1. (Continued)

INT 1AH	Read/Set Time of Day	Software
INT 1BH	Keyboard Break	User Supplied Software
INT 1CH	Timer Tick	User Supplied Software
INT 1DH	Video Parameters	Pointer
INT 1EH	Diskette Parameters	Pointer
INT 1FH	Character Generator Table	Pointer

D.2 Software Interrupts

INT Ø5H - PRINT SCREEN

This interrupt copies the screen to printer #Ø. No arguments are passed through 8Ø88 registers for this function. Byte 5ØH:Ø holds the status of the print operation where: Ø = DONE, 1 = IN PROGRESS, OFFH = ERROR. Interrupts are assumed to be enabled during printing, and any interrupts which occur may examine the status using Ø:5ØH.

INT 1ØH - VIDEO

This interrupt provides an interface to the CRT for the following functions:

AH = Ø Set Display Mode
- AL = Ø 4Ø X 25 Black & White
- AL = 1 4Ø X 25 Color
- AL = 2 8Ø X 25 Black & White
- AL = 3 8Ø X 25 Color
- AL = 4 Graphics 32Ø X 2ØØ Color
- AL = 5 Graphics 32Ø X 2ØØ Black & White
- AL = 6 Graphics 64Ø X 2ØØ Black & White
- AL = 7 8Ø X 25 Monochrome

Note: Modes Ø through 6 use color graphics video board. Mode 7 uses monochrome video board.

AH = 1 Set Cursor Type

CH (Bits 4-Ø) = Starting line for cursor
CL (Bits 4-Ø) = Ending line for cursor

Note: Setting bits 5 or 6 of CH will cause erratic or missing cursor.

AH = 2 Set Cursor Position

DH,DL = Cursor position (row, column), upper left is Ø,Ø.
BH = Page number, must be Ø if graphics mode is selected.

AH = 3 Read Cursor Position

BH = Page number, must be Ø if graphics mode is selected.

Values Returned:

DH,DL = Cursor position (row,column), upper left is Ø,Ø
CH,CL = Cursor mode

AH = 4 Read Light Pen Position

Values Returned:

AH = Ø if light pen not pressed, not triggered
AH = 1 if registers contain light pen position
DH,DL = Row,column of light pen (if AH = 1)
CH = Raster line (Ø-199, if AH = 1)
BX = Pixel column (Ø-319 or Ø-639, if AH =1)

AH = 5 Select Active Display Page

AL = Page number (Ø-7 if mode Ø or 1, Ø-3 if mode 2 or 3)

AH = 6 Scroll Window of Active Page Up, Blank New Bottom Line

Al = Number of lines (if AL = Ø, blank entire window)
CH,CL = Row,column of upper left corner of window
DH,DL = Row,column of lower right corner of window
BH = Attribute to be used on new blank line

AH = 7 Scroll Window of Active Page Down, Blank New Top Line

AL = Number of lines (if AL = Ø, blank entire window)

CH,CL = Row,column of upper left corner of window

DH,DL = Row,column of lower right corner of window

BH = Attribute to be used on new blank line

AH = 8 Read Character and Attribute at Cursor

BH = Page number, must be Ø if graphics mode is selected

Values Returned:

AL = Character value

AH = Attribute value (invalid if graphics mode)

AH = 9 Write Character and Attribute at Cursor

BH = Page number, must be Ø if graphics mode is selected

CX = Number of characters to be written

AL = Character

BL = Attribute (alpha mode), or color (graphics mode)

Note: If Bit 7 of BL = 1 in graphics mode, the color will be exclusive or'd with the current color of the character.

AH = 1Ø Write Character Only at Cursor

BH = Page number, must be Ø if graphics mode is selected

CX = Number of characters to be written

AL = Character

AH = 11 Select Color Pallette (Mode 4 Only)

BH = Ø Define the background color (BL=Ø-15)

BH = 1 Select the color pallette

BL = Ø Selects Green (1), Red (2), Yellow (3)

BL = 1 Selects Cyan (1), Magenta (2), White (3)

Note: In modes Ø-3, BH=Ø defines the border color (BL = Ø-31) where BL = 16-31 is the high intensity background set.

AH = 12 Write Dot (Modes 4-6 Only)
- DX = Row number
- CX = Column number
- AL = Color value

Note: If Bit 7 of BL = 1, the color will be exclusive or'd with the current color of the character.

AH = 13 Read Dot (Modes 4-6 Only)
- DX = Row number
- CX = Column number

Value Returned:
- AL = Color value

AH = 14 Write Character (Teletype Conventions)
- AL = Character value
- BL = Foreground color or character (if graphics mode)
- BH = Page number (if alpha mode)

Note: This function emulates a teletype by writing a character to the current cursor position, then moving the cursor one position to the right. Line wrap-around at right margin is provided. Control codes supported are:
- SP (2ØH) = Write a blank space
- CR (ØDH) = Cursor to left margin of current line
- LF (ØAH) = Cursor down one line, scroll up if at bottom
- BS (Ø8H) = Cursor left one character (non-destructive)
- BEL (Ø7H)= Sound beeper

AH = 15 Read Video State
Values Returned:
AL = Video mode
AH = Screen width (4Ø or 8Ø)
BH = Active page number

INT 11H - Equipment Report

This interrupt reports the configuration of the MPC.

