MULTIUSER CONCURRENT DOS USER'S MANUAL

Revision 1.70

May 1985

Gifford Computer Systems

P.O. Box 1917
San Leandro, CA 94577

(415) 895-0798
TELEX: 704521

Preface

The Multiuser Concurrent DOS operating system will streamline your workload and increase your productivity. Among its most valuable and innovative features are simultaneous access to 8- and 16-bit programs, time accounting for users or projects, Virtual Terminals, an electronic mail facility, a multiuser appointment calendar, and true networking capability with DR Net networking software. (If your computer is linked to a local area network, you should also read the DR NET USER'S MANUAL). As you begin to use the system you will appreciate the many additional features engineered into MC-DOS by Gifford Computer Systems.

The information in this manual is intended for both the novice user and the seasoned computer veteran. We include information to familiarize you with the system and to help you manage the system effectively. If this is your first experience with a computer, we recommend that you read BITS, BYTES AND BUZZWORDS, by Mark Garetz, 1983, CompuPro.

We urge you to practice patience and pay attention to details while you become familiar with Gifford's MC-DOS and the CompuPro computer. You will find that the time invested in learning how to use the system will pay off many times over.

Disclaimer

Gifford Computer Systems believes that the information presented in this manual is accurate and reliable, but takes no responsibility for any consequences arising out of the use of this manual. Gifford Computer Systems also reserves the right to revise the contents of this publication without obligation to notify any person of such revisions.

Questions or comments regarding this manual should be directed to Gifford Computer Systems.

Trademarks

ARCnet is a trademark of Datapoint Corporation. CP/M and CP/M-86 are registered trademarks of Digital Research Inc. Concurrent CP/M, Concurrent DOS, MP/M II, MP/M-86, and DR Net are trademarks of Digital Research Inc. dBASE II is a trademark of Ashton-Tate. MDRIVE/H is a trademark of CompuPro. MP/M 8-16 is a joint trademark of CompuPro and Digital Research Inc. CompuPro is a registered trademark of CompuPro Corporation. MS DOS is a trademark of MicroSoft. Multiuser Concurrent DOS is a joint trademark of Gifford Computer Systems and Digital Research Inc. PC DOS is a trademark of IBM. IBM is a trademark of International Business Machines Corporation. SuperCalc is a trademark of SORCIM. Virtual Terminals is a trademark of Gifford Computer Systems, a subsidiary of Zitel Corporation. WordStar is a registered trademark of the MicroPro International Corporation.

Copyright Notice

Printed in the United States of America.

Introduction

If you are about to set up a new system which you purchased from Gifford Computer Systems, it is essential that you read **Chapter 1, Getting the System Up and Running,** before you even unpack the boxes. **Chapter 1** tells how to select a location, how to unpack and hook up the computer, and how to turn the power ON. If you are a new user of an established system, you should read the section **Basics of Using the System** in **Chapter 1.**

Chapter 2 provides reference information on **Using MC-DOS.** The commands available on the MC-DOS operating system are described in alphabetical order, along with a **Summary of MC-DOS Commands and Files** and introductory comments about operating systems. **Chapter 2** also describes security features of MC-DOS.

Chapter 3 provides information on **Managing the System.** All systems should have at least one person designated as system manager, whose task should be to insure that the system is used efficiently. **Chapter 3** provides information and examples for the system manager, explaining such things as setting up the password file, customizing the system, backing up files, selecting and installing system security features, and file maintenance.

Chapter 4, Troubleshooting the System, is intended to get you out of trouble in the event that problems arise in using the computer. If you have problems with your computer, please read **Chapter 4** before you call Gifford Computer Systems. **Chapter 4** will give you step-by-step instructions on how to locate a problem, and will tell you what information you will need to have available should you need to call for assistance.

We have included a "Bug Report" at the end of **Chapter 4** to give you the opportunity to inform us of any problems you discover and to suggest improvements for the system or for the manual itself. Make photocopies of the "Bug Report" before filling it out. After you have filled out the "Bug Report," send it to our Customer Support Department for review. We have also supplied several blank pages in **Appendix P** for you to write down notes or questions while you are reading the manual. Use the **Index** if you cannot locate a topic in the **Table of Contents** that follow.

User's Guide to This Manual

Gifford Computer Systems has written this user's manual to conform to certain stylistic conventions for your reading convenience. They are as follows:

1) System prompts and any messages that appear on your screen are in **boldfaced** lettering, e.g. **0A>**.

2) The status of any toggle switch is in UPPER CASE lettering, e.g. "The printer is OFF."

3) System commands are in UPPER CASE lettering, e.g. "Type SDIR *.DOC."

4) Single keystroke commands are encased in parentheses, e.g. "Press (Y)."

5) File names and file types appear in UPPER CASE lettering, e.g. YOURFILE.DOC.

6) References to any other sections of this manual appear in **boldfaced** lettering, e.g. "Refer to **Chapter 3.**"

7) When a caret (^) appears before a character, you should press down the Control key while typing the character. Thus, ^C means to press the CTRL and C keys simultaneously.

8) References to any other manuals appear in upper case, underlined lettering, e.g. "Refer to the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE."

9) Carriage returns are assumed unless indicated otherwise. The word "RETURN" always means to press the RETURN key, rather than typing the six characters.

10) Introduction of any new term will appear in **boldface** lettering, e.g. "**Modems** enable you to communicate over phone lines."

TABLE OF CONTENTS

CHAPTER 1: GETTING THE SYSTEM UP AND RUNNING

Chapter 1 provides essential information on setting up and using your CompuPro computer and the MC-DOS operating system. We urge you to be patient and follow the instructions carefully, whether you are a novice or a seasoned computer user.

CONTENTS

CHAPTER 1: GETTING THE SYSTEM UP AND RUNNING

1.1 The Components of a CompuPro Computer

The CompuPro computer is modular in design. The modular design simplifies future expansion, transportation, and maintenance.

The **terminal** is the actual workstation of the computer. It combines a keyboard with a video screen. Terminals can be located in remote work areas and are connected to the computer by **cables.** On a multiuser system, a number of terminals can be connected to the computer. One terminal is designated the **as system console.** The system console can perform the same tasks as the other terminals, and is the terminal used for starting up, shutting down, and operating the system in single-user mode.

The **computer enclosure** is a metal box with a red power switch and a button labelled RESET on the front. The back of the enclosure has a number of sockets which accept the cables from various components. Inside this box are the actual "thinking" parts of the computer. These parts are contained on modular circuit boards. The boards referred to as **RAM** boards hold the working memory. The **CPU board** holds the central processing unit. Additional boards include disk controllers and serial interface controllers.

The **floppy disk drives** are housed in a metal enclosure similar in size to the main enclosure. Doors on the front of the drive open to accept flexible magnetic disks (**floppy diskettes**) which store information. Your system may come with either one or two floppy disk drives. If you have only one floppy disk drive, you will have to follow special instructions to use some of the commands in the MC-DOS operating system. These instructions are documented throughout this manual.

The **hard disk drive** is a mass storage device capable of storing many times the information available on a floppy diskette. Most multiuser computers have at least one hard disk.

Printers are the devices which will produce permanent "hard copy" records of your data. If the printer is to be placed in an office environment, you should consider placing it in a sound deadening enclosure.

Additional components may include **modems** for communicating over phone lines, **plotters** for drawing graphs, charts and diagrams, **magnetic tape drives** for file storage, or any number of special devices suited to your specific needs.

Systems that are linked to other computers in a DR Net **local area network** must have a **network controller board** in the computer enclosure and a **coaxial cable** connecting the computer to the network. If more than two computers are in the network, (or if the distance between networked computers is greater than 2,000 feet) then coaxial cables are connected to **active repeaters.** If between two and four computers are in the network, and if the distance between networked computers is less than 200 feet then coaxial cables are connected to **passive repeaters.**

1.2 Selecting a Location

For efficient and secure operations you should locate the computer enclosure, the floppy disk drives, the system console, and the hard disk drive in a room that can be locked. Keep reference books about the computer close to the system console.

The computer enclosure and the floppy disk drives require a surface area of 20x30 inches. This is ample space for the floppy disk drives, and the main enclosure can sit on top of the disk drives. A space at table height will allow easy access to the floppy disk drives. These dimensions are deeper than the enclosures and allow for air circulation at the back panels. It is important not to obstruct the air circulation at the back of the system where the fans are or at the left side of the system where the air vents are located. Excessive heat will impair system performance and may damage the system. For the same reason, keep the computer away from heaters, and keep the room temperature at a comfortable level.

A terminal, preferably the system console, should be located within easy reach of the floppy disk drives. This will allow floppy disks to be changed easily by the system manager operating the system console. You should have this manual, the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE, and any additional reference materials close at hand.

The hard disk drive, if you have one, should be located in a well ventilated spot. Locate it where it will not be bumped or kicked.

Where you locate your printer is primarily a function of who will use it and the noise it generates. It is handy, but not essential, to have a printer close to the system console.

1.3 Unpacking the Parts

Begin unpacking by locating the cover letter and the invoice or packing list. The cover letter gives your system number, which you will use should you need to call Gifford Computer Systems. On the back of the main enclusure, there is a pressure sensitive label on which you will find your system number. Complete a copy of the system map provided in **Appendix M.**

Verify that all the items on the invoice or packing list are in the shipment, and file the cover letter and the invoice for future reference.

Open all boxes and carefully remove each unit from the packing material. Place the units on a flat surface and inspect the cabinets for signs of shipping damage. Put aside all cables, cords, and diskettes. Save the packing materials, in the event that you have to return any of the units.

The Floppy Disk Drives

To prevent shipping damage, Gifford Computer Systems ships the floppy disk drives with cardboard shipping inserts in the drive slots. It is extremely important to **remove the cardboard inserts** before using the drives. To open the drives, press the button below the drive door. Save the inserts with the packing materials in case you need to ship the drives again. You must put the inserts back in the drives even if your are transporting the drives a few miles by automobile. Failure to do so risks damage to the drives.

The Hard Disk Drives

Hard disk drive heads are locked in place for shipping. If you ordered hard disk drives, follow the directions below for the type of drive that you ordered.

5.25-inch. Most drives supplied with Gifford computers have an automatic locking mechanism. If you have one of the few drives without automatic locking, a piece of tape will show across the front of the drive. Remove the cover of the drive, carefully remove the tape, and replace the cover.

8-inch. Remove the screws from the underside of the cabinet, remove the front cover, and find the LOCKED and UNLOCKED label on the drive. Unlock the drive by switching the lever underneath this label to the UNLOCKED position. This lever is held in place in two notched areas. Hold the lever down, pulling it out of the first notched area marked LOCKED, then slide it across and up to the second notched area marked UNLOCKED. Replace the cover.

14-inch. If you have a Control Data Corporation Cartridge Module Drive refer to the CONTROL DATA CARTRIDGE MODULE DRIVE HARDWARE AND MAINTENANCE MANUAL. The section on installation and checkout will describe the necessary procedures.

1.4 Installing the System Hardware

Your computer was thoroughly tested at the Gifford Computer Systems plant before it was shipped. You should have no problems getting it set up and running, but you should check that circuit boards in the main enclosure did not vibrate loose during shipping. You will need a Phillips head screwdriver.

Checking the Internal and External Connections

Before you check for loose boards, be sure that the 110 volt power cords are not plugged in. Never work on the inside of an enclosure when the enclosure is plugged in.

To check for loose boards, remove the screws on the sides of the enclosure and lift off the top. You will notice a number of 5-inch by 10-inch circuit boards which plug into sockets in the bottom of the case. This array of sockets is called the **motherboard.** Remove the piece of foam packing from between the top cover and the circuit boards.

Lift the boards out of the sockets by using the white plastic ears at the edge of each board as levers. Reseat the boards by pushing firmly down on the edge of the boards (**do not push down on the plastic ears**). Be sure that any cable connectors on the top of the boards are attached securely.

Connecting the Cables

The connectors on the back panel are labeled to indicate which cables to attach.

Connect the terminal and printer cables to the sockets on the back panel of the computer enclosure. Begin with the system console, then terminal 1, terminal 2, and so on, until all terminals and printers are connected. Tag both ends of each external cable and record the cable name in **Appendix M.**

If special cables are needed for your devices, the ends of these cables should be marked.

On a multiuser system, do not attach terminal cables to the computer enclosure unless they are also attached to working terminals. Terminal cables not connected to a terminal, or connected to terminals that are not turned ON, will degrade the performance of your system, sometimes significantly. Refer to Section 4.0 of the manual entitled CUSTOMIZING YOUR MULTIUSER CONCURRENT DOS SYSTEM to determine the proper cable connections for your particular system configuration.

If you make your own RS-232 cables, or if you use cables supplied by other manufacturers, see **Appendix L** for important additional information on RS-232 cables and pin assignments.

Connecting the Floppy Disk Drives

The left (50-pin) connector (rear view) of the Disk 1 or Disk 1A controller board is connected to the back panel of the computer enclosure with a 50-pin cable inside the enclosure. Be sure that the board and cable connections are secure.

Connect the back panel of the enclosure to the floppy disk drives with the 50-lead flat ribbon cable marked "COMPUTER" and "DRIVES." Both ends of the cable are marked with the word "UP" to indicate the proper orientation of the cable when it is installed. Take care not to bend the pins on the connectors, and be sure that the cable is plugged in securely. **Check that you have removed the cardboard shipping inserts from the drives.**

To check that the switches on your boards are set properly, refer to **Appendix C** or the one-page flyer labelled *****IMPORTANT***** that was provided with your system.

Connecting the Hard Disk Drives

Two cable connections are made to the hard disk drives. The flat ribbon cables come out the back panel of the system through the slots labeled "50-pin hard disk cable" and "20-pin hard disk cable." If you have 8- or 14-inch hard disk drives, the cables are 20-lead and 50-lead; if you have 5.25-inch hard disk drives, the cables are 20-lead and 34-lead.

Connect the two cables to the back of the hard disk drive. The cables are marked with the word "UP" to indicate the proper orientation of the cable when it is installed in the connectors. Take care not to bend the pins, and be sure that the cable is plugged in securely.

Do not format your hard disk. It comes from Gifford Computer Systems completely tested. All of your system files are already installed on user area 0 of drive A:. If you ordered other software, and your terminal configuration was known at the time of your order, your other software is also installed on the hard disk.

Final Connections

Replace the cover on the computer enclosure (Enclosure 2), making sure that the ventilation slots on the cover face the left of the machine (front view). Loosely fasten all screws on the side of the cover before tightening any of them. If the slots do not line up with the slots on the computer enclosure, the machine will have inadequate ventilation.

Plug in all of the power cords. The main enclosure has three utility outlets on the back panel for plugging in disk drives, terminals, or printers. These AC utility outlets are switched on by the main power button, and the sum total of current drawn on all three outlets must not exceed 5 amps.

Gifford Computer Systems recommends that you plug the floppy and hard disk drives into the utility outlets on the enclosure. Plug printers directly into the wall sockets, since the enclosure outlets are rated lower than some printers.

1.5 Handling Diskettes

The floppy diskette is a flat piece of round plastic with a magnetic coating. A protective jacket surrounds the magnetic disk. The jacket has slots for the heads of the disk drives to read and write. Handle the diskette by the jacket and avoid touching its inner magnetic disk. To protect diskettes from damage, avoid exposure to direct sunlight, magnetic fields, heat, and liquids. Do not use a pencil or ballpoint pen on the adhesive label after it has been affixed to a diskette.

Along one side of the diskette is a label. The opposite edge has two notches to position the diskette in the disk drive. A third and larger notch is found on most diskettes. This is a **write protect** notch. When this notch is exposed, the computer cannot write information on the diskette. Adhesive "write-enable tabs" can be used to cover this notch, allowing information to be written on or erased from the diskette.

If you have experience with 5.25-inch diskettes, please note that an exposed notch on 8-inch diskettes indicates write protected. On 5.25-inch diskettes, a covered notch indicates a protected diskette.

Floppy diskettes are available as either double-sided or single-sided. You can tell the difference by looking at the position of a small hole in the jacket near the large center hole. Position a diskette so the seams face away from you and the slot points down. If the hole appears at one 1:00, the diskette is double-sided. If the hole appears midway between 12:00 and 1:00, the diskette is single-sided. Your computer will accept the following diskettes:

DSDD Double-sided, double-density, SSDD Single-sided, double-density
DSSD Double-sided, single-density, SSSD Single-sided, single-density

Gifford Computer Systems recommends that you use double-sided, double-density diskettes. You can get twice as much information on a double-sided diskette for only a slightly higher cost, and you will spend less time changing diskettes in the drives. Do not use "hard sectored" diskettes (i.e., diskettes with two or more index holes).

1.6 Starting Up the System for the First Time

Once your system is running, you must back up all master diskettes. Once a master diskette has been copied, it should be stored in a safe place and used only to create additional working diskettes, should the one in use fail. Backing up the system diskette involves several steps which are the responsibility of the system manager.

Starting up the system for the first time involves:

- Turning ON the power
- Loading the operating system into the computer (booting the system)
- Formatting new diskettes
- Making backup copies of the operating system
- Rebooting with your working system diskette

Turning ON the Power

Check that all terminals and printers are connected to the computer. Turn ON the power to the terminals and printer(s). **All terminals and printers must be turned ON before the system is turned ON.**

Press the eject buttons under the drive door(s) to open the drives and eject any diskettes or inserts. **The drives should be empty when you turn ON the power to prevent power surges from affecting data on the diskette.**

CHAPTER 1: GETTING THE SYSTEM UP AND RUNNING

The computer should be plugged into a 3-prong 110 volt power socket. Hold in the RESET button on the front of the system enclosure, and turn ON the power by pressing the top portion of the red power switch on the front panel. Release the RESET button.

It is important to hold in the RESET button on the front of the main enclosure when you turn the power ON or OFF. This prevents incorrect data from being written to floppy and hard disks when the system is turned ON or OFF.

Now the system is ON and you should hear the hum of the various fans. The RESET button should be lit and the red light on the left-hand disk drive should be blinking.

If you do not use the utility outlets on the back of the main enclosure to power the floppy and hard disks, you must turn ON the drives separately. Turn ON the power to the floppy disk drives by flipping the toggle switch on the rear panel. Turn ON power to the hard disk drives with the power switch on the front panel.

If the fans are not turning, or the RESET light is not ON, or the left floppy drive light is not blinking, hold in the RESET button and turn OFF power to all components. Recheck the connections between the components as well as the power plug connections before turning the power ON again. Refer to Chapter 4 if you need further assistance.

CHAPTER 1: GETTING THE SYSTEM UP AND RUNNING

Backing Up the Original System Diskettes

It is essential to make at least one copy of the original MC-DOS master diskette. This process of copying a diskette is called backing up. **You should back up all the original diskettes you receive from software suppliers, as well as any important records you generate on the computer.** Most distribution diskettes are single-sided, 128 bytes per sector, and single-density. The MC-DOS system diskettes are double-sided, double-density diskettes.

A diskette which has the MC-DOS operating system on it is called a **system diskette.** To back up the system diskette you will need to load the operating system, format a blank diskette, and copy the information from the system diskette onto the formatted diskette using the CPY program. If your system only has one floppy drive, you cannot use the CPY program. MC-DOS comes supplied as three diskettes: the Multiuser Concurrent DOS System diskette, the Multiuser Concurrent DOS XIOS diskette, and the Multiuser Concurrent DOS Utilities diskette. You will also be supplied with the DR Net Networking diskette if you ordered DR Net. You will need to have at least three blank double-sided, double-density diskettes to carry out the procedure outlined. You should have a couple of boxes of blank diskettes handy, since you will soon need to back up files that you create on the computer.

Loading the Operating System

When the power is turned ON, the fans and lights go ON, but the RAM (the computer's memory) is empty. The computer is incapable of doing any real work until it receives instructions on how to operate. Since the computer "loses" its RAM memory every time the power is turned OFF, it must instruct itself on how to get instructions. This process is like pulling yourself up by your bootstraps. Computer users call this process "bootstrapping" the system or "booting" the system. The computer has a small amount of read-only "boot" memory which gives instructions on how to begin loading the operating system whenever power is first turned ON, or the RESET button is pushed. Once the information on the system diskette has been loaded, the computer can proceed with any of its appointed tasks.

If the left drive light is blinking ON and OFF, you are ready to load the system. Press the RESET button to clear the computer.

Locate the Gifford Computer System diskette with the MC-DOS operating system on it. Insert the diskette into the left floppy disk drive if you have two drives. The label should be on the top, and the side with the seams of the jacket showing should face down. The slot in the jacket should face the drive as you insert it. Push the diskette straight into the drive until it clicks into position. Now close the drive door.

Loading the operating system takes a few moments. The system startup messages displayed below appear on the system console stating memory size and copyright and trademark notices.

```
MC-DOS Loader Rev. 2.0A.
Loading CCPM.SYS.  Please stand by . . .


Total Memory= 1 Megabyte

    Operating system= 239K
    Available user  = 784K
MC-DOS loader 2.0A

CCPMINIT                version 1.0   (07/30/84)
Network I/O System      version 1.1   (07/30/84)
SETNET                  version 1.2   (07/30/84)

MC-DOS copyright  1984, Gifford Computer Systems
Concurrent CP/M copyright  1984, Digital Research Inc.

0A>
```

The memory message will vary with the amount of RAM in your system. The CCPMINIT message does not appear on systems that do not implement Virtual Terminals. The Network I/O System and SETNET messages do not appear on systems not linked to DR Net local area networks.

The last item on the screen, **0A>**, is the **MC-DOS prompt.** You are still in single-user mode. To enter multiuser mode, type LOGOUT. The screen will now display:

Name:

This is called the **login prompt** or the **name prompt.**

Here you must enter a valid account name, and optionally a password. Your Gifford computer is shipped with only one valid account name, "system." Initially, this account does not require a password, although the system manager should use the NEWUSER program to place a password on this account during the customization procedure.

Type SYSTEM and the screen will display the MC-DOS prompt: **0A>.** The MC-DOS prompt provides essential information on user areas and drive designations.

User Areas and Drive Designations

```
0 A >
\ \
 \ Current drive
  \
   Current user area
```

The MC-DOS prompt (0A>) tells you that you are currently logged into user area 0 of drive A:. Each drive has 16 user areas (0 through 15). If the prompt is 14E>, you are in user area 14 of drive E:.

The user areas are not physically distinct portions of a drive, but are characteristics of the programs and data stored on a drive. All of the storage capacity of a drive could be occupied by files in a single user area, but it is convenient to keep in different user areas the files used in different tasks or by different people. Each unique combination of user area and logical drive is called a **directory area.**

The number of drives on your CompuPro computer will depend on whether or not you have a hard disk on your system. The following table will tell you the drive designations of your system. If you have a system with only one floppy drive, ignore any references to the right-hand floppy drive, and assume your floppy drive to be the left-hand floppy drive.

Type of System	Logical Drive Designations		
	Hard disk Logical Drives	Left-hand Floppy Drive	Right-hand Floppy Drive
Floppy-only	none	A:	B:
5Mb hard	A:	B:	C:
11Mb hard disk	A:B:	C:	D:
16Mb hard disk	A:B:	C:	D:
21Mb hard disk	A:B:C:	D:	E:
37Mb hard disk	A:B:C:D:E:	F:	G:
40Mb hard disk	A:B:C:D:E:	F:	G:
56Mb hard disk	A:B:C:D:E:	F:	G:
84Mb hard disk	A:B:C:D:E:F:	G:	H:

Systems with two or more hard disks add the same number of logical drives as shown in the table above for each hard disk, with the two floppy drives moved through the alphabet (towards "P"). The maximum number of logical drives is sixteen (P:).

User areas and drive designations will be important when you call programs and assign tasks or personnel to specific work areas on the computer. Two user areas deserve special attention. The computer expects to find certain system programs and files in user area 0 of drive A:. You should keep frequently used programs in user area 0 of drive A:. The second special purpose area is user area 15, drive A: of your system, or user area 15, drive A: of the system node if your system is a part of a DR Net network. The computer keeps appointment and mail files here. We will discuss user areas and drive designations again after we have discussed the basics of using the computer.

Formatting a Floppy Diskette with FORMAT

Formatting a diskette is an essential operation in preparing the diskette for use. Formatting puts information on the diskette to tell the computer how the diskette is arranged. Diskettes are organized into tracks and sectors. Each 8-inch diskette has 77 tracks. You will specify the size of the sectors.

To begin formatting, select a blank diskette and cover the write protect notch with an adhesive write-enable tab.

Your screen will show:

```
0A>
```

Now type:

```
0A>FORMAT
```

After a few moments the screen will say:

```
CompuPro Disk Sub-system FORMAT Version 3.1
For CP/M-86, MP/M 8-16, and MC-DOS - Modified 7-10-84

Specify drive (A: - D:, E: - H:) :
```

The FORMAT program thinks that you have a floppy-only system and that your only drive designations are A: and B:. Note that this is an exception to the drive designation table on the preceding page. Disregard the A: - D: and E: - H: options for the time being. Refer to **page 2.5-FORMAT-1** if you need information on these options.

If you have a system with a single floppy drive, the FORMAT program thinks that this is drive A:. When you are prompted to supply the drive that holds the diskette, type (A).

For this example, insert the blank diskette in the right-hand drive and as a safety precaution, remove any diskette that you don't want to format from the other drive.

Now press (B).

The screen adds your B to its message and then asks you:

Select Disk format mode (0,1,2,3):

0 = 128 2 = 512
1 = 256 3 = 1024 >

In response, you should press (3).

You will make the most efficient use of diskettes if you format them with 1024-byte sectors. The other options are available to you in the event that you will be formatting diskettes for another computer. Commercial software distributors often supply single sided diskettes formatted with 128-byte sectors (option 0).

Because **formatting a diskette destroys information on the diskette,** the system gives you a last chance to reconsider:

Confirm ready for format on disk drive B (y).

If you are sure you are ready, press (Y).

When you press (Y) the screen displays symbols of the 77 tracks of the diskette, numbered from 0 to 76, and indicates when each track is formatted (F) and verified (V):

```
1         2         3         4         5         6         7
01234567890123456789012345678901234567890123456789012345678901234567
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
 VVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVV
```

If you get an E or an S in the row of Vs, something is wrong with a track. Try to format the disk one more time, and if it still displays an E or an S, you should discard the diskette and try again with a fresh diskette. After the verify operation, the following message appears:

```
  FORMAT complete.
Do you want to FORMAT another disk?
     R  -  request new parameters.
     cr -  duplicate last FORMAT.
     N  -  exit back to system.

                                    >
```

You now have a formatted diskette. Press (N) to exit back to the system.

Put the system diskette back in the left drive and leave the formatted diskette in the right drive.

Copying the System Diskette with CPY

The CPY program cannot be used with a single-floppy drive system. The left-hand drive should have the original system diskette and the right-hand drive should have the formatted diskette.

The last line of the screen should show:

```
0A>
```

Now type:

```
0A>CPY
```

and press RETURN.

This command invokes the MC-DOS copy utility program. CPY creates an exact copy of a diskette. Since Gifford Computer Systems supplies its system software on a double sided, double density 1024-byte sector diskette, the CPY program expects to find a double sided, double density 1024-byte sector diskette as the destination diskette. **CPY will not work if the disk formats are different.**

The program prompts:

```
Gifford Computer System Copy Utility V2.0
Do you want to copy:
  SYSTEM tracks only? (type S)
  DATA tracks only?   (type D)
  ALL of the disk?    (type A)
Exit back to System   (type X)
```

Press (A) (with no RETURN after it), since you want to copy the entire diskette. The program prompts:

```
Source drive? (A, B, . . .)
```

Enter the drive designation for your left-hand floppy drive. Refer to **page 1.6-6** if necessary. The program prompts:

```
Destination drive? (A, B, . . .)
```

Enter the drive designation for the right-hand floppy drive. The program prompts:

```
Put source disk on . . .
Put destination disk on . . .
Then type RETURN
```

The two floppy drives should contain the designated diskettes. You should verify that the drives have the proper diskettes. It is extremely important that you have the source diskette in the drive that you designated, since you will erase this diskette if you reverse the two. To prevent inadvertent destruction of the source diskette, remove its write-enable tab.

Press RETURN and the program copies each track, typing the track number on the screen.

When all of the tracks are copied, the program prompts you for the next copy operation. Press (X) to exit from the program. Remove the system diskette from the left-hand drive and put it in a safe place. Remove the copy from the right-hand drive and put a label on it to identify it. This will be your working copy of the MC-DOS operating system. Put the new working system diskette into the left-hand drive and press the red RESET button.

The system should boot with your copy. At this point you can use the FORMAT program to format blank diskettes and the CPY program to copy any additional diskettes which need to be backed up.

Before you go on to **Basics of Using the System**, you should be aware that the procedure presented above is one of several procedures you can use to make copies of programs. An alternative is the MC-DOS PIP utility to copy data tracks, and then running the SYSGEN program to copy system tracks onto the diskette. PIP will not copy the system tracks. PIP is one of the most frequently used programs in the MC-DOS system. You must use PIP if you want to copy software between diskettes with unlike formats or between double and single sided diskettes, or between a floppy diskette and a hard disk.

Copying Diskettes on Single Floppy Drive Systems

You cannot directly copy diskettes to diskettes on systems that support only one floppy disk drive using the CPY program. If you have a hard disk, you can copy all of the files on a diskette to an empty directory area. Then give the DSKRESET command, replace the source diskette with a formatted destination diskette, and PIP (described in section **2.5-PIP**) the files from the hard disk to the destination diskette.

1.7 Basics of Using the System

Login Account Names

You must **log in** to an **account** before you can use the system. An account is an entry that the system manager has made in the PASSWD file, a special file that tells MC-DOS who is allowed to use the system. One user can have many accounts, or one account name can have many users, provided that each user knows the password for that account.

Passwords

The account name may have a password associated with it. To log in to an account, you must be told by the system manager both the account name and its password, if any. Gifford computers are shipped with only one valid account name, "system," which should be reserved for the system manager. Before you log in, your terminal will display the login prompt:

```
Name:
```

To log in, you must enter a valid account name. Suppose that your PASSWD file included an account name called "guest." To log in to the "guest" account, enter:

```
Name:GUEST
```

If the "guest" account requires a password, the screen will display:

```
Password:
```

You must now enter the password that the system manager has assigned to the account. If the account "guest" has the password "secret," you must type:

```
Password:SECRET
```

The word "secret" will not actually appear on the screen. This prevents anyone who might be looking over your shoulder from finding out the login password for that account.

You should now receive the MC-DOS prompt, indicating that you have successfully logged in to the system.

Changing User Areas and Logical Disk Drives

The MC-DOS prompt (0A>) tells you that you are currently logged in to user area 0 of drive A:. Each drive can have up to 16 user areas. If the prompt is 6C>, you are in user area 6 of drive C:.

When you log in, the computer will log you in to a default drive and user area, called your initial default directory area.

To change drives, enter the drive you want to move to followed by a colon. The following example will move you from user area 0, drive A: to user area 6, drive B:.

```
0A>B:
```

The system prompt returns with:

```
0B>
```

To change user areas, type the following command:

```
0B>6:
```

You can also use the USER command (USER 6) to change user areas. If you have been allocated access to that user area by the system manager in the PASSWD file, the system returns with the prompt:

```
6B>
```

Find out from the system manager what user areas you are allowed to access.

Invoking Programs

You have already called on three programs, FORMAT, CPY, and USER, to perform specific tasks for you. These three programs are commonly referred to as MC-DOS utilities MC-DOS commands. You can display a directory of drive A: by entering the following:

```
0A>DIR
```

You will see a list of the files on drive A:. The computer identifies discrete sets of information as **files.** A **file name** has up to eight letters or numbers. It is separated by a period (.) from an optional **file extension,** or **file type,** of up to three letters or numbers. A file can be a data file, a program file, a document file, or any one of a number of different file types. CPY, FORMAT, and DIR have the file type .CMD, which indicates that they are 16-bit program files. Program files with the file type .COM are 8-bit program files.

To run a program, you simply type the file name. On some occasions, you will provide more than just the file name. The

command line that you enter after the 0A> prompt may contain the following:

```
0A>B:FILENAME options
     \    \     \
      1    2     3
```

1. The drive designation indicates the drive from which the file is to be executed. Program files on drive A: which have been given the system (SYS) and read-only (RO) attributes can be accessed from all drives and user areas without specifying the drive designation. The drive designation can also be omitted if the file is on the logical drive shown in the MC-DOS prompt.

2. The FILENAME is the name of the program that you intend to run. The filetype (.CMD or .COM) should not be typed.

3. A number of programs will require options which are specific to the file being called. Others like PIP, REN and SDIR require file names and types. Files which are password protected will require the password here. (A more detailed description of MC-DOS command lines appears on **page 2.5-1**).

Now type:

```
0A>DSKRESET
```

DSKRESET tells the computer that you are changing diskettes. If you change floppy diskettes without giving the DSKRESET command, you will not be able to read or write information on the new diskette. (Users familiar with CP/M should note that the MC-DOS DSKRESET command has the same function as ^C in CP/M. Under MC-DOS, however, ^C may abort a program, and will not execute a disk reset.)

Now type:

```
0A>SHOW
```

The SHOW utility will tell you how much space, in kilobytes, you have on the drives that users have accessed.

Type:

```
0A>HELP MCDOS
```

The HELP program can provide help on a number of MC-DOS features. For instance, type:

```
0A>HELP MCDOS PIP
```

and you will see a display on how to use the PIP command.

The HELP program provides important documentation on programs such as PIP. You can add information to the HELP files or create custom HELP files to provide information specific to your own system, as described in **Chapter 3.**

PIP is one of the most frequently used utilities. You will use PIP to copy the files from one drive or user area to another. PIP will not copy files from user areas to which the system manager has not granted you access in the PASSWD file.

Program files on user area 0, drive A:, which have been assigned system status with the SET command are accessible to all users on all drives. The SET command is discussed in **Chapter 2.**

As an example of copying and deleting files, create a test file called TEST.DOC on drive A:, copy it to drive B:, and then erase it. Type the following commands:

```
0A>MAKE TEST.DOC
0A>PIP B:=A:TEST.DOC[VROW]
0A>DIR B:
```

You will see the file TEST.DOC on the drive B: directory. The PIP command can also be used with "wildcard symbols." An asterisk designates any character, and a string of question (?) marks designates any character, or characters, up to the number of question marks. Type the following to experiment with wildcard symbols:

```
0A>PIP B:=A:*.DOC[VROW]
0A>DIR B:
```

Now drive B: contains copies of all the .DOC files. Before you go on, eliminate any unwanted files from drive B:. Type:

```
0A>ERAQ B:*.DOC
```

The system will list the files with a .DOC file type on user area 0 of drive B:. Type (Y) after each file that you want to delete.

You can refer to **Chapter 2** and also to the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE for more information on MC-DOS commands.

To call on applications programs, you enter the file name for the program you want to use. To use SuperCalc enter:

```
0A>SC
```

This loads SuperCalc. If you want to use WordStar, enter:

```
0A>WS
```

For dBASE II, enter:

```
0A>DBASE
```

Once you are in an applications program such as SuperCalc 86, you no longer have direct access to the operating system. Each applications program has its own procedure for terminating the program and returning to MC-DOS. You must read the program documentation to learn the specific command that each program requires to send you back to the operating system.

SuperCalc, for example, uses:

/QY

while dBASE II uses:

QUIT

and WordStar uses ^KX or X.

After you have invoked a program, you generally cannot talk to the operating system directly until you exit from the program. See **Appendix J** for dBASE II exceptions.

Virtual Terminals

A **multitasking** computer has the ability to perform several jobs at once from each workstation. An important improvement over ordinary computers is available with Gifford's **Virtual Terminals.** This allows a physical terminal that has been specially equipped with extra pages of memory, such as the Freedom 200, the GCS-80, or the TeleVideo 950 or TeleVideo 925, to run several programs at once.

When application programs that are not currently attached to the physical terminal try to output data, this multitasking software automatically redirects their output to RAM or to disk buffers, so-called "Virtual Terminals." When the user returns control of the actual screen to a detached program, all buffered output from that program is then transferred to the physical screen.

CHAPTER 1: GETTING THE SYSTEM UP AND RUNNING

Gifford's Virtual Terminals software allows up to four different programs to share the same physical screen and keyboard. The user can pick which program is attached to the actual screen and keyboard at any given moment, and can switch from program to program simply by touching specially programmed function keys.

This resembles the way that an ordinary television set can select a specific channel to watch from among the various programs available from the cable or antenna. Virtual Terminals software lets you select which program you want to see on screen by which function key that you press.

In giving each program the whole screen, the Virtual Terminals software also differs from the "windowing" approach to multitasking, which splits the screen among different programs. Although split screens are helpful in some situations, most people find it less confusing when they are multitasking to see the same screen display that they would see if only one process were running.

A special advantage of using Virtual Terminals software is that a single-user can have a powerful array of programs and utilities running at all times, waiting for input. Though this can make heavy demands on system RAM, it does not affect system speed as long as the background processes are simply waiting for input.

The number of simultaneous programs that you can run with Virtual Terminals on a single physical terminal is limited by the number of extra pages of memory in your physical terminal. Just as in multiuser operations, the total RAM requirements of programs run on Virtual Terminals cannot exceed the RAM available in the system.

See the VIRTUAL TERMINALS USER'S MANUAL for detailed information on the installation and use of Virtual Terminals software. **Section 3.3, pages 3.3-7** and following outline the installion of Virtual Terminals. A user may need to give a command like KEYS GCS80 to install a physical terminal for use with Virtual Terminals.

1.8 Shutting Down the System

It is important that you follow these procedures each time you turn OFF your system or RESET (re-initialize by pressing the RESET button), especially if you are running your system with a hard disk. These steps ensure that data is written from the hard disk cache. **Failure to follow these instructions may result in loss of valuable data.**

All users (except the operator at the system console) should log out of the system by entering:

```
0A>LOGOUT
```

After all the other users have logged out, the system manager at the system console enters:

```
0A>WHO -A
```

The WHO command responds with a display of which users are on the system, and specifies which Virtual Terminals are currently attached by placing an asterisk next to that Virtual Terminal number. Only a single user, the system manager (at the system console) should be listed. It is important to look and see who is on the system before you shut down or RESET. Users will lose their work unless they save their files before the system goes down.

Now type:

```
0A>DOWN
```

This disables all terminals except the system console. The following message is displayed on the system console:

The system is down.

Release the diskettes from the floppy disk drives. Hold in the red RESET button on the front of the main enclosure, turn OFF power to the computer by pressing the bottom part of the power switch down, and release the RESET button.

Place the diskettes in their paper sleeves and file them where they belong. You can turn OFF any terminals and printers if necessary. It is important to follow the procedure outlined above. These steps ensure that data is written from the cache to the hard disk.

CHAPTER 2: USING MC-DOS

Chapter 2 describes the MC-DOS operating system, its control characters, command lines, command files, and system data files. **Section 2.5** describes system commands and files individually, with examples of usage. The last part of the chapter outlines MC-DOS system security.

CONTENTS

FIXSTEP
FORMAT
GENCCPM
GENCMD
HELP
HELPNDX
HISTORY
INITDIR
KEYS
LOGIN
LOGIN.SUB
LOGOUT
LOGOUT.SUB
LPRS
MAIL
MAKE
MOTD
NAMENODE
NAMSVR.DAT
NET
name.NET
NETMOTD
NETUSERS
NEWUSER
PASSWD
PASSWORD
PIP
PRINT
PRINTER
PWASK.NET
name.REM
REN
RESET
SDIR
SET
SETMEM
SETNET
SHOW
SPACE
SPOOL
SUBMIT
SW

CHAPTER 2: USING MC-DOS

2.1 The Operating System

Like the central nervous system of a living organism, the **operating system** imposes order on the activity of a computer. When the computer starts up after power is turned ON or RESET, the RAM memory is initially empty. The computer does not know how to use the terminals, the memory, the disk drives, the printers, nor how to perform the various tasks that are required for practical work. **Applications programs** such as SuperCalc or WordStar can perform complicated tasks, but they rely on the operating system to give them access to the different parts of your system. Although you probably acquired your computer in order to use applications programs, they depend upon the operating system to be able to enter and store information, to create printed or screen output, and to locate data and program files.

The operating system is a collection of programs and files on the Gifford MC-DOS system diskette. It provides the information needed for the parts of your computer to act coherently as a system. The operating system creates an environment in which applications programs can carry out their intended functions.

2.2 The Multiuser Concurrent DOS Operating System

Multiuser Concurrent DOS is Gifford Computer Systems' proprietary enhancement of Concurrent DOS (3.1) by Digital Research Inc. MC-DOS is a powerful multitasking, multiuser, networkable operating system with many capabilities not available on most business computers. Advanced features of Multiuser Concurrent DOS include:

- o Support for multitasking operations. Each physical terminal can support up to four concurrent "virtual terminals," each performing a different task, and accessible by a keystroke.

- o Support for multiuser operations. Users at different physical terminals can use the same computer simultaneously and share programs, data, and hardware resources.

- o Optional support for DR Net local area networks. A high speed network can link many MC-DOS computers, so that many users can share the data, programs, storage, and printers of the network.

- o Advanced security features. The system manager can monitor usage and restrict file access on an individual basis, tailoring security procedures to match specific requirements.