Value Returned:

AX = Equipment configuration word, defined as follows:
- Bit 15,14 = Number of printers attached
- Bit 13 = Not used
- Bit 12 = Game interface attached
- Bit 11,1Ø,9 = Number of RS232 cards attached
- Bit 8 = Unused
- Bit 7,6 = Number of floppy disk drives attached
  - ØØ = 1 Drive, Ø1 = 2 Drives,
  - 1Ø = 3 Drives, 11 = 4 Drives
- Bit 5,4 = Initial video mode
  - ØØ = Dumb terminal, Ø1 = 4Ø X 25 color card
  - 1Ø = 8Ø X 25 color, card 11 = 8Ø X 25 monochrome card
- Bit 3,2,1 = Not used
- Bit Ø = Type of expansion floppy disk (if installed)
  - Ø = 8" drives, 1 = 5" drives

Note: In hard disk systems, the number of hard disk drives can be determined by examining the byte at 4ØH:69H (HD_INSTALL). Bits 2,1,Ø contain the highest drive address which the diskette I/O interrupt (INT 13H) will accept. Since the floppy disks are assigned to drive number Ø-3, subtract 3 from this value to obtain the number of hard disks installed. Bits 7,6 indicate the presence of each of two possible drives on the internal controller and Bits 5,4 for the external controller, using the hard disk expansion interface board. If Bit 7 or 6 = 1, then floppy drive 1 is assumed not present and the total number of floppy drives given by INT 11H would be limited to 3, assuming 2 external drives are installed.

INT 12H - Memory Size Report

This interrupt reports the size of contiguous memory in the system.

Value Returned:

AX = Number of 1K (1Ø24) byte blocks of contiguous memory which exist, starting from Ø:Ø. This value is not dependent on switch settings on the main printed circuit board.

INT 13H - Diskette I/O

This interrupt performs all data transfers between the floppy or hard disk and the system memory. It also provides a track format function for the floppy disks.

Floppy disks are numbered Ø-3, depending on their physical location in the system: Ø,1 are internal drives, and 2,3 are connected to floppy disk expansion board. Bit Ø of AX returned by INT 11H indicates the type of external drive used: Ø = 8", 1 = 5". Bit 3 of memory byte 4ØH:69H (HD_INSTALL) indicates the presence of the floppy disk expansion (Drives 2,3). Drive number Ø is the floppy disk bootstrap drive.

Hard disks are assigned sequential numbers, beginning with 4, depending on the number of drives installed. The highest drive number installed is given by memory byte 4ØH:69H Bits 2,1,Ø. Drive number 4 is the hard disk bootstrap drive.

Certain drive parameters must be defined for INT 13H. The vector location corresponding to INT 1EH must point to a disk parameter table (see description of INT 1EH).

AH = Ø Reset the Diskette System

Note: Brings the disk system to an initialized state. Recalibration will be done on the first access to each drive following this function. This function must be performed before the next use of the disk system whenever a disk error occurs.

AH = 1 Read Disk Status
Values Returned:
Carry = 1 if error
AL = Error status
ØØH = No error
Ø1H = Unrecognized command
Ø2H = Address mark not found
Ø3H = Write protected diskette
Ø4H = Sector not found
Ø8H = DMA overrun
Ø9H = Attempt to DMA across 64K boundary
1ØH = CRC error on diskette read
2ØH = Disk controller failure
4ØH = Seek failed
8ØH = No response from disk system within time allowed

Note: Error status bits may be combined by logical ORing when multiple errors occur.

AH = 2 Read Sectors from Disk to Memory
AH = 3 Write Sectors from Memory to Disk
AH = 4 Verify Sectors from Disk
AL = Number of Sectors (1-8)
ES:BX = Address of buffer for disk data (not

required if AH = 4)
CH = Cylinder Number
CL = Beginning Sector Number
DH = Head Number
DL = Drive Number (Ø-7)

Note: For hard disk drives, DH,CH,CL define a logical sector number beginning with Ø,Ø,Ø

Values Returned:
Same as for read disk status command

Note: If an error is reported by the diskette I/O code, the user should reset the system (INT 13H with AH = Ø) then retry the desired function. Since no motor start up (or head load) delay is generated, it may be necessary to allow 4 attempts on a read or verify command while motor spins up.

AH = 5 Format a Track on Floppy Disk
AL = Number of sectors on track
ES:BX = Address of track descriptor table
CH = Cylinder number
DH = Head number
DL = Drive number (Ø-3)

Values Returned:
Same as for read disk status command

Note: Track descriptor table is composed of 4 bytes (C,H,R,N) for the I.D. field of each sector on the track in physical order:
C = Cylinder number
H = Head number
R = Sector number
N = Sector length code (Ø=128, 1=256, 2=512, 3=1Ø24)

INT 14H - Serial Communications

The interrupt provides an interface to the RS-232 type serial interfaces in the system.

AH = Ø Initialize the Communications Port
DX = Number of serial port (Ø-3)
AL = Initialization parameters
Bit 7,6,5 = Baud Rate

ØØØ = 19.2 Kilobaud	1ØØ = 12ØØ Baud
ØØ1 = 15Ø Baud	1Ø1 = 24ØØ Baud
Ø1Ø = 3ØØ Baud	11Ø = 48ØØ Baud
Ø11 = 6ØØ Baud	111 = 96ØØ Baud

Bit 4,3 = Parity Type
ØØ or 1Ø = None
Ø1 = Odd
11 = Even
Bit 2 = Stop Bits (Ø=1 Bit, 1=2 Bits)
Bit 1,2 = Word Length (1Ø=7 Bits, 11=8 Bits)
Values Returned:
Same as for return port status command

AH = 1 Send Character
AL = Character value
DX = Number of serial port (Ø-3)
Values Returned:
AH = Status of operation
Bit 7 = Unable to transmit
Bit 6-Ø = Same as for return port status command

AH = 2 Receive Character
DX = Number of serial port (Ø-3)
Values Returned:
AL = Character value
AH = Status of operation
Bit 7 = No data set ready received
Bit4-1= Same as for return port status command

AH = 3 Return Port Status
DX = Number of serial port (Ø-3)
AH = Line control status
Bit 7 = Time out
Bit 6 = Transmitter shift register empty
Bit 5 = Transmitter holding register empty
Bit 4 = Break detect

Bit 3 = Framing error
Bit 2 = Parity error
Bit 1 = Overrun error
Bit Ø = Receiver data ready

AL = MODEM status

Bit 7 = Receive line detect (data carrier detect)
Bit 6 = Ring indicator
Bit 5 = Data set ready
Bit 4 = Clear to send
Bit 3 = Receive line signal detect changed
Bit 2 = Trailing edge of ring indicator
Bit 1 = Data set ready changed
Bit Ø = Clear to send changed

INT 16H - Keyboard I/O

This interrupt provides an interface to the detachable keyboard.