- o Communication and telecommunications features. Messages and electronic mail can be exchanged among users, reminders can be automated; option to access other computers by modem.

- o Customization features. Utilities are provided so the System Manager can reconfigure and control the system to optimize the productivity of individual users, without use of assembly code.

- o Support for 8-bit and 16-bit operations. Multitasking and multiuser operations by any user can simultaneously intermix programs written for both 8-bit and 16-bit microprocessors.

The last feature is important because most operating systems restrict users to a single processor, or can select one or another, but not use two at once in a true multiuser environment. Using MC-DOS is like being able to speak two languages at the same time. MC-DOS gives you access to thousands of popular 8-bit programs as well as many new 16-bit programs. Gifford's MC-DOS can support languages and programs written for the **CP/M-80, CP/M-86, MP/M II, MP/M-86, MP/M 8-16, Concurrent CP/M-86,** and **Concurrent DOS** operating systems of Digital Research Inc., the largest selection of commercially available software supported by any single operating system.

Besides these unusual features, Gifford's proprietary enhancements of MC-DOS provide a wide range of utilities to increase the ease and flexibility of using your computer for business, engineering, and scientific work. By studying the commands and files described in this chapter you can take full advantage of MC-DOS in automating your work.

Computers that support Gifford's MC-DOS can also utilize another operating system, called **CP/M-86,** a product of Digital Research Inc. The CP/M-86 operating system can only support one user (at the system console), and does not utilize the hard disk. The main use of CP/M-86 is as a diagnostic tool, in the event that you experience problems with your system. If the computer operates properly under CP/M-86, then the problems may lie in the

hard disk subsystem, which CP/M-86 ignores. For assistance in the diagnostic use of CP/M-86 or in other troubleshooting, read **Chapter 4** of this manual and contact your dealer.

Another potential use of CP/M-86 with systems having less than 512K of RAM is to support programs that can benefit from large transient program areas. (A **transient program area** is the part of RAM where an applications program or language is located when it is running.) For example, a system with 320K of RAM supports larger spreadsheets if CP/M-86, rather than MC-DOS were running, because MC-DOS occupies more RAM than CP/M-86, leaving less RAM free for applications programs.

Many 16-bit programs that are compatible with MC-DOS can also run under CP/M-86. If you plan to use CP/M-86 to operate your computer in single-user operation, you should be aware that there are important differences between MC-DOS and CP/M-86. There are fewer commands in CP/M-86 and the versions of CP/M-86 that Gifford sells do not support hard disks. There are fewer commands in CP/M-86, no virtual terminals, and the system console is the only workstation. Read the CP/M-86 OPERATING SYSTEM USER'S GUIDE for additional details on using CP/M-86.

2.3 Multiuser Concurrent DOS Control Characters

MC-DOS **control characters** are used to control programs and to edit command lines typed at a terminal. A control character is produced by typing a character while pressing the CONTROL key of your keyboard, just as you might type (A) by pressing (a) and the SHIFT key. The symbol ^ is used in this book to indicate that the next character is a control character. Thus, ^C means to press the CONTROL and C keys. The MC-DOS operating system generally makes no distinction between upper case (A) and lower case (a) letters. Exceptions are noted in this manual. There are no differences between upper case and lower case control characters, but in this manual capital letters are always used to symbolize control characters.

Applications programs which have their own control characters, such as WordStar and SuperCalc, generally do **not** recognize the MC-DOS control characters.

MC-DOS Program Control Characters

Although the control characters are intended to increase your control over the system, accidentally typing some of the them can have effects that may make you think that your terminal or the system is acting improperly. You should read the following descriptions of program control characters so that you can use them deliberately where they are helpful, and can recognize the effects of unintended control characters.

^C Aborts the process running on your current virtual terminal. This can be used to exit from most programs.

^O Suppresses screen listing of output. Unlike ^S, which makes the process pause that was creating the listing, with ^O the process continues, but its output is not displayed. If output was being echoed on the printer by ^P, a ^O stops the printer, but does not free it for other jobs. The screen resumes displaying your output after another ^O is typed, or after the process is completed and you press the RETURN key.

^P Sends all terminal output to the current printer. This command acts as a toggle. The first ^P echoes the screen output on the printer; the second ^P stops the echoing. A printer that is attached by a ^P cannot be used by other virtual terminals until you detach the printer with a second ^P. In a DR Net network, ^P cannot activate a remote printer.

^Q Resumes the terminal listing if a ^S command has been given to interrupt scrolling. Once ^S has been entered, any character other than ^Q will cause the terminal bell to sound. The terminal will ignore all commands except ^Q or the function key to change virtual terminals. The same virtual terminal that typed ^S must give the matching ^Q.

^S Temporarily stops terminal listing. This is useful to stop the scrolling initiated by commands such as TYPE or SDIR. Users familiar with CP/M should note that typing ^Q, not another ^S, resumes the listing. If by mistake you type ^S in a command line, the keyboard of your virtual terminal is disabled until you type ^Q.

MC-DOS Command Line Editing Control Characters

These are used to create **command lines** that tell the system what to do, or to correct typing errors in command lines.

^E Acts like a RETURN on your screen, but does not send anything to the computer. The cursor advances to the beginning of the next line, but the command is not processed. This is helpful with very long command lines.

^H Backspaces the cursor one space, deleting the previous character. Since the DELETE key repeats the previous (deleted) keystroke, ^H produces easier to read commands.

^I Equivalent to TAB. The cursor moves to the next tab stop on the line, whose default spacing is every eight columns.

^J Has the same effect as pressing RETURN. It produces a line feed and sends the command line to the computer.

^M Same as ^J or RETURN. The ^J or ^M characters are helpful in programming a keyboard function key to give multiple commands that must be separated by RETURNs.

^R Retypes the command line on a clean line. This may be helpful if you have used the DELETE key to echo errors. If you then type ^X, the original line remains displayed, but MC-DOS will ignore it, as if it had been erased.

^U Cancels the command line and moves the cursor down one line to begin a clean command line. The symbol # is printed to acknowledge the ^U, but can be ignored.

^X Cancels the command line and returns a blank line.

^Z Marks the end of a file, of a field, or of an operation.

Control characters not listed on these pages have no effect. Printable non-control characters from a keyboard are added to a command line. The BACKSPACE key deletes the last keystroke. See also the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE for additional information on MC-DOS control characters.

2.4 Summary of MC-DOS Commands and Files

The following pages provide a quick reference to the system commands and files available on MC-DOS. Many of the commands have powerful options, and you should refer to the more complete documentation of **Section 2.5** to use them effectively. Both an alphabetic listing is provided, and a listing of the commands and files clasified into functionally related groups.

Commands marked with an asterisk (*) are also described in the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE. Note, however, that although most of the Gifford enhancements take the form of supplementary commands and files, Gifford has modified the DATE command so that one Concurrent DOS option is not supported, and has modified various commands so that CONSOLE, GENCCPM, HELP, NET, PRINT, PRINTER, SYSTAT, TYPE, USER, and VCMODE have additional features, different displays, different defaults, or other discrepancies, sometimes only in small details, from the Digital Research Inc. documentation. The HELP, NET, PRINTER, and TYPE commands, however, are so different in Gifford's MC-DOS that the Concurrent DOS documentation for those commands is of limited use. **Appendix K** of this manual outlines the differences between standard Concurrent DOS 3.1 and Gifford's MC-DOS. If your system is part of a DR Net local area network, see Gifford's DR NET USER'S MANUAL for more information on the NET, CONFIG.NET, NAMSVR.DAT, NETUSERS, and SETNET commands and files used in networking.

Detailed documentation of Digital Research Inc.'s utilities ASM86, DDT86, ED, GENCCPM, GENCMD, and SYSTAT is beyond the scope of this manual, which is intended for the non-technical reader. **Section 2.5** provides a brief indication of what those routines do, but the interested reader should refer to the Digital Research Inc. documentation provided with MC-DOS, namely the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE for ED, and the CONCURRENT CP/M OPERATING SYSTEM PROGRAMMER'S UTILITIES GUIDE for the other programmer's utilities and for an explanation of Concurrent DOS error messages

Alphabetic Summary of MC-DOS Commands and System Files

***ABORT**	**Cuts short a process on another virtual terminal**
AP	**Multiuser appointment calendar**
*ASM86	Utility for programmers, a 16-bit assembler
ATTACH	Attaches a background process to a terminal
AUTOST	File for automatic multiuser start up
BUFFER	Controls a RAM drive buffer of the hard disk
CCPMINIT	Initialization program used with virtual terminals
*CHSET	Changes certain attributes of .CMD files
CLOCK	Sets and displays the system clock/calendar
CONFIG.NET	Data file used to configure DR Net
*CONSOLE	Shows current virtual and physical terminal numbers
CPY	A utility for copying diskettes or logical drives
CHRON	Schedules a program to run in background at a given time
***DATE**	**Displays clock/calendar's current time, date**
DAYFILE	Lists time of day after each command line
*DDT86	Programming tool, directly manipulates RAM memory
***DIR**	**Displays the names of files in a logical drive**
DOWN	Disables all terminals except the system console
DOWN.SUB	Optional file executed when shutting down the system
DSKRESET	**Resets disk drives (used when changing diskettes)**
DWN	Optional program executed when shutting down
*ED	Line editor that can create or edit text files
*ERA	Deletes files from logical drives
***ERAQ**	**Deletes files, but requires confirmation**
FILES	An advanced directory search utility
FIXSTEP	Changes the step rate on floppy disk drives
FMT	**Formats a floppy diskette, erasing any files**
*GENCCPM	Utility to create or modify MC-DOS operating system
*GENCMD	Utility that changes an .H86 file to a .CMD file
***HELP**	**Displays information to assist users**
HELPNDX	Used when creating or modifying HELP files
HISTORY	Optional data file which records system activities

* Command listed in CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE

Alphabetic Summary of MC-DOS Commands and System Files

*INITDIR	Reformats drive directory for timestamping
KEYS	**Defines the programmable keys of a terminal**
LOGIN	Optional program to individualize user login
LOGIN.SUB	Optional customization file executed at login
LOGOUT	**Logs an account off the system**
LOGOUT.SUB	Optional customization file executed at logout
LPRS	Data file containing printer/plotter descriptions
MAIL	**Electronic mail system message facility**
MAIL.HLP	Electronic mail "help" data
MAIL.NDX	Electronic mail "help" index
MAKE	Creates an empty file
MCDOS.HLP	MC-DOS operating system "help" data
MCDOS.NDX	MC-DOS operating system "help" index
MOTD	Message of the day display
NAMENODE	Automatically maintains the NAMSVR.DAT file
NAMSVR.DAT	Data file used in DR Net for system names
NET	Program to access DR Net resources
name.NET	Mapping template created by the NET MAKE command
NETMOTD	Message of the day display for the network
NETUSERS	List of account names for DR Net
NEWUSER	Security utility to maintain the PASSWD file
PASSWD	Data file of passwords and user descriptions
PASSWORD	Lets users modify their own login passwords
***PIP**	**Copies files; the most frequently used command**
*PRINT	Prints a file on the current printer
***PRINTER**	**Displays and sets the printer number**
PWASK.NET	Requires a password to connect in DR Net network
name.REM	**File to display individual reminders at login**
*REN	Changes the name of a file
RESET	Used if TTYS or LPRS files are modified
*SDIR	Displays extended disk directory

* Command listed in CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE

Alphabetic Summary of MC-DOS Commands and System Files

*SET	Sets attributes of files and of logical drives
SETMEM	Sets the upper limit on RAM allocated for programs
SETNET	DR Net utility used when CONFIG.NET is changed
***SHOW**	**Displays status of logical drives, labels**
SPACE	Shows used and empty capacity of logical drives
SPOOL	Sends files to specified printers
***SUBMIT**	**Processes a sequence of command lines**
SW	Used to execute 8-bit command (.COM) files
SYNC	Copies the files in disk buffer to the hard disk
SYSGEN	Copies system tracks of a diskette
SYSINIT.BAN	Banner file displayed before user login
SYSINIT	Optional program to customize startup
SYSINIT.SUB	Optional customization file executed during startup
*SYSTAT	Displays current status information about MC-DOS
TIME	Displays the time elapsed in executing programs
TIMELOG.SUB	File to update timelog database
TIMELOG.CMD	dBASE II command file to create timelog reports
TTYS	Data file containing terminal/modem description
***TYPE**	**Displays contents of an ASCII (printable) file**
UPTIME	Displays performance data
***USER**	**Selects or displays current user number**
USERINIT	Optional program to customize login for all users
USERINIT.SUB	Optional customization file executed at login
UTMP	Data file describing who is using the system
*VCMODE	Program that controls virtual terminals
VTTYS	Data file to configure virtual terminals
VT950.KEY	Data file for TeleVideo 950 key assignment
WHO	**Shows who is currently logged into the system**
WRITE	**Facility to send a message to another terminal**
WTMP	Data file containing a timelog information
ZWTMP	Background process used by timelog facility

* Command listed in CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE

All commands in boldface are those which are most frequently used in typical MC-DOS operations. All users should know these.

ABORT

ABORT program n

The ABORT command stops the execution of a **process.** A process is an executable command file that is running on any modem, terminal, or virtual terminal. In a valid ABORT command you must correctly specify **n,** the **virtual terminal number** that invoked the process. You do not need to enter the .CMD (16-bit) or .COM (8-bit) file extension of the process that you want to stop. An example of an ABORT command appears below:

```
0A>ABORT WS 4
```

This tells MC-DOS to abort WS.CMD or WS.COM on virtual terminal 4. If the program name and virtual terminal number that you supply do not match any currently active process, a message will appear to advise you that "Abort Failed." You cannot use ABORT to stop a process whose physical terminal is connected to two different systems within a DR Net network.

Great care must be exercised not to interfere with the work of other users, who may not wish to have their processes aborted. The WHO command (with the -A option) can be used to identify the virtual terminal number of a foreground or background process that you want to terminate. The virtual terminal number appears in the **con** field of a WHO display.

Note that the SYSTAT command with options C or U displays virtual terminal numbers, but does so in hexadecimal (base 16) notation. You must supply ordinary decimal numbers, not hexadecimal numbers, to designate the virtual terminal number (for example, 13, not 0D) in an ABORT command line.

The ABORT command is helpful when a terminal loses control of a process. Since most programs have normal exit procedures, or can be stopped by the ^C control character, it is rarely necessary to invoke the ABORT command. See the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE for additional information.

2.5 Documentation of Multiuser Concurrent DOS Commands

The following pages provide information on the command files and system files of MC-DOS, presented in alphabetical order. For each command, the acceptable forms of command line syntax are presented at the beginning of the description. A description of the command follows, and examples of how to use each command are provided.

The most general form of an MC-DOS command line is:

```
d: KEYWORD [options] d:file [options] d:file [options] @node
 \  \       \       \  \     \       \  \     \       \
  0  1       2       3  4     5       6  7     8       9
```

(Don't be alarmed— most command lines include only a few of the elements shown above.) The possible elements are:

1. KEYWORD — This is the name of the command, as it appears at the beginning of the description of each command in **Section 2.5** (or as renamed by REN).

2,5,8. OPTIONS or OPTION LIST — These are special symbols or words that apply either to the entire command or to a file or a list of files appearing in a command line. Some commands can have up to ten options in a single list. Several commands require that square brackets, [ ], enclose a list of options.

0,3,6. DRIVE DESIGNATION — This is a letter, followed by a colon (:), that indicates the logical drive affected by the command, usually ommitted if same as the current drive. Some commands accept a list of drives.

4,7. FILE NAME or FILE LIST — The file or list of files affected by the the command. Most commands require you to specify the file extension, but some do not.

9. NODE NAME — This shows the name of a DR Net node, prefixed by the @ symbol; only a few commands need this.

In addition to the standard notation used throughout this book, the following conventions are used at the beginning of each command description, where examples of syntax are presented:

Boldface **Boldface** is used to indicate information which **must be supplied.** Information not shown in boldface is optional.

UPPER CASE — UPPER CASE characters represent information that must be typed exactly as shown, letter for letter.

lower case — Words in lower case indicate information which the user provides, substituting some specific name, option, or modifier for the variable name given in the example.

file — The word **file** means file specification (the file name and extension), such as SC.CMD. A few commands which can only be used with files of a single type do not require that the file extension be specified.

@node — The symbol @ followed by the word **node** means that the name of a DR Net network node, prefixed by the @ symbol, must be specified in the command.

d: — The notation **d:** indicates a logical drive specification. The logical drive specification is an upper case or lower case letter ranging from A to P, followed by a colon (:).

c — The letter **c** indicates a single-character command, such as those which the AP, DATE or MAIL programs accept.

n — The letter **n** specifies a number, such as a user area number, printer number, or number of a virtual terminal.

... — Indicates a continuing list of indefinite length; do **not** use this notation in a command line.

[],;()-<>!? — All other special symbols like brackets, semicolons, commas, etc. are used in commands lines exactly as they appear. Many commands, for example, require brackets ([]) around a list of options (CHSET, DIR, ERA, ERAQ, PIP, PRINT, SDIR, SET, SHOW, SYSTAT, and TYPE).

AP

AP c

The AP appointment calendar allows you to schedule appointments and reminders for yourself, and review your calendar. Each account name listed in the PASSWD file can have a calendar. AP files are kept in user area 15 of drive A: of the system node.

Appointments

The information which can be retained for each appointment includes: the year, month, day of the month, day of the week, beginning time, ending time, the duration of the appointment, the number of days notice requested, as well as any message. Not all of this information is applicable to every appointment. For example, an annual appointment has no year specification. Also, the time of day does not always have to be specified. If it is not, then "anytime" is assumed. This will become important during the "search for time" option.

Displays

Displays are always printed in chronological order, with the most imminent appointments at the top of the list. After every four appointments, the AP program pauses to give you time to read the current screen display, and then continues when you press RETURN.

Entering Time Information

To specify time, in hours, on an AP command line you may enter numbers ranging for 100 to 2359, or you may enter a number in the range 1 to 12. The advantage of using the former is that the AP program is able to deduce the hour of the day, the minute of the hour, as well as am or pm. In the latter case, however, the AP program will interpret the entry as the hour of the day then AP will prompt for further specification regarding am or pm.

Operating Procedures

To use the AP program, type:

```
0A>AP MENU
```

The following menu will be displayed on your screen:

```
                Gifford Computer Systems
                  Appointment Calendar
          Version 4.0 of ap updated on Aug. 3, 1984


                    "login account name"

        A) Make an appointment.
        B) Change an appointment.
        C) Cancel an appointment.
        D) Display today's reminders.
        E) Display every appointment.
        F) Finished.
        G) Schedule a periodic event.
        H) Display a month's calendar.
        I) Display your week.
        J) Search for free time.
        K) Search by context.
        L) Someone else's calendar.
```

Menu Options

Enter your selection and press RETURN. Upper and lower case letters are treated identically when you enter your menu option. Whenever you press RETURN, regardless of whether you are in the main menu or a sub-menu, the AP program will assume that you intend to "default" to the current value. Your login account name will be displayed at the top of this menu.

A. Make an Appointment

This is the most frequently accessed option. Use this option to schedule a one-time specific event. You can schedule an appointment up to one year in advance. If you attempt to schedule an appointment that would overlap an existing appointment, the program will display a message on your screen indicating that a conflict exists. You will have the option of overriding the conflict and scheduling a simultaneous appointment, or selecting another time.

To schedule an appointment, type (A) and a usage example will be displayed on your screen. As indicated by this example, to enter a date you must type the date followed by the month followed by the time of your appointment. After you have entered this information, the program will repeat that data on your screen and ask you to verify your appointment. The program will then prompt you to write a one line note to yourself.

This mandatory note should contain information pertinent to your appointment. You will also be asked how many days notice you want before the actual date of your appointment. The limit for this field is 28 days. Any time you enter anything over 28 days, you will get an error message. The next time that you press (D) from the main menu to display your appointment reminders, you will see your message indicating the nature and date of your appointment.

B. Change an Appointment

To change the time specification or the message for an existing appointment entry, press (B). The program will list, the next appointment positioned at the top of the list, all of the appointments you have scheduled thus far. Each appointment on this list will include the appointment number, the date, time, amount of advance warning, and your note to yourself. The program will display a submenu from which you should select what appointment, or which aspect of an appointment is to be altered.

Make the appropriate choice from this sub-menu. If you wish to make more than one change, press RETURN and the same sub-menu will reappear on your screen. To return to the main menu, press (C) for "no more changes."

C. Cancel an Appointment

When you use this option, the appointment is eliminated from the list. The actual change to the data file that contains the appointment information is not enacted until you choose F (Finished) or L (Someone else's calendar) from the list of menu options. Appointments whose date have already passed are automatically removed from the system. Also, appointment entries with a blank message disappear automatically when you log out of the system.

To cancel an appointment, press (C) and the AP program will list all of your scheduled appointments with the most imminent appointment positioned at the top of the list. Designate which appointment you intend to cancel by entering the number at the prompt.

D. Display Today's Reminders

Reminders that are appropriate for the current day are automatically displayed when you choose this option.

Each appointment has associated with it a certain number of advance notice days. Advance notice takes weekends into account. For example, if you schedule an appointment for Monday with one (1) day advance notice, you will be notified on the preceding Friday. Also, you will be notified every day after the advance notice date and until the appointment has come to pass. For example, if you schedule an appointment for Wednesday with three (3) days advance notice, you will be notified on the preceding Friday, Monday, and Tuesday.

To display the current day's reminders, type (D) and a display similar to the one below will appear on your screen:

Today at 12:00 noon
call the bank

Tomorrow from 12:00 noon to 1:00 pm
racquet ball

Friday from 3:00 pm to 4:00 pm
staff meeting

E. Display Every Appointment

When you choose this option, a list of every appointment, in chronological order, will be displayed on your screen. All pertinent information relating to the appointments, e.g. index number, date, time, advance notice, and the brief message, will be displayed on your screen.

F. Finished

When you press (F) from the main menu, the appointment calendar file is updated, the program is terminated, and you are returned to the operating system. This method of exiting the program is recommended over typing Control C (^C), as the latter command leaves the appointment file unchanged.

G. Schedule a Periodic Event

This option is very similar to option A (make an appointment), however, this option allows you to schedule indefinite or repeating events. Possibilities include annual, monthly, weekly, and daily events. A birthday is an example of an annual event. Keep in mind that periodic events are not removed from the appointment system until you cancel them with option C or until their logical longevity has expired.

To schedule a periodic event, type (G) and the AP program will display the current day's date, and then prompt:

Every year?
Every month?

If you respond by typing (N) for "No" to the "every month" prompt, a list of the 12 months will be displayed on your screen. Select the month in which you want to schedule the periodic event and the program will prompt:

Weekly?

After you respond to this prompt by typing (Y) for "yes" or (N) for "no," the program will display a list of the parameters that you have specified thus far. Then it will prompt you to supply the time of day that you wish to schedule the event, the estimated duration of the event, a note to yourself, and how many days notice you request. After you have entered all of this information, you will return to the main menu.

H. Display a Month's Calendar

This choice simply displays a month's calendar. When the program prompts you to choose a month, a calendar for that month will be displayed on your screen. This option will only go eleven (11) months into the future.

I. Display Your Week

When you press (I), you will see a week's list of your scheduled engagements. The program will first ask you to specify the appropriate month, and then a starting date for the week that you wish to see. You will see your appointments for the seven (7) days following the starting date. If no hour specification was made for that appointment, no time is shown on the weekly list.

J. Search for Free Time

This option searches through the calendar in ascending chronological order for a free space to match your specifications. The program will always find one, no matter how far into the future it needs to go. The valid time for each day is restricted to a certain user-specified time band. This means that you will be prompted to supply a block of time in which the AP program should search for free hours.

First you will be asked from which month and date the program should start searching. The search will begin at midnight of the given date, and proceed forward into the future. You must also specify how large a time slot for which you are looking. This specification conforms to the rest of the AP program time entry format. Example: 100 for one (1) hour, and (30) for a half hour, 115 for an hour and a quarter, and 300 for three (3) hours. It is assumed that 9 to 5 might be a common time band. To convey these paramaters to AP, you must type 900 to 1700. You are asked the earliest and latest times that you will consider. After you have entered your search specifications, the program will ask you if you want to include weekends too.

To begin searching for free time, type (J), the program will display the current day's date on your screen, and then prompt you to select the appropriate month.

When you designate the month in which you want to search for free time, the program will display that month for you on screen.

You must supply the date from which the program should begin searching for free time. You will then be prompted to supply a series of specifications indicating the general time of day for which you wish to search for free time.

Free time slots will always begin on a multiple of fifteen minute intervals like 1:00, 1:15, 1:30, etc. Only time slots which fit entirely into the period specified will be reported. When a free time slot is found, it will be displayed on your

screen, and you will be asked if you want the program to continue searching from that point on, or break off the search. If you want the search to continue, it will recommence immediately after the available time slot that was just displayed on your screen.

For example, suppose that you are searching for a three hour time slot and the program finds one, from 9:00 to 12:00, but you are unhappy with this one. If you instruct the program to continue searching, the next possible slot would be from 12:00 to 3:00, not from 9:15 to 12:15.

K. Search by Context

All appointments whose message contains the same word or words can be found using this search option. These appointments will be displayed in chronological order, with the most imminent appointment positioned at the top of the list. Upper and lower case are equivalent during this context search. This means that if you want to get a list of all of the appointments you have had with a person with the last name "Jones," you can enter jones or JONES. The AP program will search for both.

To search for something by context, press (K) and the program will request which pattern it should search for.

In response to this prompt, you should type a key word that you know occurs in a certain category of engagements that you have set up. For example, suppose that you meet regularly with Mr. Jones, and you want to find out how many times this month you have had appointments with him. To do this, type JONES after the prompt, and the program will display for you all of the meetings that you have had with Mr. Jones.

L. Someone Else's Calendar

By default, the AP program operates on the calendar of the currently logged-in account. This account name is displayed just above the display box of the AP main menu. Option "L" (someone else's calendar) enables you to view or change the calendar of another account. If you choose the "L" option, a list appears on screen of all of the accounts for whom appointment calendars (files called **name**.AP) exist in user area 15 of drive A: in a

non-networked MC-DOS system, or on the system node of a network. (This ability to access other accounts can be restricted by the system manager by setting a write-protect or read-protect file password on individual calendars. A protected calendar can only be modified or accessed if the same password were the default terminal password. See the description of SET in **Chapter 2** for details.)

When you press (L) from the main menu, the AP program will ask you to supply the account name whose calendar file you intend to access. A file **name**.AP must already have been created by the account previously using AP to schedule an appointment. If AP has a record of that account, you can schedule appointments or display existing appointments, just as if you had logged in using that login account name and password.

In a DR Net network, every account name listed in the NETUSERS file can have a calendar. In an MC-DOS system without networking, valid account names are listed in the PASSWD file.

ASM86 file

ASM86 is a programmer's utility that is not required in ordinary operations. The ASM86 program assembles assembly language statements and produces an object file in hexadecimal format. For example:

```
0A>ASM86 ECHO
```

This will take the file ECHO.A86 and produce a file ECHO.H86.

See the CONCURRENT CP/M OPERATING SYSTEM PROGRAMMER'S GUIDE for additional information.

ATTACH

ATTACH process

The ATTACH command attaches a process to its console. This command is used for multitasking. Processes can be initiated on a terminal and then detached with a system function call. Detached processes will run until they have completed execution, or require terminal input or output, in which case the process will wait to be attached. ATTACH allows the user to define which detached process is to be attached. Processes which are waiting for a terminal when a user gives the LOGOUT command will continue to wait until they are attached or aborted. For example:

```
3C>ATTACH GENLEDGR
```

attaches the background process GENLEDGR to the current virtual terminal. Programs can be detached from any virtual terminal by MC-DOS function calls. **Appendix J** illustrates some dBASE II routines that could be used with function calls (**page J-4**) to detach a process. Attach is a programmer's utility not required in ordinary MC-DOS operations.

AUTOST

The AUTOST file is used during system startup as a flag to jump automatically into multiuser mode. AUTOST must be located in user area 0 of drive A:. If the file does not exist, the system remains in single-user mode until the user at the system console gives the LOGOUT command. It is recommended that the system manager perform administrative functions such as setting passwords when the MC-DOS system is in single-user mode.

AUTOST is created by typing:

```
0A>MAKE AUTOST
```

If AUTOST does not exist on user 0 of drive A:, the system console will not require its user to log in until after the execution of the logout command. This means that without the use of AUTOST there is initially no password protection for the system console.

During the system boot sequence, the operating system automatically erases any UTMP files, then creates an empty UTMP file and puts a special write password on the files UTMP and PASSWD. If AUTOST does not exist, this process takes place only after the execution of the logout command. Maintenance of this file is under the jurisdiction of the system manager.

BUFFER

The BUFFER command is run automatically when the system boots up. It allocates a cache buffer to your hard disk. If you type BUFFER, the following message will appear on your screen:

The buffering system is already loaded.

The following files must exist on user area 0 of drive A: in order for the BUFFER command to execute properly:

BUF0.CMD
BUF1.CMD
BUF2.CMD
BUF3.CMD

These files can be found on your MULTIUSER CONCURRENT DOS SYSTEM AND UTILITIES DISKETTE.

CCPMINIT

CCPMINIT is a program that sets up the characteristics for Virtual Terminals under the MC-DOS operating system. You must have the CCPMINIT command file on user area 0 of drive A: in order for the Virtual Terminals feature to operate. If this file is not found during the boot process, then the Virtual Terminals feature will not work.

When the system is first booted, the CCPMINIT program reads the VTTYS files and configures virtual terminal tables that are resident within the operating system.

CHRON

CHRON is a program which schedules a program to run on the system in the background at a given time. You can use CHRON to run a job unattended at night when no one else is on the system. The CHRON program runs detached from the terminal. This means that if any program requires terminal input, that program cannot be executed with CHRON.

The CHRON program requires three files: CHRON.CMD, CHRONTIM, and CHRONTAB. KRON.RSP invokes the CHRON program. CHRON creates and maintains the CHRONTIM file. Every time CHRON is executed, CHRON updates CHRONTIM. KRON.RSP runs every minute to check the date and timestamp on CHRONTAB. If CHRONTAB has been modified, KRON.RSP will run CHRON. If there has been no change on the datestamp of CHRONTAB, KRON will check to see if there has been any change of the datestamp on CHRONTIM. If the current date and time is greater than or equal to the date in CHRONTIM, then KRON.RSP will run the CHRON program.

CHRON.CMD and CHRONTIM must reside in user area 0 of drive A: and should be SET to the SYS attribute.

The CHRON program requires 96K of RAM plus the amount of RAM necessary for the scheduled command to be executed. For example, to execute a PIP command you would need 96K for CHRON and 12K for PIP.

The CHRONTAB file contains all of the useful information which CHRON reads to execute a command or program at a given time and date. The CHRONTAB file must reside in user area 0 of drive A: and must be SET to read-write (RW). CHRONTAB can be edited while in non-document mode of WordStar. You are provided with a sample CHRONTAB file called CHRONTAB.SAM.

The CHRONTAB file contains the following fields:

```
;COUNT MINUTE HOUR MONTHDAY MONTH WEEKDAY OUTPUT COMMAND
  *      *     *     *       *     *      -     command

 \ \     \     \     \       \     \      \     \
  1 2      3     4     5       6     7      8     9
```

1. A semicolon at the beginning of a line signifies comments. Each field should be separated by at least one space.

2. **Count** represents the number of times that the command should be executed. This amount is decreased by one each time the command is executed until the count is zero, at which time the entry is automatically deleted from the CHRONTAB file. A negative count causes the command to be executed continually at the scheduled times until that entry is deleted from the CHRONTAB file.

3. **Minute** of the hour.

4. **Hour** of the day (24 hour clock).

5. **Monthday** (from 1 to 31 depending upon the month).

6. **Month** (January = 1, February = 2, March = 3, etc.)

7. **Weekday** (Sunday = 0, Monday = 1, Tuesday = 2, etc.)

8. **Output** is the standard output file. This file is used for capturing output of the command that is executed. If the name of this file begins with a plus sign (+), then CHRON will append the output to the end of that file. To leave the Output field blank, enter a hyphen in that field.

9. **Command** specifies the program or command which should be executed by the CHRON program. **Do not run programs that require input from the terminal.** If the command that you execute requires terminal input, that command will wait forever (until ABORTed) since CHRON cannot attach any command

to a physical terminal. This field can take a command and any arguments which you would normally enter at a command line. This field can take as a command, the name of a SUBMIT file as long as none of the commands within the SUBMIT file require any terminal input. For example, in this field you can specify:

```
PIP A:[g0]=B:*.*[g5VROW]
```

to PIP all files from user area 5 of drive B: to user area 0 of drive A:.

For each field which takes numbers for arguments (COUNT, MINUTE, HOUR, MONTHDAY, MONTH, and WEEKDAY), multiple arguments for a single field should be separated by commas. An asterisk in one of those fields tells CHRON to always match the current date or time.

If the MINUTE field has a zero as its only entry, then the command specified on that same line is executed once an hour on the hour. If the MINUTE field has the following entry:

```
0,5,10,15
```

then the command is executed on the hour, five minutes past the hour, ten minutes past the hour, and fifteen minutes past the hour.

```
; MINUTE     HOUR MONTHDAY MONTH WEEKDAY

0,5,10,15    *      1        3       1
```

This entry will execute the command four times every hour on every March 1 that falls on a Monday. If you entered a 3 in the hour field, then the command would be executed at 3:00 a.m., 3:05 a.m., 3:10 a.m., and 3:15 a.m. on every March 1 that falls on a Monday.

A percentage sign (%) followed by a number in one of the five fields which takes numbers for arguments tells CHRON to execute the scheduled command on a periodic basis. That is, if the minute field had an entry "%3," then the command would be executed every time that the current minute was a multiple of three.

An example of a CHRONTAB entry follows:

To execute a tape backup to occur at 11:00 p.m. every night, edit the CHRONTAB file in non-document mode of WordStar or a compatible text editor and make the following entries:

```
;COUNT MINUTE HOUR MONTHDAY MONTH WEEKDAY OUTPUT        COMMAND
  -1    0      23    *        *     *       +b0:tip.out  ntip input.tip
```

Output is appended to the end of the file called TIP.OUT which is located in user area 0 of drive B:. At 11:00 p.m. every night, CHRON executes the NTIP program which reads input from the file called INPUT.TIP. This command will be executed every night until the entry is deleted from the CHRONTAB file. If you only wanted to execute this command for a period of five nights, place a "5" in the COUNT field instead of a "-1."

The CHRON program requires an available physical console with an FF console number to execute a command. CHRON will look for an available physical console with the FF console number and if found will load the process. To see if the system has a physical console available with the FF console number, run the SYSTAT program and specify (C) for consoles. This will display the physical consoles which are available for your system.

When generating your XIOS file, you must have at least one more virtual terminal number than you need. These unassigned virtual terminals will have the console number of FF.

CHSET

CHSET d:**file** [option=status, ..., option=status]

The CHSET command modifies or displays special attributes of 16-bit command files. Any file named in a CHSET command line is assumed to have the command file extension .CMD, and only files of type .CMD can be modified or described by CHSET. CHSET can show or modify three attributes of .CMD files:

(1) whether several users or terminals can share simultaneously the same copy of program code in RAM;
(2) whether the program uses an Intel 8087 or 80287 arithmetic coprocessor (device that speeds numerical calculations);
(3) whether execution is suspended when a user switches virtual terminals and the program becomes a background process.

The eight options available for CHSET are:

CHSET d:file	= Displays the file's status on shared processing, 8087 usage, multitasking.
CHSET d:file [SHARED=ON]	= Allows shared processing from a single image of the file in RAM.
CHSET d:file [SHARED=OFF] (DEFAULT)	= Prevents shared processing from a single image of the file in RAM.
CHSET d:file [8087=ON]	= Prevents the program from running unless an 8087 or 80287 is available.
CHSET d:file [8087=OFF] (DEFAULT)	= Use this setting for programs that cannot use an 8087 or 80287.
CHSET d:file [8087=OPT]	= Allows the program to run with or without an available 8087 or 80287. If no coprocessor is available, the program must emulate in software the functionality of an 8087 or 80287.

CHSET d:file [SUSPEND=ON] = Prevents multitasking: execution is suspended if the program is not the foreground screen.

CHSET d:file [SUSPEND=OFF] (DEFAULT) = Allows a non-foreground process to output.

One, two, or three CHSET attributes can be specified in a single command line. For example, the following CHSET command:

```
0D>CHSET A:DMC [SHARED=ON, 8087=OPT, SUSPEND=OFF]
```

specifies that the file DMC.CMD on user area 0 of drive A: is set for shared processing, use of an 8087 or 80287 if one is available, and with processing not suspended in background mode. If the program DMC is written in a way that supports these features, several users (or the same user on several virtual terminals) could share use of an 8087 or 80287 (if a coprocessor were physically present).

Unless CHSET is used to set these special attributes, all of them are OFF, the default status. For example:

```
0A>CHSET NW
```

The system responds with a display of the status of NW.CMD:

```
NW       .CMD settings are [8087=OFF,SHARED=ON,SUSPEND=OFF]
```

Not all command files can actually take advantage of some of these attributes. The "header fields" controlled by CHSET can interfere with a program's capability of being shared [SHARED=ON], of utilizing an 8087 or 80287 [8087=OFF], and of producing output as a background process [SUSPEND=OFF].

Note that a command of the form:

```
5B>CHSET AJAX [SUSPEND=ON]
```

will stop a program (here "AJAX.CMD") from executing if it is subsequently detached from the screen, regardless of whether the program is doing console I/O. The above command would override a VCMODE BUFFERED or VCMODE DYNAMIC command that otherwise would allow all of the background processes of a given virtual terminal to create output.

Using CHSET to assign special attributes to a command file will not provide the file with new capabilities unless the program was originally written to support those features, just as awarding a diploma will not necessarily enable an illiterate recipient to read. CHSET only instructs the operating system to treat a program as if some indicated capability were present. If it is not, the program cannot take advantage of CHSET.

For example, using CHSET to set [SHARED = ON] will not allow multiple users to share the same copy of a program that mixes code and data, nor will setting [8087=OPT] allow a spelling checker program to use an 8087 or 80287 coprocessor if its code performs no arithmetic. You will have to experiment to see if CHSET is helpful for any specific applications program. See the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE and CONCURRENT CP/M OPERATING SYSTEM PROGRAMMER'S GUIDE for additional information.

CLOCK

CLOCK HELP
CLOCK @node

The CLOCK command is used to set the system clock/calendar or to display the time and date. CLOCK can be used on an individual MC-DOS computer, or within a DR Net network to display or set the clock/calendar on either the local node or any remote nodes. Many of the features in MC-DOS use the system clock to determine the current day and time. It is important that the system manager make certain that the system clock is always set accurately. Since the DATE command can read the clock, the system manager may decide to set a password on CLOCK to prevent unauthorized setting of the clock/calendar.

MC-DOS uses the real-time hardware clock to set the time. The DATE command can also be used to display the time of day. Unless the NiCad battery backup is disconnected from your system, you should only have to set the time once. The NiCad battery lasts for about two years. When you type CLOCK, the program responds:

GIFFORD COMPUTER SYSTEMS
System Clock Maintenance Program Rev. 1.3

OPTION MENU

S = Set time and date.
T = Set time only.
D = Set date only.
P = Print date and time once.
C = Print date and time continuously.
X = Exit program.

Option:

Press (S) to set the time and date. Time is entered in a 24-hour clock format (such as 14:22 for 2:22 pm). The date is entered with the number of the month, day, and year between slashes: 6/21/84 for June 21, 1984.

You can set the "time only" by pressing (T), or you can set the "date only" by pressing (D). To check the date and time, press (P). You will see a display similar to the one below:

```
Current time: 09:52:54
Current date: 06/21/84
```

Press (C) to continuously print the time and date on your terminal screen.

Use in DR Net Local Area Networks

In a DR Net local area network, the @node option of CLOCK can be used to set the time and date on any remote nodes whose names follow the symbol (@) in the command line. A remote node must be up and running for you to set its clock/calendar. The CLOCK program is helpful if you want to synchronize the times of all of the systems on a DR Net network. You can set the clock/calendars or display the current time and date of all the remote nodes by typing:

```
0A>CLOCK @
```

or you can set the clock of the SALES system node by typing:

```
0A>CLOCK @SALES
```

The program displays the menu:

GIFFORD COMPUTER SYSTEMS
System Clock Maintenance Program Rev. 1.3

Command Menu Options:

S = Set time and date from this node's setting.
P = Print date and time for the list of nodes.
X = Exit program.

Option:

Press (S) to set the time and date for the nodes that are designated in the command line. The time and date are set from the time and date of the local node. Time is entered in a 24-hour clock format (such as 14:22 for 2:22 pm). The date is entered with the number of the month, day, and year between slashes: 6/21/84 for June 21, 1984.

To check the date and time, press (P). You will see a display similar to the one below:

Current date and time at node:

sales 08/21/84 09:32:11

CONFIG.NET

The CONFIG.NET file is used only in DR Net local area networks. It provides the network input/output system (NIOS) with information about network configuration and default resource mapping. By changing the CONFIG.NET file the system manager can modify features of the network initialization process. If an operator would prefer to have a terminal attached to the network by a different resource mapping, the NET command can change CONFIG.NET default resource mapping assignments.