AH = Ø Read Character

Values Returned:

AL = ASCII value or Ø
AH = Scan code of key pushed (if AL = ASCII)
Extended code (if AL = Ø)

AH = 1 Read Status

Values Returned:

Zero Flag = Ø if a character is available
AX = Same as in read character if Z=Ø, character returned if AX remains in buffer.

AH = 2 Return Shift Status

Value Returned:

AL = Keyboard Status

Bit Ø = Right shift depressed
Bit 1 = Left shift depressed
Bit 2 = Control depressed
Bit 3 = Alternate depressed
Bit 4 = Scroll lock toggled
Bit 5 = NUM lock toggled

Bit 6 = Caps lock toggled
Bit 7 = Insert state active

Extended Code	Function
3	NUL Character
15	←
16-25	ALT Q,W,E,R,T,U,I,O,P
3Ø-38	ALT A,S,D,F,G,H,J,K,L
44-5Ø	ALT Z,X,C,V,B,N,M
59-68	F1-F1Ø Function Keys Base Case
71	Home
72	↑
73	Page Up & Home Cursor
75	←
77	→
79	End
8Ø	↓
81	Page Down & Home Cursor
82	INS
83	DEL
84-93	F11-F2Ø (Upper Case F1-F1Ø)
94-1Ø3	F21-F3Ø (CTRL F1-F1Ø)
1Ø4-113	F31-F4Ø (ALT F1-F1Ø)
114	CTRL PRTSC (Start/Stop Echo to Printer) Key 55
115	CTRL ← Reverse Word
116	CTRL → Advance Word
117	CTRL END Erase EOL
118	CTRL PG DN Erase EOS
119	CTRL HOME Clear Screen and Home
12Ø-131	ALT 1,2,3,4,5,6,7,8,9,Ø,-, = (Keys 2-13)
132	CTRL PG UP TOP 25Lines of Text & Home Cursor

INT 17H - Parallel Printer

This interrupt provides an interface to the parallel printer devices.

AH = Ø Print Character
  AL = ASCII character
  DX = Printer number (Ø-3)
 Value Returned:
  AH = Printer status

Bit 7 = Not Busy	Bit 3 = I/O Error
Bit 6 = Acknowledge	Bit 2 = Not Used
Bit 5 = Out of Paper	Bit 1 = Not Used
Bit 4 = Selected	Bit Ø = Time Out Error

AH = 1 Initialize Printer Port
 Value Returned:
  Same as for print character command

AH = 2 Read Printer Status
 Value Returned:
  Same as for print character command

INT 19H - System Bootstrap

This interrupt boots the system from floppy disk drive Ø. On hard disk systems, if no disk is present in drive Ø, hard disk drive 4 will be used. In either case, the boot sector is read from the disk, loaded into memory at Ø:7CØØH and control transferred to it at that address.

No parameters are passed through registers.

This interrupt is automatically invoked by the system initialization code in the ROM BIOS.

INT 1AH - Read/Set Time of Day

This interrupt allows the time of day clock to be read or set.

AH = Ø Read Time of Day
Values Returned:
CX = High order word of time of day count
DX = Low order word of time of day count
AL = Ø if day has not changed since last read

Note: The time count in CX,DX runs at the rate of 18.2Ø65 Hz or 54.9254 milliseconds/count.

AH = 1 Set Time of Day
CX = High order word of time of day count
DX = Low order word of time of day count

Note: Time count is initially set to Ø when MPC is reset or powered on.

D.3 User Supplied Routines

INT 1BH - Keyboard Break

Control will vector to this interrupt when a break is commanded from the keyboard. The ROM BIOS initializes this vector to point to the cold start routine.

INT 1CH - Timer Tick

Control will vector to this interrupt when the timer interrupt occurs. The ROM BIOS initializes this vector to point to a null interrupt service routine.

D.4 Pointers

INT 1DH - Video Parameters

This vector points to a table of video intialization parameters for the Motorola 6845 CRT controller chip on the color and monochrome video boards. The tables consist of the data to be output to the CRT controller's 16 registers, RØ-R15. All four strings must be reproduced to maintain all possible modes of operation. The vector initially points to the following table in the ROM BIOS:

```
DB  38H, 28H, 2DH, ØAH, 1FH, Ø6H, 19H, 1CH  4ØX25
DB  Ø2H, Ø7H, Ø6H, Ø7H, ØØH, ØØH, ØØH, ØØH  Color

DB  71H, 5ØH, 5AH, ØAH, 1FH, Ø6H, 19H, 1CH  8ØX25
DB  Ø2H, Ø7H, Ø6H, Ø7H, ØØH, ØØH, ØØH, ØØH  Color

DB  38H, 28H, 2DH, ØAH, 7FH, Ø6H, 64H, 7ØH  Color
DB  Ø2H, Ø1H, Ø6H, Ø7H, ØØH, ØØH, ØØH, ØØH Graphics

DB  61H, 5ØH, 52H, ØFH, 19H, Ø6H, 19H, 19H  8ØX25
DB  Ø2H, ØDH, ØBH, ØCH, ØØH, ØØH, ØØH, ØØH Monochrome
```

INT 1EH - Diskette Parameters

This vector points to a table of parameters used for generating command strings to the INTEL 8272 floppy disk controller. If floppy disk drives of various types are to be used, this vector must point to an appropriate table when the diskette I/O function is performed. The vector initially points to the following table in the ROM BIOS:

Table Data		Meaning	8272 Command
DB	CFH	SRT = 12, HUT = 15	Specify
DB	Ø2H	HLT = 1, ND = Ø	Specify
DB	37	Motor Turn off Delay [Ticks]	
DB	2	Sector Length Code (N)	RD/WR/FMT
DB	8	End of Track (EOT)	RD/WR
DB	42	RD/WR Gap Length (GPL)	RD/WR
DB	FFH	Data Length (DTL)	RD/WR
DB	8Ø	Format Gap Length (GPL)	FORMAT
DB	F6H	Data Fill Value (D)	FORMAT
DB	25	Head Settle Time [ms.]	
DB	4	Motor Start Time [1/8 sec.]	

INT 1FH - Character Generator Table

This points to a user supplied extension of the character generator table for graphics video modes. The user may define 8X8 graphics patterns corresponding to character values 128-255 by pointing this vector to a table of 1K bytes of pixel data. Each pattern is defined by 8 bytes which describe the lit pixels in each row (from top to bottom) of the 8X8 block. (Example: A value of ØØØØØØ11 (binary) will light the two rightmost pixels.)

The ROM BIOS intializes this vector to Ø:Ø, which indicates that character values 128-255 are not defined.

APPENDIX E

MPC ROM MONITOR - TEST ROUTINES AND SOFTWARE FRONT PANEL

APPENDIX E: MPC ROM MONITOR - TEST ROUTINES AND SOFTWARE FRONT PANEL

E.1 General

The MPC contains a read only memory based monitor (ROM monitor) which allows the user to test various devices in the system and which can serve as a software front panel for the computer. These routines co-exist with the basic input output system routines (ROM BIOS) in the MPC's read only memory. (See Appendix D for a definition of the user interface to the ROM BIOS.)

To use the ROM monitor, type [ESCAPE] in response to the "TEST MEMORY?" prompt at system start up time. To interrupt a booted system, typing [CTRL-ALT-INSERT] (or [ESC] then [BREAK] on the dumb terminal) will pass control to the monitor. The following menu of functions will appear on the console:

Table E-1. ROM Monitor Menu

B[a]	[set] Brkpoint	Pa[,d]	[mod] Port
Da[,c]	[list] Memory	R[i,d]	[mod] Registers
Fa,c	Fill Memory	S[a][,c]	Step
G[a]	Go	Ta,c	Test Memory
M	Menu	Y[a,c]	Disk
E	Echo	K	Kybd/Prntr Test
Q[c]	Test Disk	Ii	Interrupt

a = Address, d = Data, c = Count, i = I.D., []= Option

The monitor will then prompt the user for a command by printing a "#" at the left margin of the screen.

The user must then type a function letter (B,D,E,F,G,I,K,M,P,Q,R,S,T,Y) followed immediately by the arguments required by that function. Separate the arguments in the command line by a comma ",". Do not enter any spaces in a command line. When the command line is fully entered, type [RETURN] to execute the command. Type "." to abort a command before the command line is fully entered. When entering numeric values, use hexadecimal; to correct a number entered incorrectly, retype it as a four digit number (using leading zeros as necessary) since the monitor will use only the last four digits entered.

The menu will not be displayed again unless the user requests it using the "M" command.

E.2 Monitor Commands

The remainder of this section is a description of each monitor command in alphabetical order.

(B) Set/Display Breakpoint

This command sets a breakpoint in a RAM-based program for debugging purposes. It will be set relative to the current code segment (CS) in the user registers. To clear the breakpoint, set the breakpoint at an unused address, such as FFFF. Use the "R" command to select the desired CS.

Examples:

B	Displays Current Breakpoint
B1234	Sets Breakpoint at CS:1234H

(D) Display/Modify Memory

This command operates in two modes. If a single argument is entered, the modify mode is selected; if both arguments are entered, the display mode is selected. This command addresses memory relative to the user data segment (DS). Use the "R" command to select the desired DS.

In modify mode, the contents of the selected address are displayed to which the user may respond in one of four ways:

1. Enter new data and type [RETURN] to modify memory.
2. Enter only [RETURN] to advance to next address.
3. Enter "^ " to regress to previous address.
4. Enter "." to exit the comand.

In display mode, the command line specifies the starting address and the number of bytes to display. This will produce a memory listing with the character equivalents of each byte printed at the end of each line.

(E) Echo Test (Serial Port Test)

This command tests the "SERIAL" port located on the rear panel of the MPC. To run this test, an RS-232 compatible serial terminal must be attached to this port. When the "E" command is entered on the console, this function will echo all entries made on the serial device's keyboard to the serial device's screen.

To end the test, type "." on the console, then type any key on the serial device.

(F) Fill Memory

This command fills a range of memory with a constant data value. The first argrument specifies the starting address (relative to the user DS), and the second argument specifies how many locations beyond the starting address to fill. The contents of the specified address is the data value used for the fill. Use the "R" command to select the desired DS. Example:

F8ØØ,5   Fills locations DS:8Ø1H thru DS:8Ø5H with data from location DS:8ØØH

(G) Go (Execute User Program)

This command begins or resumes execution of a user program. The 8Ø88 registers are loaded with the current user register values, then control is transferred to the user program. The ROM monitor will regain control of the CPU if a previously set breakpoint is encountered. Use the "R" command to select the desired code segment (CS). Examples:

G   Resume Execution (Begin at User CS:IP)
G6789   Begin Executing at CS:6789H

(I) Interrupt

This command executes the specified software interrupt. On completion, it displays the CPU registers returned by the interrupt. Use the "R" command to set the registers to the desired values prior to executing the interrupt.