The CONFIG.NET file is read by the SETNET program, invoked during network initialization, which uses that information to configure the network. A sample CONFIG.NET file is provided with the DR Net system diskette. If you want to change the sample file, use ED, WordStar (non-document mode), or a similar text editor. A sample CONFIG.NET file is listed below:

```
password   NETWEGHT              ;Node password, also default password
node       1                     ;Local node number
default    0                     ;Default node number
servers    10                    ;Number of server processes
requester  6                     ;Number of requester processes
buffers    4                     ;Number of message buffers
timeout    5                     ;Seconds till timeout
rct        9                     ;Number of RCT's
private    f: g:                 ;Private drives (no remote access)
drive      l: = a:               ;Maps a: drive of node 0 onto l:
drive      m: = b:               ;Drive m: is b: drive of node 0
drive      n: = c:               ;Drive n: is c: drive of node 0
printer    5  = 1 3              ;Map printer 1 of node 3 as printer 5
queue      MXSPL = MXSPL 02      ;Map the queue of node 2
queue      SPLIN = SPLIN 02
queue      SPLOUT = SPLOUT 02
nodename   MARKETIN              ;Local node name
```

For a more detailed explanation of the CONFIG.NET file, read Chapter 3 of Gifford's DR NET USER'S MANUAL.

CONSOLE

The CONSOLE command displays the number of the user's terminal, as found in the TTYS file, as well as the number of the foreground virtual terminal screen.

Example:

```
0A>CONSOLE
```

The system responds with a display of the form:

```
Physical terminal is 2
Virtual  terminal is 8
```

The virtual terminal number is needed by the ABORT command.

CPY

CPY -V

The CPY program enables you to copy an entire diskette or an entire logical drive of a hard disk. Rather than copy individual files like the PIP program, CPY transfers the entire image of one diskette or logical drive (the "source") onto another drive or diskette (the "destination"). The destination is usually a blank diskette or an empty logical drive, since CPY erases whatever information may have been in the destination drive. Because it copies the exact image of the source, CPY does not "pack" files into contiguous tracks unless they were already arranged in this way on the source. Thus, PIP is used more often than CPY.

The CPY program is a convenient way to back up diskettes. When used with hard disks, the program can copy one drive onto another, or in 84 megabyte systems, CPY can copy a drive onto the removable platter. When copying diskettes, CPY requires that both diskettes have the same number of bytes per sector (128, 256, 512, or 1024), and that both be either single or double sided. Typing the keyword CPY displays the main CPY menu:

```
0A>CPY

Gifford Computer Systems Copy Utility V2.0

Do you want to copy:
        SYSTEM tracks only? (type S)
        DATA tracks only?   (type D)
        ALL of the disk?    (type A)
Exit back to the System     (type X)
```

Select the appropriate response (with no RETURN after it). The program prompts:

```
Source drive? (A, B, . .)
```

The source drive is the drive which contains the information you want to copy. When you respond, the program prompts:

Destination drive? (A, B, . . .)

The destination drive is the drive onto which you want the information to be copied. When you respond, the program prompts:

Put source disk on ...
Put destination disk on ...
Then type RETURN

Put the source and destination diskettes in the proper drives. It is extremely important that you have the source disk in the designated drive. You will erase information on the source diskette if you reverse the two. If you have any doubt about the identity of the source or destination drive, abort the program with ^C, or type from another virtual terminal:

0A>SDIR d: [user=all]

where **d** is the drive designator that you will use with the CPY program. Now press RETURN; if the diskettes have the same format, the program copies each track. When all of the tracks are copied, the program prompts you for the next copy operation. If you type:

0A>CPY -V

the CPY program will bypass the data verify pass on the destination drive. If the CPY program gives you error statements, confirm that the diskettes in the source and destination drives are physically identical (same sector size, and both single or double sided).

Copying Diskettes on Single Floppy Drive Systems

You cannot directly copy diskettes to diskettes on systems that support only one floppy disk drive using the CPY program. If you have a hard disk, you can PIP all of the files on a diskette to an empty directory area. Then give the DSKRESET command, replace the source diskette with a formatted destination diskette, and PIP the files from the hard disk to the destination diskette. See **2.5-PIP** for a description of PIP.

If no empty directory area is available on the hard disk, set ON the Archive attribute to all files in some directory area of the hard disk, and PIP the source diskette files to this directory area. Give the DSKRESET command. Now you can use PIP with the [A] option to copy the files to a destination diskette. If you do not need copies of the source files on the hard disk, use the ERAQ command to delete them.

In a DR Net network, a diskette can be copied to a destination diskette on a drive elswhere in the network, provided that the destination drive has been reset by a DSKRESET command and attached to your virtual terminal by a NET command. You must use PIP, however, rather than CPY, to transfer files between a 5.25" diskette and an 8" diskette.

DATE

DATE c

The DATE command displays the current time and date on the terminal screen. To display the date, type:

0A>DATE

The program responds with a display like the following:

Wednesday April 18, 1984 9:08 pm

For a continuous display that shows the time to the nearest second, type any character after the keyword "DATE." An example follows, although this printed manual cannot imitate the continuous display of the advancing seconds:

0A>DATE C
Wednesday April 18, 1984 9:08:12 pm

With any character as an argument, the displayed time (and eventually the date) will advance as the clock/calendar records the passage of time. Pressing any key stops the continuous display and returns you to the system prompt. The ^P toggle with DATE prints the date and time to your curent printer. (Remember to free the printer with a matching ^P.)

Digital Research Inc.'s SET option of the DATE command, to set the clock/calendar, is not implemented in Gifford's MC-DOS. Use instead the CLOCK command to set the clock/calendar.

See the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE for additional information.

DAYFILE

DAYFILE options

The DAYFILE command controls whether the screen lists after each command line the **time of day** (hours/minutes/seconds for a 24-hour clock), and the **logical drive and user area** from which the command file is loaded into RAM. You can toggle the DAYFILE feature by two options, ON (to enable the display) and OFF (to disable it). Any other characters display a usage message:

```
3A>DAYFILE ?
Usage is "DAYFILE ON" to enable the dayfile option
or "DAYFILE OFF" to disable the dayfile option.
```

If you type the keyword DAYFILE with no argument, the screen displays the current status of the dayfile option, and indicates how to change it. After a DAYFILE ON command enables the dayfile option, a dayfile line follows any command. For example:

```
5B>MAKE AUTOST
14:32:52 A:MAKE.CMD (User 0)
```

If the command DAYFILE OFF were given, the same MAKE AUTOST command would not produce the second line of the display above.

The DAYFILE option of the GENCCPM command also causes the current time to be displayed after each command, but using GENCCPM instead of the DAYFILE command to control this feature is much less convenient. Commands that are resident system processes or that are built into the MC-DOS shell (like LOGOUT, RESET, SYNC) are not affected by the DAYFILE command. Since a DAYFILE command affects every terminal on an MC-186 system, the system manager may elect to assume jurisdiction of the DAYFILE command. All of the examples of commands lines in this manual (except the example above) represent the screen display as it would appear with the dayfile option OFF.

DDT86

DDT86 filename

The DDT86 utility enables a programmer to examine and modify programs and files, and to directly address the CPU or any location in random access memory (RAM). Do not use DDT86 when other users are active, since an error in using DDT86 could disrupt their work or even disable the operating system. It is recommended that the system manager set a read-protect password on the DDT86 program, to prevent its unauthorized use.

Information that is loaded into RAM is stored in specific **addresses** within RAM, potentially ranging from 0 (where MC-DOS itself begins) to the upper limit of memory in your system. DDT86 uses a **hexadecimal** notation to refer to locations in RAM. In hexadecimal numbers, the first sixteen numbers are written as single digits: the familiar 0, 1, 2, 3, ..., but with "9" followed by the special hexadecimal numerals A (= 10), B (= 11), C (= 12), up to F (= 15). In the previous sentence, ordinary "decimal" equivalents appear at the right of each equal sign. Use of hexadecimal notation allows DDT86 to write big numbers more compactly. For example, hexadecimal "FFFF" = 65,535 in ordinary numbers, but the hexadecimal version needs only four places, instead of the five figures required to write "65,535."

In DDT86 (and in other programmer's utilities like SYSTAT), RAM addresses where files are stored are written as a pair of 4-digit hexadecimal numbers, separated by a colon (:). The first part is called the "segment," and the second part is called the "offset." The segment shows where to begin counting; the offset must be added to the address. This may seem complex, but people use a similar notation in a date like "12/25/84" which we understand to mean the 25th day (=offset) of the 12th month (=segment) of the 84th year. (Here "84" is really a second offset: the century "1900" is understood as added to the year.) Remember that DDT86 interprets a number like A000:7850 as a specific RAM location, namely A000 (hexadecimal) "paragraphs" (16-byte units) plus 7850 (hexadecimal) 8-bit "bytes" above the lowest address (0000:0000) in RAM. In the description of DDT86 command options on the next page, "s:o" means "segment:offset." (See **Appendix N**).

Typing its keyword invokes DDT86, as illustrated below:

```
0A>DDT86
DDT86 1.2
-
```

At the DDT86 prompt (-) various letters (with RAM addresses) can be entered as command options. Several are listed below:

OPTION	MEANING
As:o	Begin entering assembly language code at s:o
Bs:o1,o2,o3	Find any differences in two blocks of RAM, one from s:o1 to s:o2, and the other extending to s:o3
^C	Exit from DDT86
Ds:o	Display contents of the 192 bytes of RAM beginning at s:o, in both hexadecimal and ASCII formats
Ds:o1,o2	Similarly display everything from s:o1 to s:o2 Any key stops the scrolling, returns the - prompt
Efile	Load a specified executable command file into RAM
Fs:o1,o2,c	Fill all the RAM from s:o1 to s:o2 with character c
Gs:o	Begin execution of command file, starting at s:o
H#1,#2	Calculate sum (#1 + #2), and difference (#1 - #2), for hexadecimal numbers #1, #2
Ls:o1,o2	Display contents of s:o1 to s:o2 as assembly code
Msl:o1,o2,o3	Move contents of (sl:o1 through sl.o2) to an area beginning at sl:o3 (or at s2:o3, if s2 specified)
QIn (or QOn)	Read input from port n; or write output to port n
Rfile,s:o	Read a file from a disk into RAM, starting at s:o If s:o not specified, DDT86 displays start address
Ss:o	Display contents of s:o, replace with keyboard input
SRs:o1,s:o2,x	Search for pattern x from s:o1 to s:o2, and display where x begins, if x is found
Tn	Execute n program steps, displaying CPU states
V	Show the location of a file manipulated by E or R
Wfile,s:o1,o2	Save on disk as "file" the contents of s:o1 to s:o2.
Xr	Display and modify register r of the CPU

Use the MC-DOS control character ^C to exit from DDT86. The CONCURRENT CP/M OPERATING SYSTEM PROGRAMMER'S GUIDE provides additional detail on the use and capabilities of DDT86.

DIR

DIR[S] d:file[Gn]

The DIR command displays a list of files in a user area of a specified drive, or that resemble the indicated files. The DIR command is used to determine what files are in a specified directory area. DIR accepts wildcard symbols (* and ?) in file names.

In its simplest form, DIR without any arguments or modifiers displays all of the files with the DIR attribute in your current directory area that have the DIR attribute. The command:

4C>DIR

displays the names of all the files with the DIR attribute on user area 4 of drive C:, in a four-column display. File names with fewer than eight characters before the extension are listed as if they had blanks preceding their file type. The above command provides the same display as "DIR *.*," since the default value is "all names and all extensions." If a display has more directories than will fit in a single screen, you must use the ^S system command to stop the scrolling (^Q restarts it), or request a more restrictive listing. For example, to display only filenames starting with "M" in the same directory, type:

4C>DIR M*.*

To list only the DIR files with one-, two-, three-, four-, or five-letter names and with the extension .BAK in your current user area, type:

4C>DIR ?????.BAK

To list files in a user area other than the one which you are in, use the [Gn] option with **n** the number of the user area. For example, to list the names of all the files that have the DIR attribute on user area 6 of drive B:, give the command:

```
0A>DIR B:[G6]
```

Each account, however, has a unique entry in the PASSWD file which specifies its accessible user areas. The [Gn] DIR command cannot be used to display the names of files in directory areas where you are not allowed access in your login PASSWD file.

The [S] option causes the DIR command to display the names of **all** files in a specified directory area, including both of DIR and SYS attributes. (See the SET command for modifying the DIR or SYS attribute of a file.) For example, the command:

```
6C>DIR[S] A:*.CMD[G0]
```

will display all of the file names on user area 0 of drive A:, regardless of whether they have the DIR or SYS attributes. Unlike the SDIR command, however, the DIR command with the [S] option does not indicate which attribute a specific file has.

The DIR[S] command can be used, in conjunction with the ^P control character, to print out the names of all of the files that are in a single user area on an archival diskette. This listing can be attached to the paper sleeve of the diskette, or kept in a bound catalogue, to assist in retrieving files from diskette storage. Remember to repeat the ^P command when you no longer want the printer to echo the screen of your terminal.

See the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE additional information.

DOWN

The DOWN command disables all physical terminals except the system console and writes the contents of the cache buffer to the hard disk. The console screen displays the message:

The system is down.

All other terminals become inoperative until the MC-DOS system is reloaded by pressing the RESET button.

Example:

```
OA>DOWN
```

This command should only be given by the system manager. Before giving a DOWN command, type WHO -A to be sure that no other users are on the system, since DOWN will disable their terminals. If WHO -A shows that other users are logged in, use the WRITE command to warn them that you plan to shut down the system.

DSKRESET

DSKRESET
DSKRESET d:

The DSKRESET command resets the floppy disk drive. Type the DSKRESET command **before** you change a floppy diskette. If you do not give an argument to DSKRESET, all drives are reset. You can specify the individual drives that you want to reset.

Examples:

0A>DSKRESET

This resets all floppy disk drives.

0A>DSKRESET B:

This resets floppy drive B:.

0A>DSKRESET D:,E:

This resets floppy drives D: and E:.

DOWN.SUB

DOWN.SUB is an optional file containing commands which are to be executed as part of the shutdown procedure. Read about the SYSINIT.SUB and USERINIT.SUB for ideas on how this file might be used.

An example of a DOWN.SUB file is:

```
;example of a DOWN.SUB file
TIP
```

The maintenance of this file falls under the jurisdiction of the system manager. It is discussed again in **Chapter 3.**

DWN

DWN.CMD is an optional program which will run after the DOWN command is given. This program file can be used to customize your system by executing a program which you want to be part of your shutdown procedure. It must be located on user area 0 of drive A:.

An example of a command to create a DWN.CMD command file is:

```
0A>PIP DWN.CMD=DATE.CMD[VRW
```

This command would cause the system to always display the date and time when you shut down the system.

The maintenance of this file falls under the jurisdiction of the system manager. It is discussed again in **Chapter 3.**

ED

ED d:file1 d:file2

ED is a text editing utility provided by Digital Research, Inc. It can be used to create and modify various files used by MC-DOS, including LPRS, MOTD, TTYS, VTTYS, and the .SUB files used by the system manager to customize MC-DOS.

Unlike **character-oriented** text editors like WordStar that are commonly used in office work, ED is a **line editor** that operates by creating lines of text in a memory buffer. Although line editors like ED are widely used in programming, engineering, and scientific applications, readers who have experience using character-oriented text editors may regard ED as not very easy to use. Gifford recommends the use of other editors, such as WordStar, instead of ED for most business applications.

Readers who have no other text editor, however, can create or modify a file by typing the keyword ED followed by the name of the file that you wish to create or modify. For example:

2B>ED TEST.DOC

will create a new file TEST.DOC if the file does not already exist. If TEST.DOC already exists, ED will allow you to modify it, and will save the modified version as TEST.DOC and rename the original file TEST.BAK. If a drive name (like "C:") had followed "TEST.DOC" in the command line above, the edited version would have been saved on that drive, rather than on your current drive.

You can also provide two file names in the command that invokes ED. The first file is the "input" file that you are modifying, and the second file is the "output" file that ED will create. If you type the ED keyword without any file names, the program will prompt you to identify the input and output files. An extensive description of ED and a tutorial on its use is provided in Section 4 of the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE.

All of the MC-DOS editing control characters (**page 2.3-3**) can be used with ED, but ^C, ^P, and ^S (**page 2.3-2**) do not work normally when ED is in "insert mode" (after **I** or **i**). Listed below are some command options that can be used with ED:

Command	**Action**
#A	Put entire input file into memory buffer
nA	Add next **n** lines from input file to memory buffer
B, -B	Move pointer to beginning (B) or end (-B) of buffer
^C	Abort an ED macro command (option M below)
nC, -nC	Move pointer **n** characters forward (nC) or backward (-nC)
nD, -nD	Delete **n** characters before (-nD) or after (nD) pointer
E	Exit from ED, save edited file, return to MC-DOS
Fstring^Z	Find a string of characters, ignore upper/lower case
fstring^Z	Find exact character string (^Z marks end of string)
H	Save file and reedit, using current output file as input
I	Enter insert mode, translating lower to upper case
i	Enter exactly (upper/lower case) keystrokes to buffer
Istring^Z	Enter a string of characters at the pointer
J, j	Used in commands to insert and delete strings
nK, -nK	Delete **n** lines before (-nK) or after (nK) pointer
nL, -nL, 0L	Move pointer **n** lines forward (-nK) or backward (nK)
nMcommand	Execute specified command **n** times
n, -n	Move pointer **n** lines forward or **-n** lines backward
n:	Move to line **n:**
n:command	Execute command (K/L/T) down to but not beyond line n:
Nstring^Z	Search for a string in input file beyond memory buffer
O (Y/N)	Abandon all changes, return to original input file
nP, -nP	Move pointer **n** pages (23 lines) forward or back (-nP)
Q (Y/N/^C)	Exit from ED to MC-DOS without saving any work
R	Read into the buffer a block that you created with X
Rfile^Z	Read into the buffer a file called **file**.LIB
S,s	Used in string substitution commands
nT, -nT, 0T	Display the previous (-nT) or next **n** lines
U, -U	Translate lower case to upper case (-U = toggle off)
V, -V	Toggle line numbering on (V), off (-V)
0V	Display the amount of space remaining in memory buffer
nW	Write first **n** lines of buffer to disk as output file
nXfile^Z	Write **n** lines to the disk as temporary .LIB file
nZ	Wait **n** seconds before executing the next command
^Z or ESC	End of string; separates file names in R commands

ERA

ERA d:**file**
ERA d:**file** [XFCB]

The ERA command erases files that have the RW ("read-write") attribute. ERA can be used to to make disk space or directory space available by deleting unwanted files. You can protect sensitive files from unauthorized access by copying them to floppy diskettes, removing the diskettes to a secure location, and then using the ERA command to delete the files from the hard disk. To delete the file AMBRIAN in user area 8 of drive C:, type:

8B>ERA C:AMBRIAN

The SET command can be used in several ways to protect files from deletion by the ERA command. With the SET command a file can be given the RO ("read only") attribute, or a password can be required to erase a file. For example:

2B>ERA D:FILEX;PASSWORD

This erases the file FILEX with the password PASSWORD from user area 2 of drive D:. See the description of the SET command for more details on the RW and RO attributes and file passwords.

The ERA command accepts wildcard specifications (*,?) in file names. (Great caution is advised in the use of the ERA command with wildcard specifications, since the wrong wildcard specification can destroy valuable files. The ERAQ command is much safer to use than ERA.) If you command the system to delete all of the files in a directory area with the *.* notation, the operating system will ask:

Confirm delete all user files (Y/N)?

Type (Y) or (y) to erase all unprotected files in the directory; any other key returns the system prompt without erasing any files. Additional examples of the use of wildcards with ERA are:

2C>ERA *.BAK

This erases all unprotected files of type .BAK in user area 2 of drive C:.

4A>ERA B:FL??.*

This erases all of the unprotected files with two, three, and four-letter names beginning with "FL" in user area 4 of drive B:.

If the [XFCB] option is specified, then only the extended file control block is erased. An extended file control block is created by the SET command, and contains supplementary information used for password protection of files.

4E>ERA RECORDS.OLD[XFCB]

If protection has been SET to OFF, this erases the password of the file RECORDS.OLD, but **not** the file RECORDS.OLD itself. (If the label of drive E: has been SET with PROTECT = ON, and the XFCB contains a file password, then the ERA program will request that you type the file password before it will erase the XFCB. The description of the SET command in this manual provides information on drive labels, file passwords, and password-protected labels.) See also the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE for DRI's documentation of the ERA and SET commands.

The most common operator errors in MC-DOS involve using ERA with wildcards and unintentionally erasing useful files. By using the safer ERAQ command (see next page) this risk can be substantially reduced.

ERAQ

ERAQ d:**file**
ERAQ d:**file** [XFCB]

The ERAQ command deletes files with the RW ("read-write") attribute. It can erase unwanted files, thereby making directory space or file space available on your disk drives. It is normally used with wildcard (*,?) file specifications.

The ERAQ ("erase with query") command resembles the ERA command, but ERAQ asks for confirmation before each deletion. This feature makes accidental deletion less likely, and lets you delete files selectively within the group of files matching your wildcard specification. ERAQ has the same syntax as ERA, but with dialogue. The following is a ERAQ command and a prompt:

```
OB>ERAQ *.*
B: DOCUMENT.TXT ?
```

Since B:*.* means every RW file in user area 0 of drive B:, ERAQ will ask you for a response for each of the files in this directory area. Type (Y) or (y) to delete DOCUMENT.TXT, or type any other key to keep the file in your directory. ERAQ may propose erasing a file with the RO attribute, but even if you affirm that it should be deleted, ERAQ will print a message that the RO file was not erased. The program will then go on to ask you about the next file, until it has queried you on every file in the directory area that matches your wildcard specifications, or until you exit with a ^C control character.

The SET command can be used to protect files from deleletion by ERAQ (or by ERA), by assigning to files either the RO ("read only") attribute or a file password that prevents deletion. The label of a drive must be set with PROTECT=ON for file passwords to be effective. (The ERA and ERAQ commands are disabled on any logical drive that has been SET with the RO attribute, but this last method is not dependable protection, since a DSKRESET command or ^C restores all drives to RW status.)

The following command would propose for deletion every file whose file extension is .PAT on user area 0 of drive C:.

```
0A>ERAQ C:*.PAT
```

If you affirm that you want to delete a file that is protected by a file password, and if the drive label is set with protection ON, the ERAQ program will ask for the file password, which you must present to delete the file. If you cannot present the file password (which is not echoed on the screen), ERAQ does not delete that file, but queries you about the next file, if any. If a group of files have the same password, you can include the password in the command line, but in this case the file password that you type will appear on your screen:

```
1C>ERAQ A:*.DOC;MYSTERY
```

The above command refers to all the files on user area 1 of drive A: whose extension is .DOC and whose file password is MYSTERY.

The ERAQ command can delete file passwords, but unless protection is set OFF on the logical drive label, you must be able to provide MC-DOS with any file password that you want to delete. The following example with the [XFCB] option:

```
2B>ERAQ DR???.* [XFCB]
```

causes ERAQ to ask you whether to erase the passwords of all the files with two, three, four, and five-letter names beginning with "DR" on user area 2 of drive B:. The files themselves will not be deleted. If the label of drive B: is set with protection ON, the program will demand after each affirmative response that you type the password that you want deleted. Only if you present the correct password will the extended file control block be erased.

See the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE for additional information on the ERAQ command.

FILES

FILES
FILES d:@node
FILES d:file[Gn]
FILES c...

FILES is a program that lets you list your file directory using various formats. It lets you access extensive information about each of the files in any directory on a Gifford system or in a DR Net network. FILES also recognizes more powerful wildcard symbols than other MC-DOS utilities. Displayed below is a list of the options that are available with the FILES program.

-l Long format: give additional information about each file
-r Reverse order: list in reverse alphabetical order
-s Record size: give the size of the file in records (128 bytes)
-t Time order: list in order determined by time stamps
-u Usage time: display last access and modification time

Long Format:

The option lists additional information about each of the files in your directory. Below is a sample command and output:

```
2C>FILES -L TEST.DOC
rsa rwd     1024   Oct 18 12:33 pm   test.doc
 \   \       \          \                \
  1   2       3          4                5
```

The meaning of each of these fields follows:

1) "RSA" stands for "read only," "system," and "archive." Reflects the current status of those respective attributes. If an attribute has not been set for a file, the display reflects this by showing a hyphen ("-") in the corresponding position. For example, if this field shows "---," none of the three attributes have been set.

2) "RWD" stands for "reading," "writing," and "deleting." Reflects the status of the file's password protection. A file may be protected with a password from other users reading, writing, or deleting that file. For example, if this field shows "r--," the file has been protected from being read by other users.

3) This field indicates the size in bytes of the file in question. In our example, the file TEST.DOC is 1024 bytes. If the record size (-s) option had been specified, the number 8 would have been displayed instead of 1024, for that is the size in 128K records of the file TEST.DOC.

4) This time and date is the last time that the file was modified. For instance, it would reflect the last usage time the file was edited by WordStar. If the -u option had been specified, the file's last access time would have been specified. For example, if you used the TYPE command to view the contents of the file, the time that you issued this command would be recorded by the -u option. This field only can display date and timestamps if you have timestamping enabled for that drive. If you have not enabled timestamping for that drive, the time field is displayed as all blanks.

5) This field displays the name of the file. File names are displayed in lower-case lettering. If you are using the FILES program over the network, node names will be displayed in upper-case lettering.

Sometimes the name will be simple, as in:

```
test.doc
```

If the file is in another directory, drive, or node, the name will be complicated, as in:

```
14b:test.doc@SALES
```

The second display is interpreted as the file named TEST.DOC on user area 14, drive B: on the node named SALES.

Wildcards

The FILES program is equipped to recognize a much more extensive array of wildcard characters, than the DIR or SDIR commands. A list of these wildcards and a short definition of each follows:

?	Any single character. (Note: ? has a more restrictive meaning here than the standard "?" wildcard symbol.)
*	Any number of characters.
[ ]	Any of a classs of characters.
[^]	Any but a class of characters.

These characters may appear anywhere, any number of times on a command line. Here are some examples:

FILES ???	All files with three-character file names
FILES A*	All files that start with "A"
FILES *A	All files that end with "A"
FILES A*B*C*	All files that have an "A" somewhere, followed by a "B" somewhere, followed by a "C" somewhere
FILES [^A]*	All files that don't start with "A"
FILES *[ABC]*	All files that contain either an "A," "B," or a "C," anywhere in the name
FILES [A-D]*[J-M]	All files that start with a letter in the range "A-D," and end with a letter in the the range "J-M"
FILES A?C	All three-character files whose first character is "A" and whose last character is "C"
FILES [A-FP-RIZ]???	All four character files whose first character is in the range "A-F" or "P-R" or is "I" or "Z"

The matching patterns described above may be applied to the following three areas:

1) the user area and drive (before the colon)
2) the file name
3) the network node name (if you purchased DR Net)

This means that you could submit a command like the following:

```
2C>FILES [024-7][AC]:*TERM*@*TIO*
```

This command is interpreted as all files on user areas 0, 2, or 4 through 7, on drives A: or C:, and whose file name contains the string "term" anywhere in the name, but only on network nodes whose node name contains the string TIO anywhere in the network node name.

When specifying user areas and multiple drives, as in the example above, some rules must be adhered to.

1) If you are going to look in more than one drive, (for example, [ABC]:) you must specify user area numbers. For example, you could submit a command like this:

   ```
   0A>FILES [024][ACD]:*.DOC
   ```

   but not like this:

   ```
   0A>FILES [ACD]:*.DOC
   ```

You can also use wildcards to specify user area numbers. For example:

?[ABC]: Looks in user areas 0-9 of drives A, B, and C.

??[ABC]: Looks in user areas 10-15 of drives A, B, and C.

*[ABC]: Looks in user areas 0-15 of drives A, B, and C.

So if you want to look for a file or files in user areas 0 through 15, specify the asterisk wildcard and not [0-15]. The FILES program will interpret this to mean that you want to look in user areas 0, 1, and 5.

FIXSTEP

Versions of MC-DOS for S-100 systems with version numbers equal to or greater than 1.30 incorporate the FIXSTEP capability. The FIXSTEP program is used to change the step rate and head load time on floppy disk drives that are connected to a CompuPro Disk 1 or Disk 1A floppy disk controller. Your operating system has been configured with a default step rate of 4 milliseconds, and head load time of 40 milliseconds, to accommodate most floppy disk drives. If your disk drives can step at a faster rate than 4 milliseconds per track, or if they need a slower rate, you can run the FIXSTEP program to change the step rate. The FIXSTEP program patches your MC.DOS system file. You must reboot the computer after running FIXSTEP to implement the new step rate.

Before you invoke the FIXSTEP program, make a new backup copy of the boot diskette. Suppose that your floppy disk drive is called drive F:. Insert the backup copy of your boot diskette into the floppy drive, and type:

```
0F>FIXSTEP
```

The FIXSTEP program will prompt you to supply the name of the system file that you want to patch:

```
Enter filename:
```

Press the Carriage Return key for the default name: MC.DOS. If you have named your system file something other than MC.DOS, enter that name.

You will then be prompted to enter the disk step time in milliseconds:

Enter step time:

This step time can range from 1 to 16 milliseconds. The default step time is 4 milliseconds. The value that you enter here depends on the type of floppy disk drive that your system supports. For most half-height drives, this value should be set to 4. For full-height Qume drives, this value may be shortened to 3, or in some cases, even 2 milliseconds. If you have Shugart SA 800 single-sided drives, set the step rate to 8.

Next you will be prompted to enter the head load time in milliseconds:

Enter head load time:

This number can range from 1 to 254 milliseconds. The default head load time is 40 milliseconds. The number that you enter here depends on the type of floppy disk drive that your system supports. For most drives, this value should be set to 40.

The program gives you one final chance to make sure that you have designated the correct values. If you have no further changes to make, press the RETURN key. The FIXSTEP program will patch the operating system file on your floppy boot diskette to incorporate the new values. These values will not be implemented until you reboot the system. Boot your system with the new boot diskette that contains the patched version of the operating system.

Impaired performance of your floppy disk drives such as verify errors in PIP, can indicate that the step rate has been set to too small a value.

The FIXSTEP program should only be executed by the system manager.

FORMAT

The FORMAT program is used to prepare new diskettes for use. FORMAT will work with 5.25-inch diskettes or 8-inch diskettes and automatically detects and formats single sided or double sided diskettes. A number of formatting options are available for 8-inch diskettes. Gifford Computer Systems recommends that for 8-inch, you use double sided, double density diskettes formatted with 1024-byte sectors. For 5.25-inch diskettes, the FORMAT program only formats in the single sided format of CP/M-86 for the IBM PC, and requires a 5.25-inch floppy disk drive.

When you run the FORMAT program for 8-inch diskettes, the left-hand floppy drive becomes drive A: and the right-hand floppy drive is always drive B:, regardless of whether those designations are normally used by logical drives of your hard disk. If your system has a single floppy drive, the FORMAT program thinks that drive is drive A:.

If you are running MC-DOS in multiuser mode, make sure that no one else is using the diskette drive.

To begin formatting, put the diskette to be formatted in a right-hand floppy disk drive, with the write-protect notch covered by an adhesive tab, and enter:

```
0A>FORMAT
```

This program prompts:

```
CompuPro Disk Sub-System FORMAT Version 3.1
For CP/M-86, MP/M 8-16, and MC-DOS - Modified 7-10-84

Specify Drive (A: - D:, E: - H:) :
```

The options allow for up to eight (8) floppy disk drives on a system. The FORMAT program does not recognize hard disk logical drives. If you have two floppy drives, the left-hand drive is drive A: and the right-hand drive is drive B:. If you have four drives on one Disk 1 controller card, the third and fourth drives become drive C: and drive D:. Drives E: - H: refer to 5.25-inch drives on the Disk 1A controller card. (The FORMAT program is the only program you will receive with MC-DOS that refers to physical rather than logical drives.) If your system has a single floppy drive, enter (A).

Enter the drive in which your diskette is to be formatted. Do not type the colon or a RETURN. The program responds:

Select Disk format mode (0, 1, 2, 3):

0 = 128 2 = 512
1 = 256 3 = 1024

This displays how many bytes (roughly equivalent to keystrokes) that each sector would hold under each format option. Gifford Computer Systems recommends that you use 1024-byte sectors since this gives you the most data storage. Software from commercial suppliers on 8-inch diskettes is usually distributed formatted with 128-byte sectors (option 0).

If you are formatting a diskette in the right-hand drive, the program responds:

Confirm ready for format on disk drive B (y).

Insert the diskette into the drive and press (Y). Do not press RETURN. The FORMAT program formats all tracks, and verifies that they were formatted correctly. The program shows its progress during the format and verification phases. The program displays an F for each track it formats and a V for each track it verifies. If S, C, or E appear instead of V, reformat the diskette. If it occurs again on the diskette, this indicates the diskette is bad and should be discarded.

If the FORMAT program consistently gives S errors when formatting, it is likely that the disk drive needs repair, and you should contact your dealer. Similarly, if you have two floppy drives, and the same diskette shows errors when formatted on one drive, but not on another, or if one drive cannot read diskettes formatted by the other, you should call your dealer to discuss possible disk drive problems. Do not call your dealer if you cannot format diskettes with exposed write-protect notches, since your system is designed to respect that protection.

Upon completion of formatting, the program prompts:

```
FORMAT complete.
Do you want to format another disk?
R  - request new parameters.
cr - duplicate last FORMAT.
N  - exit back to system.

>
```

Type your response after the prompt.

If you select the R option, the program requests new parameters, allowing you to reformat your floppy diskette with new specifications. If you press RETURN, the program will ask you to confirm that a fresh diskette is ready for formatting. After you type (y), FORMAT will format the diskette.

Remember that the FORMAT program obliterates any information that existed on a diskette prior to the FORMAT operation. If you make a mistake and format an important diskette, whatever files were on it cannot be recovered after FORMAT has run.

GENCCPM
GENCCPM <inputfile >outputfile

The GENCCPM command is used to generate the MC.DOS system file. It is very unlikely that you will ever need to run the GENCCPM command to make a new version of the MC.DOS file.

The GENCCPM command should be run on a disk with all of the files supplied on the system diskette, especially all that have file types of .CON and .RSP. We recommend that you use a copy of your original boot diskette.

GENCCPM Input and Output Files

The GENCCPM program will accept an input file which contains all of the parameters which must be specified and changes which should occur in the new MC.DOS file. Input files can be created in the non-document mode of WordStar or any word processing program. Gifford supplies you with a series of input files on your MULTIUSER CONCURRENT DOS XIOS DISK. You can send console output to an output file.

Each command in an input file should appear in the same order in which you entered it in the GENCCPM menu program. Every command should appear on a separate line followed by a RETURN. The last command in the input file should be:

GENSYS

To access the GENCCPM menu program type:

0A>GENCCPM

and the program's menu will be displayed on your screen.

The short summary below lists the steps in generating a new operating system. Refer to the manual entitled CUSTOMIZING YOUR MULTIUSER CONCURRENT DOS SYSTEM for a more detailed explanation of this procedure.

1) Choose an XIOS file based on the number of physical and virtual terminals, the number of printers, and the number and type of hard disk drive(s). Or use the same XIOS file that is indicated on the label of your master system diskette (e.g. X1070702.F21). See pages 3-11 of the "CUSTOMIZATION" manual.

2) Rename your new XIOS file to XIOS.CON. See page 12 of the "CUSTOMIZATION" manual.

3) Choose the correct input file for GENCCPM depending on the cache size and type, the number and size of the hard disk(s), whether the floppy or the hard disk drive should be designated as drive A: (e.g. HRD64K-1.F21). See page 13 of the "CUSTOMIZATION" manual.

3a) If you are adding an RSP into the operating system that was not previously included, you must edit the input file chosen in step #3 and add the new RSP name (e.g. IN=KRON.@CRSP).

4) Run GENCCPM specifying the input file from step #3 (e.g. GENCCPM <HRD64K-1.F21). Note the "<" symbol must be included in the command line. This generates a new system file called CCPM.SYS.

5) Rename the new CCPM.SYS file to MC.DOS.

Additional steps if the system will be connected via network controller to other systems:

6) Run ADDNET specifying the MC.DOS file to use (e.g. ADDNET MC.DOS). This creates a new system file called CCPM.SYS.

7) Rename the resulting CCPM.SYS file to MC.DOS.

If the boot loader needs to be changed because of a change in the hard disk size, or if the hard disk has been reformatted:

8) Run SYSGEN to put the boot loader on the hard or floppy disk drive specifying the boot loader (e.g. BOOTH.LDR), and loader file name (e.g. MC44.LDR), and the destination drive name (e.g. A:, F:, etc).

See Gifford's manual CUSTOMIZING YOUR MULTIUSER CONCURRENT DOS SYSTEM for additional information on using this program.

GENCMD file

The GENCMD creates a .CMD command file from a .H86 file (similar to files created with the ASM86 command). GENCMD is a programmer's utility not required in ordinary applications.

The GENCMD command converts hexadecimal format (.H86) files to executable (.CMD) files. Hexadecimal files are usually created by the ASM86 assembler language translator. For example:

```
0A>GENCMD ECHO
```

takes the file ECHO.H86 and creates the file ECHO.CMD.

See the CONCURRENT CP/M OPERATING SYSTEM PROGRAMMER'S GUIDE for additional information.

HELP

HELP topic subtopic
HELP topic ALL

The HELP command supplies information about the system, or about programs supported by the system. You can also make your own HELP files (files with extension .HLP that HELP can display), or modify the HELP files supplied by Gifford Computer Systems.

Gifford Computer Systems supplies a MC-DOS HELP file called MCDOS.HLP, describing commands and files in MC-DOS. Some other programs also have their own HELP files. To get help, type HELP, the name of a .HLP file subject (with no .HLP extension), and the subtopic of the subject on which you need help. For example, to find out about the TYPE command in MC-DOS, give the command:

```
2C>HELP MCDOS TYPE
```

To get **general** information on a subject, type HELP and the HELP file name. The HELP program checks the .HLP file that you specify for the subject of GENERAL. For a list of the subtopics in a HELP file, enter "ALL" as the subtopic. The following example lists a HELP file called PASSWD.HLP, with three subtopics "GENERAL," "NEWUSER," and "PASSWORD:"

```
?general
    The PASSWD file defines the encrypted login password,
initial default user number, drive, printer, optional application
program, terminal access, and user area access for every valid
account name on a Gifford MC-DOS system.
?newuser
    The NEWUSER utility must be used by the system manager to
create and maintain the PASSWD file.  Only the "system" account
can use the NEWUSER program.
?password
    The PASSWORD program permits users to change their own
encrypted login passwords in the PASSWD file.
```

Users can get help on PASSWD by typing any of the following commands:

```
0A>HELP PASSWD

The PASSWD file defines the encrypted login password,
initial default user number, drive, printer, optional application
program, terminal access, and user area access for every valid
account name on a Gifford MC-DOS system.
```

In the example above, the HELP program displays the message under the subject "GENERAL." Alternatively, you can type:

```
0A>HELP PASSWD ALL

general     newuser     password
```

This displays a list of subtopics in the PASSWD.HLP file. To get information on a specific subtopic of the PASSWD.HLP file, such as the subtopic NEWUSER, type the command:

```
0A>HELP PASSWD NEWUSER

The NEWUSER utility must be used by the system manager to
create and maintain the PASSWD file.  Only the "system" account
can use the NEWUSER program.
```

Users who take the trouble to learn how to ask the HELP command to display information from .HLP files, and how to use the ALL option to list subtopics, will find that they can use more of the features of MC-DOS and of other software for which HELP files are available. HELP files are more convenient for most users than printed documentation as a source of information, and can reduce the need for users to refer questions directly to the system manager or to customer service engineers.

Creating Customized HELP Files

HELP files are files with extension .HLP that can be displayed by the HELP command. A HELP file can be created or modified with a text editor like ED or WordStar in non-document mode. To add a subtopic to a HELP file, you must enter the name of the subtopic on a separate line as a single word preceded by a question (?) mark. The text to be displayed follows, beginning on the next line. If the text of a subtopic occupies more than 23 lines, type .PAGE (beginning with the period in the first column) to indicate the bottom of a screen of text. The HELP program will prompt the user to press RETURN for additional text. If you do not type ".PAGE" in this way, long texts will scroll off the top of the screen before users can read your advice.

To be accessible from every user area and logical drive, all .HLP files must be SET with DIR and RO attributes, and must be kept in user area 0 or drive A:. If it is necessary to modify or add text to a HELP file, the SET command must be used to temporarily assign the RW attribute to the .HLP file so that it can be modified by a text editor.

Each HELP file should have the subject of ?GENERAL in it, so that users can get general information about the program by simply typing "HELP" and the name of the subject.