(K) Keyboard/Printer Test

This command tests the keyboard and printer ports on the rear panel of the MPC. To run this test, the console must be a dumb terminal. Connect a keyboard to the keyboard port, and connect a Centronics compatible parallel printer to the parallel port. When the "K" command is entered on the console (dumb terminal), this function will send all codes typed on the detachable keyboard to the printer. Note that the scan codes generated by the keyboard are not ASCII, so the printout will be different from the keys struck.

Also, the printer will only print a line when a [CR] code (ASCII ØDH) is received. It will therefore be idle until this code is generated by the keyboard. Refer to the keyboard diagram (Figure 2-3) for a definition of the scan code for each key. Normally, to run this test, type the sequence B N M . , / <Right Shift Key> <PRT SC> <ALT> <Space Bar> = 9 and the printer will print:

Ø123456789

The final key of the sequence "9" must be from the upper row of keys, not the numeric pad. This key generates a line feed and may be pressed several times to advance the paper.

To end the test, type "." on the console (dumb terminal).

(M) Display Menu

This command displays the menu of functions to be used as a reference. No arguments are used in this command.

(P) Port Input/Output

This command allows the user to input data or output data to and I/O port in the system. The first argument on the command line specifies the port address, and the second argument (if present) specifies the data to be output. If the second argument is absent, the port will be input.

Examples:

P39,18	Output 18H to Port 39H
P78	Input Data from Port 78H and Display
P	Input Data from Last Port Addressed
P,29	Output 29H to Last Port Addressed

(Q) Random Disk Read/Write Test (QWAVER)

This command performs random read and write sector commands to the desired floppy disk. If an argument is entered, it specifies how many read/write cycles will be performed; otherwise, the test will run until "." is typed. Any errors which occur will be displayed. See "Y" command description for required setup and error interpretation. As an audible confirmation of the test, the "S" key may be typed to enable or disable the speaker.

(R) Register Display/Modify

This command allows the user to display or modify any of the 8Ø88 register for use by user programs and for data segment (DS) of the monitor commands which reference memory. Each register (including the flag register) is numbered, as follows:

IP=#1	AX=#2	BX=#3	CX=#4	DX=#5	SI=#6	DI=#7
SP=#8	BP=#9	CS=#A	DS=#B	SS=#C	ES=#D	Flags=#E

To display the current registers, enter no arguments on the command line. To modify a register, the first argument gives the register I.D., and the second gives the new value. Examples:

R	Displays Registers (and I.D.'s for Reference)
RB,1ØØ	Put 1ØØH in DS Register

(S) Single Step

This command executes one or more instructions from the user program. The first argument specifies the address of the instruction, and the second argument is the number of steps desired. If the first argument is missing, the monitor will use the current CS:IP to resume execution. If the second argument is missing, one step will be done. Use the "R" command to select a new code segment (CS), if required. Each use of the "S" command will display the current registers, and the new registers after each step.

Examples:

S	Execute Next Instruction
S,2	Execute 2 Intructions from Current CS:IP
S4ØØØ	Execute 1 Instruction from CS:4ØØØH
S8Ø,1Ø	Execute 16 (1ØH) Instructions from CS:8ØH

(T) Test Memory

This command tests a range of memory. The command line specifies the starting address, and the number of bytes to test. Specify a count of Ø to test all 64K bytes of memory in the selected segment. Use the "R" command to load the DS register with the proper memory segment prior to execution of this command.

If a memory error occurs, the address range of the error and the written and read data values will be displayed. As an audible confirmation of the test, the "S" key may be typed to enable or disable the speaker.

(Y) Disk Functions

This command allows various floppy disk functions to be performed. If a read or write function is to be done, the command line specifies the address of the disk buffer in memory (off of the current DS), and the number of sectors to tranfer; otherwise, this function takes no arguments. Use the "R" command to select the desired data segment (DS) before reading or writing. If a read, write, format or "Q" (QWAVER) test is to be done, the disk type must be selected first. The normal disk type used in the MPC is type 5 (5" 8Ø track). For each use of the "Y" command, the monitor will ask for drive number, which is defined as follows:

Ø = Left Hand Drive in MPC Cabinet
1 = Right Hand Drive in MPC Cabinet (If Present)
2 = Primary External Drive
3 = Secondary External Drive

It will then ask for the functions to be performed, which are:

Ø = Read a Sector or Range of Sectors into Memory
1 = Write a Sector or Range of Sectors into Memory
2 = Select the Disk Type
3 = Format a Floppy Disk

If a read or write is selected, it will then ask for the starting sector and track address. Enter this as a four digit hex value (SSTT).

If an error occurs, a message similar to the following will be printed:

ERROR Ø4 4Ø Ø4 ØØ ØØ ØØ 66 Ø2

Where the first byte is the diskette I/O (INT 13H) error status returned in AL and the remaining seven bytes are the results string returned by the 8272 floppy disk controller chip. See the definition of INT 13H error codes in Appendix D, and refer to the INTEL 8272 specifications for the interpretation of the results string.

APPENDIX F

PERIPHERALS AND SWITCH SETTINGS

APPENDIX F: PERIPHERALS AND SWITCH SETTINGS

F.1 Using a Dumb Terminal as a Console

The MPC can be operated without a video (color graphics or monochrome) board, monitor and keyboard serving as the console. Instead, a "dumb" CRT terminal can be attached to the rear panel CONSOLE port. With the configuration switches adjusted for this case (see installation, Section 2), the terminal will act as the system console.