Each HELP file has an index file associated with it, whose file type must be .NDX. To create the index file, give the HELPNDX command with the file name. For example, if you had just created or modified the example HELP file called PASSWD.HLP, you must give the command:

```
0A>HELPNDX PASSWD
```

Unless you run HELPNDX to update the index every time you modify a HELP file, the HELP command cannot access the modified text. Maintenance of HELP files falls under the jurisdiction of the system manager. By supplementing Gifford's MC-DOS HELP files with custom files reflecting the current needs of users, the system manager can automate some facets of on-site support.

HELPNDX file file file ...

The HELPNDX command updates the index to the HELP files after a HELP file has been created or modified. The HELPNDX program must be run if any .HLP file has been modified or if a new .HLP file is created. If you do not run the HELPNDX program after changing your HELP files, the HELP command will not function properly with the new or modified .HLP file.

To use the HELPNDX program, your current directory area must be user area 0 of drive A:, and the .HLP files must be in the same directory area. Type the keyword "HELPNDX," followed by the name (or list of names) of any HELP files that have been created or modified since the last time HELPNDX was invoked, without typing the file extension .HLP of the new HELP file. For example, suppose that a HELP file called PASSWD.HLP had been modified by adding new subtopics. The system manager must type:

```
0A>HELPNDX PASSWD
```

This creates a new index file called PASSWD.NDX. Once an index file is created or updated by the HELPNDX command, the HELPNDX command need not be used again until a .HLP file is created or modified by ED or by WordStar (in non-document mode).

All HELP files must have the extension .HLP, and must be SET with the DIR and RO attributes in user area 0 of drive A:. A HELP file contains the text that the HELP command displays on screen, separated into subtopics. Each subtopic has a single word for a heading, followed by a text describing that subtopic. The heading must begin with a question (?) mark. When a user invokes the HELP command, the HELP program uses the index files created by HELPNDX to find the specified subtopic within a HELP file and to display on screen the text for that subtopic. Use of the HELPNDX command falls within the jurisdiction of the system manager. It is discussed again in **Chapter 3, Section 3.7.**

HISTORY

The HISTORY file records and timestamps all commands given to the system. HISTORY is an optional file. If the file does not exist, then no historical records are maintained. If there is reason to think that unauthorized use of a system is being attempted, the HISTORY feature of MC-DOS may assist the system manager in identifying and correcting the problem.

Because the operating system writes every command line to the HISTORY file (if it exists), implementing HISTORY somewhat reduces the speed at which the system operates. The effect on system performance of a HISTORY file varies with the number of users, and with how they are using the system. To implement the HISTORY feature, give the command:

```
0A>MAKE HISTORY
```

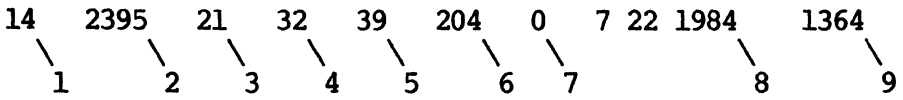
This will create an empty file in user area 0 of drive A: to which the system can add information as it is generated. Only commands given by users who log in to the system after you give the command above will be recorded. Once in use, the contents of HISTORY will resemble the example that follows:

```
tty6   0mary 14 23952132392040 7221984   1364 pip a:=e:data.fyl[vrw]
```

The first three fields (in this example "tty6 0mary") are the **name** (from TTYS file) of the physical terminal or modem, and the **relative terminal number** of the virtual terminal that issued the command. In MC-DOS a physical terminal can support up to four "screens," corresponding to virtual terminals, numbered 0, 1, 2, and 3. The next field is the **account name** by which the operator logged in to the virtual terminal. The last field lists the actual **command line** entered from the keyboard.

The Meaning of HISTORY

In the same example of a line from HISTORY (but with blanks inserted for greater clarity) the other numerical fields are:



1. **Virtual Terminal number**
2. **Days** since January 1st, 1978
3. **Hour** of the day (24 hour clock)
4. **Minute** of the hour
5. **Second** of the minute
6. **Julian day** of the year (1-366)
7. **Day** of the week (Sunday = 0, Monday = 1, Tuesday = 2, etc.)
8. **Date** in month-day-year format
9. **Program file link number,** a serial number showing how many commands had been recorded in the HISTORY file when this command was recorded. The link number of the last command is also written in the first line of the HISTORY file.

The complete HISTORY line is reproduced below.

tty6 0mary 14 23952132392040 7221984 1364 pip a:=e:data.fyl[vrw]

The meaning of this entry in the HISTORY file is:

"From virtual terminal 14 (the first screen of physical terminal 6) someone who logged in as "mary" gave a PIP command to copy and verify a file called DATA.FYL to drive A: from drive E: at thirty-nine seconds after 9:32 pm on Sunday, July 22, 1984, the 1,364th command recorded since the beginning of HISTORY."

If you erase the HISTORY file, command lines are no longer recorded by the operating system. The creation and maintenance of HISTORY falls under the jurisdiction of the system manager.

INITDIR

INITDIR d:

The INITDIR command modifies the directory of a logical drive to support timestamping of files. The SET command, not INITDIR, activates date and timestamping, but SET cannot do so until an INITDIR command has reformatted the directory. The INITDIR procedure is necessary because time and datestamping occupy extra space in a directory. INITDIR only reformats the directory, not the data area where files are stored on a disk. The INITDIR command can be issued either from the drive whose directory you intend to reformat or from a different drive. All the user areas of a logical drive are affected by INITDIR. No other users can be active on the system when INITDIR is invoked. To reformat the directory of logical drive B:, for example, type:

0A>INITDIR B:

The program will ask you to confirm the command:

INITDIR WILL ACTIVATE TIME STAMPS FOR SPECIFIED DRIVE
Do you want to re-format the directory on drive B:(Y/N)?

Type (Y) to modify the drive B: directory, or any other key to exit without reformatting. If any files are in use on that drive, INITDIR will not work, and you will get an error message. To avoid this potential problem, the system manager must run INITDIR before any other users have logged in to the system.

If the disk has already been reformatted for time and datestamping, the INITDIR program displays the message:

Directory already reformatted
Do you want to recover time/date space (Y/N)?

Type (Y) to eliminate time and datestamping. If you type (N), timestamping remains active on your disk, but INITDIR will ask you if you want to remove the current time and date stamps:

Do you want the existing time/date stamps cleared (Y/N)?

Press (Y) to clear, or (N) to keep the existing stamps.

If the disk label is password protected, the INITDIR program will display the message:

Directory is password protected
Password, please. >

If you do not enter the correct password, INITDIR will terminate without modifying the directory.

A directory of files with date and timestamping requires more space on the logical drive than a directory of files without date and timestamping, and fewer entries in the part of the logical drive that holds the directory of files. You must have at least 25% of the total directory space available to accommodate the memory requirements of date and timestamping. If there is not enough space on the disk directory when you type the INITDIR command, the INITDIR command displays the error message:

ERROR: Not enough room in directory.

A logical drive whose directory has been modified by INITDIR remains reformatted, even if the system is powered down or the RESET button is pressed. That is, INITDIR is a "one time" command that need not be repeated unless you decide to eliminate timestamping or extend timestamping to another logical drive.

See the description of the SET command and the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE for additional information.

KEYS

KEYS file

The KEYS program makes it convenient to program the special function keys of the TeleVideo 950, the Freedom 200, the GCS-80, the WY-50, and any other terminal whose function keys use TeleVideo 950 protocols. The KEYS program uses special template files that you can create with any ASCII text editor, such as ED or WordStar (in non-document mode). You can create as many different template files as you wish, and so you can develop a whole library of function key definitions that are optimized to suit various projects. To assign a template file to the function keys of your physical terminal, you invoke the KEYS program by entering a command of the following form:

0A>KEYS template

Here **template** is the name of a file whose file type is .KEY. All of the virtual terminals supported by your physical terminal will use the key definitions provided by the template file. For example, to use a template file called SAMPLE.KEY to program your terminal's function keys, you must type:

0A>KEYS SAMPLE

Template files have a simple structure. A template file consists of a series of lines, each one dedicated to a specified function key. The first character in the line, called the **key symbol**, represents the key to be programmed. The remaining characters, the text, represent the characters which the associated key will send whenever it is pressed. These template files must have a file name of no more than eight (8) characters, and must have .KEY as the file extension. For example, a file of key assignments for SuperCalc might be named SC.KEY, while a file of key assignments for dBASE II might be called DBASE.KEY.

To assign the definitions contained in a file called DBASE.KEY to programmable keys, type:

```
0A>KEYS DBASE
```

To reassign these keys according to a file called SC.KEY, type:

```
0A>KEYS SC
```

If KEYS.CMD is assigned SYS and RO status by the command:

```
0A>SET KEYS.CMD[SYS,RO]
```

then a copy of KEYS.CMD in user area 0 can be invoked by all user areas of a logical drive.

To create a template file, use WordStar to create a file in the non-document mode (i.e., from the "OPENING MENU" enter (N) rather than (D)). Give your file any name that you wish, but make sure that it has the .KEY extension. For example, you could call a file TEST.KEY, but do not call it TEST.TRY.

Your file can have as many lines as you have function keys to program. For the TeleVideo 950 and the Freedom 200, a single key can send one message unshifted and a different message shifted, so in effect you have twice as many logical function keys as physical ones. The first character in each line must contain the symbol for the key that you wish to program. The actual symbol used to specify a function key is not dictated by the KEYS program, but is dependent on the terminal itself and should be defined in the user's manual for that terminal. The rest of the line can contain characters representing the data to be transmitted by the specified key.

The characters which each function key is to send are not stored in the main memory of your computer, but in the internal memory of the terminal itself. Since this internal storage capacity is in all cases quite limited, you are restricted as to the number of characters you can assign to each function key and to each terminal. For example, the TeleVideo 950 allows up to 63 characters per function key and up to 256 characters total. See Chapter 3 of Gifford's VIRTUAL TERMINALS USER'S MANUAL for a list of these special function keys.

In counting programmed characters to determine whether they exceed the maximum number permitted on a given terminal, the following rule applies: A carriage return counts as a character, so any line in your definition file that does not terminate with a caret (^) should have no more than twenty-six (26) data characters for a GCS-80, 255 characters for a Freedom 200, and 63 for a TeleVideo 950. The caret (^) itself does not count as a character.

See Chapter 3 of Gifford's VIRTUAL TERMINALS USER'S MANUAL for additional information.

LOGIN

LOGIN.CMD is an optional program on an account's initial default drive and user area, which will run before the login process is completed. This program file can be used to customize your system by executing a program which you want to be part of an individual account's login procedure. LOGIN.CMD has limited application on most systems. The LOGIN.SUB file will have greater value to most users, since it can execute multiple commands.

You can rename any command file "LOGIN.CMD" with a PIP command. For example, suppose that you wanted a LOGIN.CMD that would automatically check the appointment calendar of any account whose initial default directory in the PASSWD file was user area 12 of drive B:. This could be done by making AP the LOGIN.CMD, and creating a copy of the program in that directory area with the following command:

```
0A>PIP B:LOGIN.CMD[G12]=AP.CMD[VROW]
```

Note that this creates multiple copies of command files, since each initial default area requires its own LOGIN.CMD. If you wanted **every** account to automatically check for appointments, the USERINIT.CMD feature is more efficient in the use of disk space. The LOGIN.CMD and LOGIN.SUB files are appropriate where special initialization procedures are needed for some (but not all) accounts, in contrast to USERINIT.CMD and USERINIT.SUB, which are designed to initialize all accounts in the same way.

The maintenance of this file falls under the jurisdiction of the system manager. It is discussed again in **Chapter 3.**

LOGIN.SUB

LOGIN.SUB is an optional file on an account's initial default drive and user area which contains commands to be executed before the login process is completed. The following example shows you how you can use LOGIN.SUB to check your appointment calendar and electronic mailbox, reload the special function keys, and dislpay a screen of advice before the login process is completed. All of the files mentioned must exist:

```
; Sample LOGIN.SUB file to check your AP schedule
AP
; Display all MAIL entries in your mailbox
MAIL
; Load function keys for application
KEYS FR200
; Display MENU
Type ADVICE.APV
```

The description of the SUBMIT command in this chapter provides detail on the creation and use of .SUB files for multiple commands. Note that the LOGIN.SUB file affects only those accounts whom the PASSWD file assigns to an initial default directory where a LOGIN.SUB file exists, while USERINIT.SUB affects all accounts at login. For that reason, USERINIT.SUB should be used for initialization routines intended for everyone, while LOGIN.SUB files should be created for individual accounts that would benefit from special treatment at login.

The maintenance of this file falls under the jurisdiction of the system manager. Since each LOGIN.SUB file only affects a single directory area, there is not much risk in allowing users to experiment with this feature if they understand how it works. The system manager should read about SYSINIT.SUB, USERINIT.SUB, and the remaining customization files in this chapter for ideas on how to use the customization features of MC-DOS.

LOGOUT

The LOGOUT command is given to log an account and its current virtual terminal out of MC-DOS. After the LOGOUT command is given, the computer displays the banner and the **Name:** prompt. When a LOGOUT command is given, the system updates the UTMP and WTMP files. The LOGOUT command is given by typing its keyword:

```
0A> LOGOUT
```

The system responds by logging out the account and by displaying the banner and the **Name:** prompt on the screen.

Failure to use the LOGOUT command will reduce the usefulness of the time accounting utilities and the WHO command, and can compromise system security by allowing unauthorized persons to access the system via an unattended but logged-in terminal. If you have a workstation that supports multiple virtual terminals, be sure that you log out from every virtual terminal when you are finished using the system. The keyboard of a terminal that is logged out can still switch screens to other virtual terminals.

The status line of most terminals can be programmed to show if background terminals are still logged in, as illustrated in the sample VTTYS file on **page 3.3-9** of this manual. All should be logged out when you finish your work. To make it easier to give the LOGOUT command, the KEYS command can be used to program a function key to type "LOGOUT" in a single keystroke.

If the AUTOST file is not on user area 0 of drive A:, the system will not go into multiuser mode until the user at the system console gives the LOGOUT command.

The LOGOUT command does not detach a printer that has been attached to a virtual terminal by a ^P control character. LOGOUT is not an ordinary command file, but is part of the MC-DOS shell, so it cannot be erased nor assigned attributes by SET or CHSET.

LOGOUT.SUB

LOGOUT.SUB is an optional file in an account's current drive and user area which contains commands to be executed before the logout process is completed. Read about LOGIN.SUB, SYSINIT.SUB, USERINIT.SUB and the remaining SUBMIT files for ideas on how to use the customization features of MC-DOS.

The following example file shows how you can use LOGOUT.SUB to view your appointment calendar and your mailbox before you log out of the system:

```
;Sample LOGOUT.SUB file to check AP and MAIL for messages.
AP
MAIL
```

To create or modify a submit file like this, use a text editor like ED or WordStar (in non-document mode). If a user logs out from a directory area in which no LOGOUT.SUB file exists, no LOGOUT.SUB commands will be executed.

The maintenance of this file falls under the jurisdiction of the system manager. It is discussed again in **Chapter 3.**

LPRS

The LPRS file describes the characteristics of each printer attached to the system. It has the same format as the TTYS file.

An example of the file is:

```
0:8:1:82n:0,lpr0:System printer, Texas Instruments
1:8:2:81n:1,h-p:H-P graphics plotter
```

The file has one entry for each printer or plotter on the system, and the fields are defined as follows:

```
DEV #:BAUD RATE:HANDSHAKING:PROTOCOL:LPR NAME:COMMENT
  \        \          \          \        \       \
   1        2          3          4        5       6
```

1. **Number of the device,** as printed on the back panel. Printer 0 is the default printer for users, unless the PASSWD file specifies another printer or plotter. If you have a parallel printer, there is no need to designate this is in the LPRS file, as this designation is hardware dependent.

2. **Printer baud rate.** The default is 9600. Baud rates codes are the following:

   0 = 9600
   1 = 110
   2 = 300
   3 = 600
   4 = 1200
   5 = 1800
   6 = 2400
   7 = 4800
   8 = 9600
   9 = 19200

3. **Optional RS-232 hardware handshaking.** The default is 0. If handshaking is necessary, enter the number corresponding to the type of handshaking needed from the following:

   0 = No handshaking
   1 = Hardware handshaking (DTR) (RS-232 pin-20)
   2 = Software handshaking (XON/XOFF)
   3 = Reserved for custom application

4. **Optional communications protocol.** The default protocol is 8 data bits, 2 stop bits, and no parity, which is the setting for most printers. For example, if your printer requires a protocol of 7 data bits, 2 stop bits, even parity, enter 72E in field 4.

Data bits	Stop bits	Parity
8	2	N (none)
7	1	O (odd)
		E (even)

5. **Printer name.** The names given here are used in the MC-DOS PRINTER command, which changes the printer. In the example file on the preceding page, to access the printer on port 1, give either one of the following two commands:

```
2B>PRINTER 1
2B>PRINTER H-P
```

6. **Optional comment field.**

This allows up to 100 characters to provide additional information about a printer or plotter.

Check the printer manufacturer's manual for printer baud rate, hardware handshaking, and communications protocol.

The LPRS file can be modified while the system is running in multiuser mode. Once the file has been modified, type:

```
0A>RESET
```

This will reset the system to the new specifications without the need to shut down the system.

The number of printers and plotters that can be supported by an MC-DOS system is specified in the XIOS (extended input/output system). If a system is reconfigured to support a greater number of printers and plotters than was known to Gifford when the system was ordered, it may be necessary to install a larger XIOS. Read the Gifford publication CUSTOMIZING YOUR MULTIUSER CONCURRENT DOS SYSTEM for details on how to modify the XIOS.

Please Note: If you have a parallel printer attached to your system, there is no need to designate this in the LPRS file, as this designation is hardware dependent.

The maintenance of this file falls under the jurisdiction of the system manager. It is discussed again in **Chapter 3.**

MAIL

MAIL name
MAIL user1 user2 user3
MAIL ALL
MAIL <file NAMELIST namefile
MAIL -C
MAIL -R -A -V

The MAIL utility allows a user at a terminal to send messages to other users attached to the network running MC-DOS. The message is sent to a "mailbox" where the "addressee" reads and deletes the message. This mailbox resides in user area 15, drive A: of the system node. If the system node is down, mail service is unavailable for the entire network. For the MAIL program to work, the NETUSERS and UTMP files must be on the system.

Mail messages can be either a file created by a text editor such as WordStar or a message typed directly to a "mailbox" by the MAIL program.

The syntax that you must use for sending, receiving, and checking for mail follows:

Sending mail: MAIL <file NAMELIST namefile ALL user...

Receiving mail: MAIL -R -A

Checking mail: MAIL -C

A definition of each of the command arguments listed above follows:

<file This is the file name of a file containing a message. The < symbol must precede the file name. This file may be created with any text editor.

NAMELIST This word must be typed on the command line before you type the name of the file that contains a list of account names to which you want to send your message.

namefile This is the file that contains, on separate lines, the account names to which you want to send your message.

ALL Mail can be sent to all accounts by using this command argument.

-R (Reverse messages) By typing MAIL -R, you will see a list of all of the messages in your mailbox in reverse order, where the most recent message will be positioned at the top of the list. This enables you to read the most recent message first if you scroll through the messages by pressing RETURN.

-A (All messages) By typing MAIL -A, you will see a list of all of the subject headers for the messages that you have been sent, plus all of your mail will be displayed without prompting. This list will have the oldest message positioned at the top of the list, unless you type MAIL -A -R, which will reverse the chronological order.

-C (Check for mail) Typing MAIL -C enables you to check if you have any mail without having to log out and then log back in to the system. Users who frequently receive mail while they are logged in may want to include a "MAIL-C" line in their LOGOUT.SUB file to see if new messages have arrived.

-V (Display version number) Typing MAIL -V displays the version number and copyright message of the MAIL program as well as a list of all of the subject headers for the messages that you have been sent.

Sending Mail

To send a message to a user on your node or to a user on any other node that is currently attached to the network, type:

```
2A>MAIL LINDA
```

where LINDA is the account to whom you want to send the message. The names used by the MAIL program must correspond exactly to the account entries in the PASSWD file. In a DR Net network, the account to whom the mail is sent must appear in the NETUSERS file.