The dumb terminal is suitable for the teletype mode of the video functions. It can duplicate the following video functions:

1. 8ØX25 alphanumeric mode
2. Display upper case ASCII characters (Codes 2ØH-5FH)
3. If terminal permits, display lower case ASCII (Codes 6ØH-7EH)
4. Control codes supported by teletype video mode:
   Carriage return
   Line feed
   Backspace (non-destructive)
   Bell

The following video functions are not provided by the dumb terminal:

1. 4ØX25 alphanumeric mode
2. Graphics modes
3. Color modes
4. Direct cursor positioning or reading
5. Light pen
6. Multiple display pages
7. Clear screen
8. Scroll or clear window
9. ASCII character set extensions

Depending on the user program, the dumb terminal may be used to its full capability. The console port will support baud rates from 11Ø to 19,2ØØ which permits faster screen writing than the standard video system. All ASCII codes (Ø-7FH) passed to the video teletype function in the ROM BIOS will be passed directly to the terminal, so escape sequences, clear screen, cursor positioning, color and graphics may be done as defined by the specific terminal used.

The dumb terminal console feature of the MPC also permits the use of telephone links between the computer and a remote console.

System reset, (comparable to [control-alternate-delete] on the standard keyboard) is accomplished by the [BREAK] key of the dumb terminal.

System interrupt (comparable to [control - alternate - insert] on the standard keyboard) is accomplished by typing [ESC] then [BREAK]. This will activate the ROM Monitor (see Appendix E).

The dumb terminal should be configured for the following ASCII word format:

7 Data Bits

No Parity Bit (Parity Inihibited)
2 Stop Bits

Baud Rate: 19,2ØØ 96ØØ 48ØØ 24ØØ 12ØØ 6ØØ 3ØØ 15Ø 11Ø

The MPC chooses 19,2ØØ baud automatically unless a "." (period) is pressed on the terminal keyboard which adjusts the computer to the baud rate selected on the terminal. This key must be pressed within 5 seconds after the second "beep" tone following system reset.

F.2 Printer Interfacing

The MPC interfaces with both standard RS232-C serial devices and Centronics parallel devices. Description of pinouts for the console and serial ports and the parallel port are shown in Appendix A, Tables A-4 and A-5.

In CP/M-86, the parallel device is assigned as the list device when CP/M is booted. The program REDIRECT allows you to assign input and/or output to a serial port. The program SPEED allows you to configure the baud rate, parity, stop bits, and word length of the serial ports. Descriptions of these programs can be found in Section 4.3.8 and 4.3.9.

In MS-DOS, the function Install MS-DOS sets the parameters of the serial devices. Description of this program can be found in Section 3.1.7.

The manual for a printer or other devices must be consulted before connecting to the MPC to insure the pin out of each serial match-up. In the case of serial devices, information concerning the number of start bits, stop bits, word length, and baud rates should be ascertained from the printer manual.

You should obtain the proper cables to connect serial devices or Centronics-type parallel devices from your CDP dealer or verify carefully correct match-up of pins.

On the MPC Serial Input/Output ports, signal and handshake lines are reversed. Data is output on Pin 2 and is inputed on Pin 3 (refer to Reversing Cable Configuration, Table A-6).

To nterface to a serial device (printer, MODEM, etc.):

1. Tie the ground of the device to Pins 1 and 7 of the MPC.

2. Tie the input data line of the device to output Pin 2 of the MPC.

3. Tie the output data line of the device to input Pin 3 of the MPC.

4. Tie the input handshake lines of the device to the output handshake Pins 4 and/or 2Ø of the MPC.

5. Tie the output handshake lines of the device to the input handshake Pins 5, 6, 8, 22 of the MPC.

Pins 5 and 6 are the input handshake lines most often used by software on the MPC.

If your printer or other serial device does not use two sets of handshake lines, the device can be interfaced by tying Pins 4 and 5 together and/or Pin 6 and 2Ø together on the plug which is connected to the MPC.

F.3 Configuration Switch Settings

The MPC contains 2 banks of switches on the main circuit board which specify the configuration of the system. With the cover removed, they are located just to the left of the disk drives and toward the front of the cabinet. Switch bank #1 is the rearmost bank. Each bank has 8 switches which are defined as follows:

Table F-1. Memory Switch Settings

Switch Bank #1

	Total System Memory						
	128K (STD)	192K	256K	320K	384K	448K	512K or More
1	ON	ON	ON	ON	ON	ON	ON
2	OFF	ON	OFF	ON	OFF	ON	OFF
3	ON	OFF	OFF	ON	ON	OFF	OFF
4	ON	ON	ON	OFF	OFF	OFF	OFF

5	ALWAYS OFF	These Switches Are Not Used
6	ALWAYS OFF	
7	ALWAYS OFF	
8	ALWAYS OFF	

Switch Bank #2

1	NORMALLY OFF	(Turn On if 8" Floppy Disks are Used)
2	NORMALLY ON	(Turn Off if 8087 Co-Processor is Installed)
3	ALWAYS OFF	
4	ALWAYS OFF	

	Type of Console Used			
	Monochrome Video Board	Color Board w/Color or B & W Display 80 X 25	40 X 25	Dumb Terminal
5	OFF	ON	OFF	ON
6	OFF	OFF	ON	ON

Table F-1. (Continued)

	Number of Floppy Disk Drives Installed			
	1	2	3	4
7	ON	OFF	ON	OFF
8	ON	ON	OFF	OFF

The switches are factory set for the configuration ordered. If the color video board is included, the 8Ø X 25 setting is selected. In hard disk units, the number of floppy drives configured is 1, otherwise, it is 2.

These switches need only be changed if the configuration of the system is changed from the ordered configuration:

1. Add memory boards.
2. Change video board (Not if change in monitor without board change. Color board supports color or B&W monitors.)
3. Add or remove dumb terminal as console.
4. Add expansion floppy disk drives (Not if hard disks).
5. Change to low resolution monitor with color video board.
6. Add 8Ø87 numeric data processor circuit.

WARNING

Shock Hazard - DO NOT remove the cover of this equipment without first turning the unit off and unplugging.

Should it become necessary to remove the cover of the MPC, remove the two phillips head retaining screws at the back of the unit. Facing the back of the unit, remove the middle phillips head screw on the left flowing edge and the one on the right edge. From the front of the unit, slide the cover carefully toward you and lift off, being careful not to drop it or hit it against other equipment. With the cover removed, the basic inside of the MPC will appear similar to that in Figure 1-2. The disk drives and power supply are mounted securely in a removable tray that rides just above the components of the main processor board. The only operator function within the MPC would be to set any switch settings (Table F-1) or add/remove any expansion boards. Any other adjustments/maintenance in this area should be left strictly to a qualified technician.

F.4 Optional Board Additions

WARNING

Make sure the unit is OFF prior to installing or removing any expansion board.

Choose a vacant board slot on the MPC's expansion bus. Remove the back panel slot cover corresponding to the slot to be filled, by removing the phillips head screw on the inside top of the back panel.

Be sure that the board to be added is properly configured for the system (refer to instructions with the board). Also, be sure to configure the MPC's switches (if necessary) for the upgrade (see Section F.3). Install the new board by pressing down firmly so that the "gold fingers" are firmly inserted into the connector on the MPC's main board.

The board should include a metal bracket to replace the one previously removed. Finally, secure the bracket with the screw previously removed, then replace the cover.

Note: The bracket retaining screw is necessary for chassis ground and if not used, could cause intermittent operation of an expansion board.

APPENDIX G

KEYBOARD CODE GENERATION

APPENDIX G. KEYBOARD CODE GENERATION

Table G-1. ASCII Control Character Chart

LEFT DIGIT

RIGHT DIGIT

	Ø	1	2	3	4	5	6	7
Ø	NULL 0 @	DLE 16 P	SPACE 32	Ø 48	@ 64	P 80	` 96	p 112
1	SOH 1 A	DC1 17 Q	! 33	1 49	A 65	Q 81	a 97	q 113
2	STX 2 B	DC2 18 R	" 34	2 50	B 66	R 82	b 98	r 114
3	ETX 3 C	DC3 19 S	# 35	3 51	C 67	S 83	c 99	s 115
4	EOT 4 D	DC4 20 T	$ 36	4 52	D 68	T 84	d 100	t 116
5	ENQ 5 E	NAK 21 U	% 37	5 53	E 69	U 85	e 101	u 117
6	ACK 6 F	SYN 22 V	& 38	6 54	F 70	V 86	f 102	v 118
7	BELL 7 G	ETB 23 W	' 39	7 55	G 71	W 87	g 103	w 119
8	BS 8 H	CAN 24 X	(40	8 56	H 72	X 88	h 104	x 120
9	TAB 9 I	EM 25 Y	) 41	9 57	I 73	Y 89	i 105	y 121
A	LF 10 J	SUB 26 Z	* 42	: 58	J 74	Z 90	j 106	z 122
B	VT 11 K	ESC 27 [	+ 43	; 59	K 75	[91	k 107	{ 123
C	FF 12 L	FS 28 \	, 44	< 60	L 76	\ 92	l 108	| 124
D	CR 13 M	GS 29]	– 45	= 61	M 77	] 93	m 109	} 125
E	SO 14 N	RS 30 ^	. 46	> 62	N 78	^ 94	n 110	~ 126
F	SI 15 O	US 31 _	/ 47	? 63	O 79	_ 95	o 111	• 127

EXAMPLE: A = 41HEX, R = 52HEX

Table G-2. ASCII Code Table
(Used in Conjunction with Table G-1)

Key	Unshift	Shift	Control	Control And Shift
1	8Ø	8Ø	8Ø	8Ø
2	81	81	81	81
3	1B	1B	1B	1B
4	31	21	31	21
5	32	4Ø	32	ØØ
6	33	23	33	23
7	34	24	34	24
8	35	25	35	25
9	36	5E	36	1E
1Ø	37	26	37	26
11	38	2A	38	2A
12	39	28	29	28
13	3Ø	29	3Ø	29
14	2D	5F	1F	1F
15	3D	2B	3D	2B
16		NOT USED		
17	Ø8	Ø8	Ø8	Ø8
18		NUM LOCK		
19	8B	8B	8B	8B
2Ø	82	82	82	82
21	83	83	83	83
22	Ø9	89	Ø9	89
23	71	51	11	11
24	77	57	17	17
25	65	45	Ø5	Ø5
26	72	52	12	12
27	74	54	14	14
28	79	59	19	19
29	75	55	15	15
3Ø	69	49	Ø9	Ø9
31	6F	4F	ØF	ØF
32	7Ø	5Ø	1Ø	1Ø
33	5B	7B	1B	1B
34	5D	7D	1D	1D
35		NOT USED		

Table G-2. (Continued)

Key	Unshift	Shift	Control	Control And Shift
36	B7	37	B7	37
37	11	38	11	38
38	B9	39	B9	39
39	2D	2D	2D	2D
4Ø	84	84	84	84
41	85	85	85	85
42		CTRL		
43	61	41	Ø1	Ø1
44	73	53	13	13
45	64	44	Ø4	Ø4
46	66	46	Ø6	Ø6
47	67	47	Ø7	Ø7
48	68	48	Ø8	Ø8
49	6A	4A	ØA	ØA
5Ø	6B	4B	ØB	ØB
51	6C	4C	ØC	ØC
52	3B	3A	3B	3A
53	27	22	27	22
54	6Ø	7E	6Ø	7E
55	ØD	ØD	ØD	ØD
56	12	34	12	34
57	35	35	35	35
58	13	36	13	36
59		NOT USED		
6Ø	86	86	86	86
61	87	87	87	87
62		SHIFT		
63	5C	7C	1C	1C
64	7A	5A	1A	1A
65	78	58	18	18
66	63	43	Ø3	Ø3
67	76	56	16	16
68	62	42	Ø2	Ø2
69	6E	4E	ØE	ØE
7Ø	6D	4D	ØD	ØD
71	2C	3C	2C	3C
72	2E	3E	2E	3E

Table G-2. (Continued)

Key	Unshift	Shift	Control	Control And Shift
73	2F	3F	2F	3F
74		SHIFT		
75	2A	AA	2A	AA
76	B1	31	B1	31
77	14	2	14	32
78	B3	33	B3	33
79	2B	2B	2B	2B
80	88	88	88	88
81	8A	8A	8A	8A
82		ALT		
83	20	20	20	20
84		CAPS LOCK		
85	B0	30	B0	30
86	7F	2E	7F	2E

Table G-3. Keyboard Scan Codes

Key Position	Scan Code in Hex	Key Position	Scan Code in Hex
1	Ø1	43	2B
2	Ø2	44	2C
3	Ø3	45	2D
4	Ø4	46	2E
5	Ø5	47	2F
6	Ø6	48	3Ø
7	Ø7	49	31
8	Ø8	5Ø	32
9	Ø9	51	33
1Ø	ØA	52	34
11	ØB	53	35
12	ØC	54	36
13	ØD	55	37
14	ØE	56	38
15	ØF	57	39
16	1Ø	58	3A
17	11	59	3B
18	12	6Ø	3C
19	13	61	3D
2Ø	14	62	3E
21	15	63	3F
22	16	64	4Ø
23	17	65	41
24	18	66	42
25	19	67	43
26	1A	68	44
27	1B	69	45
28	1C	7Ø	46
29	1D	71	47
3Ø	1E	72	48
31	1F	73	49
32	2Ø	74	4A
33	21	75	4B
34	22	76	4C
35	23	77	4D
36	24	78	4E
37	25	79	4F

Table G-3. (Continued)

38	26	8Ø	5Ø
39	27	81	51
4Ø	28	82	52
41	29	83	53
42	2A		

APPENDIX H

GUIDE TO "PERFECT SOFTWARE"

APPENDIX H. GUIDE TO "PERFECT SOFTWARE"

This appendix provides the MPC operator with a brief product description and start-up procedure for the following:

PERFECT WRITER PERFECT-CALC PERFECT-FILER

Product Description: **PERFECT-WRITER**

PERFECT-WRITER is an advanced design software system that represents the latest available technology in word processing.

STANDARD FEATURES:

* Document Creation and Editing
* Help Menus
* Simplified CONTROL Commands
* Data Storage and Security Routines
* Adaptable PRINTING Options

ADVANCED FEATURES:

* Virtual Memory Architecture
* Multiple File Buffers
* Multiple File Display
* Absolute ASCII File Transportability

ADVANCED DOCUMENT DESIGN FEATURES:

* Table of Contents
* Index
* Footnotes
* In-Text Referencing
* Form Letter Design

Start-Up Procedure: **PERFECT WRITER**

1) Insert MD-DOS system disk into drive A.
2) Insert blank disk into drive B.
3) Turn power switch on.

MS-DOS will boot automatically after a short delay

4) Enter "FORMAT B: /S" [cr]

This formats a new disk and transfers the MS-DOS to first 2 tracks

5) Replace disk in drive A with PERFECT-WRITER disk.
6) Enter "COPY *.* B:"

This copies all PERFECT-WRITER files to the newly created disk in drive B

7) Replace disk in drive A with disk in drive B.
8) Reboot MPC
9) Enter "MENU" [cr]

Follow menu options
For details, see PERFECT-WRITER manual

10) To quit, enter "CTRL-X" [cr] followed by "CTRL-C" [cr]

Product Description: **PERFECT-CALC**

PERFECT-CALC is an electronic financial spreadsheet providing convenient data entry and computation with minimal user knowledge of financial and/or scientific computing algorithms.

STANDARD FEATURES:

* Simplicity of use
* Help Menus
* Versatile command structure
* Multiple spreadsheet buffers

* Virtual memory
* Multiple spreadsheet display
* User extendable functions library
* Regional recalculate
* Spreadsheet analysis programs
* Locking formulas, replication, variable formats
* Spreadsheeting with PERFECT-WRITER
* Absolute ASCII file transportability

Start-Up Procedure: **PERFECT-CALC**

1) Complete steps 1-8 for PERFECT-WRITER start-up substituting PERFECT-CALC for PERFECT-WRITER.

2) Enter "PC" [cr}

PERFECT-CALC will initiate by displaying a blank spreadsheet with the cursor cell located in the upper left corner

For usage details see PERFECT-CALC manual

3) To quit, enter "CTRL-X" followed by "CTRL-C"

Product Description: **PERFECT-FILER**

PERFECT-FILER is a file-oriented data base management system.

STANDARD FEATURES:

* Built-in mailing list forms
* Help menus
* Sort routines (up to 5 consecutive sorts)
* Subset definitions (20 per data base)
* List/report format procedures
* Command structure integrated with PERFECT-WRITER to allow for data base editing
* Absolute ASCII file transportability

Start-Up Procedure: **PERFECT-FILER**

1) Complete steps 1-8 for PERFECT-WRITER start-up substituting PERFECT-FILER for PERFECT-WRITER.

2) To start PERFECT-FILER, enter "FILER".

PERFECT-FILER will initiate asking for date and then displaying a menu of action-choices
For usage details, see PERFECT-FILER manual

3) To quit, enter "CTRL-X" followed by "CTRL-C".

INDEX

INDEX

W

Z

COLUMBIA DATA PRODUCTS 8990 Route 108 Columbia MD 21045 (301) 992-3400