The MAIL utility now prompts you with:

Subject:

You should respond to this prompt by typing a brief description of the subject of the mail that you are about to send. Once you have entered this information, the MAIL program will prompt:

>

You can enter any text you want on this line, and on subsequent lines. If you have WordStar on your system, you can enter directly into a WordStar file, thereby having the advantage of using WordStar text editing commands, by typing the following:

> ~E

By typing the command above, you are now writing your MAIL message from inside WordStar. You can save your file and exit using the normal WordStar command procedures, which will take you back to the MAIL program prompt. (If you are using a text editor other than WordStar and you want to use the ~E option, rename a copy of your text editor program to WS.CMD. For example, if you are using Newword, make a copy of the file NW.CMD and name the copy WS.CMD, because the MAIL program will be looking for the WS.CMD file. This will enable you to use the ~E option to invoke a WordStar file into which you can save your MAIL message.) You can now exit from the MAIL program by typing a single period (.) on a blank line. MAIL then sends your message to the user(s) that you indicated on your command line. If you want to cancel a message, type:

>~X

If a portion of the text of your message exists in a pre-existing file, you can read this file into the text of your message by typing:

```
>~R [filename]
```

The file which you specified will be incorporated into the text of your message. The text of the file which was read into your message will not appear on your screen. However, if you would like to read your message prior to sending it, type the following:

```
>~P
```

and the text of your message will be displayed on your screen.

You can send the same message to more than one account by adding the account names to the end of the command line. For example, to send a message to acounts named "chris" and "robin," type the command:

```
2A>MAIL CHRIS ROBIN
```

You can also create a file in WordStar, non-document mode, that contains a list of all of the people to whom you want to send a message. Suppose you want to send a message called ANNUAL.RPT to everyone in the Sales Department. Create a file

called SALES.LST and list, on separate lines, the account names of everyone in the Sales Department. Now type:

```
2A>MAIL <ANNUAL.RPT NAMELIST SALES.LST
```

and your ANNUAL.RPT file will be sent to everyone in the Sales Department who was listed in your file called SALES.LST. You must type the word "NAMELIST" after your filename for this feature to work. The "<" in the command above tells the MAIL program that the attached word is the name of a MAIL message file.

Reading Mail

To read mail, type:

```
2A>MAIL
```

If you have mail, you will be shown a numbered list of all of the MAIL message subjects headers that are in your mailbox, as well as a sum total of all of the messages that you have been sent, from whom each message was sent, and the time and day that each message was sent. This list might look something like the following:

```
Copyright (C) 1984 Gifford Computer Systems
Gifford mail version 2.1 Type '?' for help.

3 messages
1 john   Wed  Jun 11, 1984 11:09 am  "new price list"
2 eric   Thu  Jun 12, 1984  8:55 pm  "racquet ball"
3 paula  Fri  Jun 13, 1984 12:34 pm  "benefits meeting"
```

Each of these messages may be read by typing the number that corresponds to the appropriate letter. Suppose that you want to read Eric's message first. Type (2). Now if you want to read Paula's message (3), press the RETURN key, and the MAIL program will take you from message #2 to message #3.

After you read a message, you can tell the MAIL utility what to do with it (such as save it in a file, or forward it to another account, or simply delete it). Displayed below is a list of options that the MAIL program recognizes:

D	**Delete current message**
~E	**Enter WordStar**
F	**Forward the current message to someone**
H	**Print the headers of all pending messages in mailbox**
M	**Mail a message to someone**
~P	**Prints your message so far**
Q	**Quit, remove deleted messages permanently**
R	**Reply to sender of current message**
~R fn	**Reads a file into your current message**
S fn	**Save current message to a file**
U	**Displays a list of users attached to the network**
X	**Exit from MAIL, but do not delete marked messages from mailbox**
~X	**Cancels the current message that you're writing**
~?	**Prints the Mail program's help menu**
.	**Display the current message**
+	**Display the next message**
-	**Display the previous message**

Listed below are definitions for some of the options listed above:

D This will delete the current message, and type the next message. The current message is only removed from your mailbox when you exit from the MAIL utility by typing (Q).

R This option is used to reply to a MAIL message after you have read it. By typing (R), you can instantly respond to the current message, and that response will automatically be sent to the sender of the original message. After you type (.) to end your response, the following message will be displayed on your screen:

Sending reply.

F This will forward a message to someone. If you read a message and decide that it would be useful for someone else on the network to read that message, type (F) and then the name of the account or the namefile containing a list of account names to whom you want to forward that message.

Suppose that you logged in to your system as Gail, and you read a message with the subject header "test" that was sent from Kent and that you want to forward it to Nancy. To do this, type:

```
F NANCY
```

When Nancy logs in to her system and types MAIL, the following line will appear on her screen, indicating that Kent sent a message to Gail who has forwarded it to her.

1 gail<-kent Wed Jun 13, 1984 8:48 pm "test"

M This option is used to send mail to other users. It is the equivalent of typing MAIL.

H This will print out a list of all of the subject headers of all pending messages in a mailbox. These messages are listed with the oldest subject header positioned at the top of the list.

S fn The "S" option, followed by a file name will save the designated message to a file. This is useful if the message that you are reading contains lengthy text that you want to save for future reference. These files include the postmark, which indicates the sender, the time and date sent, and the account(s) to whom the message was sent. You can specify many files at once. If you do not give a file name, the message is written into the file MBOX.

+ This will display the next message, and leave the current message in your mailbox.

- This will display the previous message and leave the current message in your mailbox.

Q This option will exit you from your session with the MAIL program. Any messages that you have not read, or ones which you did not delete, are left in your mailbox.

X Enables you to exit without modifying the mailbox. With this option, any messages that you marked for deletion are left in your mailbox. All messages that you have forwarded to other accounts, or saved in files, are sent and saved.

Additional Notes on Mail with Networking

Every account that you list on the command line must be listed in the NETUSERS file. If you give a name of an account that is not listed in the NETUSERS file, MAIL gives you an error message, and puts the message that you sent into a file called DEADMAIL. If you gave both valid and invalid names, MAIL sends the message to the valid accounts.

When you send a message, the MAIL utility puts a postmark (the current time, date, and location from which you sent the mail) at the beginning of the message. If the system file UTMP does not exist, MAIL cannot determine who you are or from where you are sending the mail; your message is marked as being sent from "UNKNOWN."

Mail is kept on user area 15 of drive A: of the system node. The "mailboxes" are files with the account's login name for the file name, and .MBX for the file type. Mail files continue to grow unless mail is deleted. The system manager should periodically check the files on user area 15 of drive A: to be sure that none are wasting disk space.

If the same person tries to send mail from two different terminals simultaneously, the result is unpredictable. Also, if two users send mail to the same account at the same time, the result is unpredictable.

The MAIL utility allows you to send mail to and read mail from user area 15 of drive A: even if your account in the PASSWD file does not give you permission to access user area 15.

MAKE

MAKE file

The MAKE command creates an empty file on your disk. This is useful if you need to make a file that does not contain any information such as AUTOST or a file such as WTMP or HISTORY to which information will be added later. To make a file, give the MAKE command and the file name at the system prompt. For example:

```
0A>MAKE NWTMP

0A>MAKE NAUTOST

0A>MAKE NTEST.FIL
```

In all of these examples, an empty file of the name specified after the keyword "MAKE" is created in the directory of drive A:, in user area 0.

MOTD

MOTD is the message of the day file. When a user logs in, the message of the day is displayed. The file name for the message of the day is MOTD and the file is kept on user 0 of drive A:. Use this file for messages that you want all users to see when they log into the system. This file can contain anything you wish, including escape sequences to control the user's screen. Create and update the MOTD file with an editor such as ED or WordStar (in non-document mode). This file is discussed again in **Chapter 3.**

NAMENODE
NAMENODE R

The NAMENODE program is used only in DR Net local area networks. The NAMENODE program is used to modify the list of node names that are attached to the network. This list of node names is contained in a file called NAMSVR.DAT which is located in user area 0, drive A: of the system node. The NAMSVR.DAT file is automatically updated each time the NAMENODE program is run. The NAMENODE program is part of the SYSINIT.SUB file, so as soon as your system starts up, your node name will be added to the NAMSVR.DAT file. This file should only be modified by the system manager.

The NAMENODE program is run automatically by the local system when that system is booted. This program logs in to the system node and writes the node ID number and node name to a communication queue. The "receive" option is invoked by typing:

```
0A>NAMENODE R
```

from the system node. The NAMENODE program will detach and hang in the background to read the communication queue for the node ID number and node name to update the NAMSVR.DAT file. The NAMENODE scanning option from the system node scans the communication queue periodically to maintain the NAMSVR.DAT file.

To display a list of the existing node names and their corresponding ID numbers that are attached to the network, use the NET NAMES command.

See the DR NET USER'S MANUAL for additional information about the NAMENODE program.

NAMSVR.DAT

The NAMSVR.DAT file is only used in MC-DOS systems connected to DR Net local area networks. NAMSVR.DAT is a binary data file which contains a list of the names of the nodes that exist on the network. The names of network nodes listed in NAMSVR.DAT can be used by the MAIL, NET, PRINTER, WHO, WRITE, and other MC-DOS commands to identify nodes. The NAMSVR.DAT file will automatically be created when the SYSINIT.SUB file runs during system startup. The SYSINIT.SUB file runs the NAMENODE program which automatically adds to the NAMSVR.DAT file the name of the node from which the program was initiated. The NAMSVR.DAT file should always be located in user area 0 of logical drive A: of node 0 (the "system node").

To view the NAMSVR.DAT file, use the NET NAMES command or the NAMENODE program. If you change the names of network nodes, or add more MC-DOS systems to your network, you must modify the NAMSVR.DAT file with the NAMENODE command. See the DR NET USER'S MANUAL for additional information on the NET and NAMENODE commands and on the NAMSVR.DAT file.

NET
NET option localdevice = remotedevice @node
NET DISPLAY file
NET HELP

The NET program is only used if your MC-DOS computer is part of a DR Net network. NET is the only special network command that most users need to learn if their computer is linked to a DR Net local area network. The other special DR Net commands and files (CONFIG.NET, NAMENODE, NAMSVR.DAT, and SETNET) are usually only of concern to the system manager.

The NET program can operate in either of two ways:

- In a **menu-driven** mode, requiring you to select options from menus that are displayed if you type the NET keyword alone;
- In a **command** mode, requiring you to provide information after the NET keyword, to specify what you want NET to do.

The menu-driven mode is **easier to use** if you are unfamiliar with DR Net. The command mode is **quicker to use** if you are an experienced DR Net user who needs to change the way in which your virtual terminal is connected to the resources of the network.

In either the menu-driven or command mode, the NET program is powerful and versatile, giving you access to the drives, programs, data, printers, plotters, and "queues" of the network. NET also provides information about the current status of the network, and allowing you to "map" network resources onto your terminal, so that ordinary MC-DOS commands can activate devices that you have mapped with NET. Even more remarkably, NET allows you to save your current mapping of resources as a file. If in the future you need to connect a virtual terminal to the same network drives, printers, and plotters of the network, NET allows you to reconfigure your terminal to match a previously saved state in a single command line. This feature of Gifford's MC-DOS replaces the elaborate step-by-step mapping that is required to configure ordinary local area networks.

If you type NET, this menu appears:

What do you want to do?

0) Change a printer, drive, or queue assignment
1) Display the NETwork status
2) Display the NETwork node names
3) Connect to a NETwork node
4) Disconnect from a NETwork node
5) Save the current NETwork configuration
6) Configure the NETwork to a previously saved state
7) Change the SPOOLER configuration
8) Clear all printer, drive, queue, and node assignments
9) Disconnect completely from the NETwork
10) Display command line help information
11) Exit

Your choice:

By typing a number in the range 0 to 11 you select the corresponding option. Each menu option displays a submenu, which requests additional information on what you want the program to do.

The meaning and use of each NET option and modifier, default values, and other uses of NET in command mode are discussed in the DR NET USER'S MANUAL. Readers who have experience with other implementations of Digital Research's DR Net should be aware that the NET program in MC-DOS combines the functions of several standard DR Net commands. Do not refer to the documentation of NET by other publishers as a guide to Gifford's NET command.

name.NET

The **name.NET** file is only used in DR Net local area networks. This file is created by the NET command, and is a mapping template that stores an image of the current network mapping scheme of the virtual terminal from which you give the NET command. The scheme recorded in this template is recreated automatically when the account that matches **name.NET** logs in to the system. Commonly used DR Net commands can be stored in your **name.NET** template.

To create a **name.NET** file, type NET MAKE. By typing NET MAKE, a mapping template file will be created, using the current mapping scheme of the virtual terminal from which you issued the command. The name of this mapping template file will default to your current login account name, plus **.NET** as the file extension. For example, if you have loged in as "lucy" and have attached your terminal to network resources by various NET commands, type:

```
3D>NET MAKE
```

This creates a mapping template file called LUCY.NET. A message like the following will be displayed on your screen, indicating that the program is creating your template file:

```
Creating file lucy.net
```

If a **name.NET** file exists in your initial default user area, DR Net will use **name.NET** to recreate the network mapping scheme saved in that file. Upon login, the system will respond:

```
Reading configuration from: name.NET
```

Refer to the DR NET USER'S MANUAL for additional information about creating and changing your mapping scheme.

NETMOTD

NETMOTD is the message of the day file for all systems attached to the network. When a user logs in, the network message of the day is displayed as well as the message from the MOTD file of the local node. The file name for the network message of the day is NETMOTD and the file is kept on user area 0 of drive A: of the system node. Use this file for messages that you want all users on the network to see when they log in to the system. This file can contain anything you wish, including escape sequences to control the user's screen. Create and update the NETMOTD file wih an editor such as ED or WordStar (in non-document mode).

NETUSERS

The NETUSERS file is only used if your MC-DOS computer is part of a DR Net network. The NETUSERS file must be located in user area 0 of drive A: on the system node. This file contains the names of the accounts on the network that are recognized by any program that needs to know if a user's account name is valid across the network.

To create the NETUSERS file, the system manager must create a file, with WordStar or any text editor in non-document mode, and enter each account name across the network on separate lines.

There should be no duplicate entries in this file. Otherwise, each time that someone uses the ALL option in the MAIL program to send a message to every account on the local system, the same message will be sent to that user twice.

NEWUSER

The NEWUSER program is a security utility used by the system manager to maintain the PASSWD file. To use NEWUSER, the system manager must first have logged in as "system." If someone who has not logged in as "system" attempts to invoke NEWUSER, the system responds:

```
Sorry but the 'newuser' program can only be run from
the 'system' account. Please check with the system
manager.
```

To invoke NEWUSER, typing the keyword NEWUSER causes the main NEWUSER menu to appear, as shown in the following dialogue:

```
0A>NEWUSER

                Gifford Computer Systems - newuser
                        Maintenance Package
            Copyright (C) 1984 by Gifford Computer Systems, Inc.
                Version 2.0 of newuser updated on Aug 3, 1984

        A) Add a user.
        B) Brief listing of users.
        C) Change a user's parameters.
        D) Delete a user.
        E) List users in detail.

        X) Exit.
        ^C) Abort, make no change.

Option:
```

This program falls under the system manager's jurisdiction. For further information regarding this utility refer to **Section 3.3.**

PASSWD

The PASSWD file specifies account login names, encrypted login passwords, initial default user areas, initial default drive designations, optional terminal access restrictions, initial printer assignments, optional default program assignments, a list of accessible user areas, and comments on each account. A sample PASSWD file is listed below:

```
guest::c:14:epson:1:submit demo:14:run demo only
michael:qJNrC8Fby3Mn:c:1:ti:::all terminals and user areas
modem:yT5Mo9b34Axl:b:8::7:message:8:modem bulletin board
nancy:j9hPI92fzl9j:c:10:0:2,3:ws:10:correspondence only
sales:7BoJqvNtCEpd:d:4:ti:4,5,6:dbase prospect:12:
steve:dCTPcl24w9On:a:5:ti:2:dbase mail:5:mailing list only
system:::diablo:::::no restrictions
```

Each entry in the file has nine fields. The fields have the following functions:

```
NAME:PASSWORD:DRIVE:USER:PRINTER:ACCESS:PROGRAM:USER AREAS:COMMENTS
  \       \       \     \       \       \       \          \          \
   1       2       3     4       5       6       7          8          9
```

1. **Account login name.** This can have up to eight characters.

2. **Optional password field.** If it exists, the password is encrypted.

3. **Initial default drive.** This will appear in the MC-DOS prompt.

4. **Initial default user area number.** This will appear in the MC-DOS prompt.

5. **Initial default printer** (default=printer 0). You can enter printer names or numbers. The name or number must be in the LPRS file.

6. **Optional selective terminal access numbers.** This is a list, separated by commas, of the console numbers (from TTYS, field 1) that this account is allowed to log in from. If no assignment is made in this field, the user has access to any terminal.

7. **Optional default application program,** such as SuperCalc 86. Accounts with an entry in this field are automatically entered into the program specified. The system searches for the program exactly as if the user had typed it at the default prompt. The account is automatically logged out after exiting the program. The extension (file type) is not included with the program name.

8. **Accessible User Areas.** This is a list of numbers, separated by commas or hyphens for a range, of the user areas that an account is allowed to access. If no assignment is made in this field, the account has access to any user area. If an account tries to access a restricted user area, the system will deny permission to access that particular user area. For example, if you have been allocated user areas 3, 4, and 14, and you try to move into user area 6 from user area 4, the system will respond with the following message:

You don't have permission to access user area 6.
User number = 4.

9. **Optional comments field.**

This contains any comments that the system manager may wish to include regarding a PASSWD account.

Additional Notes on the PASSWD File

The PASSWD file must not be empty. If the file is empty or damaged, the system will lock up on startup and you will have to call your dealer. The PASSWD file should only be changed by the system manager, and the NEWUSER program should be used to used to implement any changes in the file. Since modification of this file is restricted to the system manager, it is documented in more detail in **Chapter 3.**

Do not use a text editor to create a PASSWD file, since you will not be able to encrypt any passwords that you put on account names. MC-DOS will not allow you to log in to an account with an unencrypted password. Use the PASSWORD or NEWUSER commands to put passwords on accounts in the PASSWD file.

MC-DOS will put a special write password on the PASSWD file during the boot process. If the label on drive A: has PROTECT set ON, only the NEWUSER and PASSWORD commands can modify the PASSWD file. Never put your own password on the PASSWD file.

In MC-DOS systems that implement a high level of security, only the system manager's account ("system") should be allowed to access user areas 0 or 15. That is, field 8 of the PASSWD file should not be left blank, but "1-14" or some more restrictive list of user areas should be entered for accounts other than the "system" account. In a secure system, every account should have a password in field 2, to prevent easy access to the system by unauthorized users.

PASSWORD

PASSWORD @node

The PASSWORD program permits individual users on the system to change their login password in the PASSWD file of either the local system or any designated remote system. Because the PASSWD file is password protected, and because the login password field on PASSWD is encrypted, the PASSWD file cannot be edited by ordinary means such as ED, WordStar, or any other text editor. You can only use PASSWORD to change the password of the account to which you are currently logged in. The PASSWORD program cannot be used to put passwords on files or devices.

To run the PASSWORD program, type:

0A>PASSWORD

For example, if you logged in as "Lucy," the program will respond:

Gifford Computer Systems - Password - version 2.0
Changing password for Lucy

If you already have a password, the program will prompt:

Old password:

If what you type does not match your old login password, the program will respond:

Sorry. You did not type the correct password.

At this point, if you did not correctly enter your old password, you will automatically exit from PASSWORD. You will see the system prompt, and you must invoke the PASSWORD program again to make another attempt to change your password. (The purpose of this feature is to prevent unauthorized users from changing your password.) If you have no current password, or if you typed it correctly in response to the earlier prompt, the system will ask:

New password:
Retype new password:

When you respond to this prompt, your keystrokes will be echoed as blank spaces on the terminal screen. This is to insure that no one sees your password as you enter it into the PASSWD file.

Your password can be any string of up to eight ASCII characters, including control characters, followed by a carriage return. Unlike in file names, any characters, including periods, commas, semicolons, and blanks (the spacebar) can be embedded in a password. As before, the screen will show a blank space for each character that you type, rather than your actual keystrokes. Press the RETURN key, and the screen will say:

Type it again:

To successfully change your password, you must type exactly the same thing both times. If you do, you will have a new login password, and you will automatically exit to the system prompt without further dialogue from the PASSWORD program. If your two versions of the new password are not identical, you will first get the error message:

Mismatch - password unchanged

You will also automatically exit from PASSWORD, but without changing your password. The purpose of this feature is to make it unlikely that you will give yourself a new password that is different from what you had meant to type. Be sure to remember your new password, since you cannot refresh your memory by displaying the PASSWD file as the password will always appear encrypted as twelve random characters. If you ever forget your password, you must contact the system manager for help.

Changing Your Password on a Remote System

If you want to change your password on a remote system for which you already have an account in the PASSWD file, you must type PASSWORD, followed by @, and then the name of the remote system. Suppose that you have an account established on a system called SALES, and you want to change your password for that account. To do this, type:

```
0A>PASSWORD @SALES
```

The PASSWORD program will display the same screen prompts that it does if you were changing your password on the local system. It will ask you to enter your old password, and then enter your new password twice.

The next time that you connect to the SALES system, or login from a terminal that is connected directly to the SALES system, you must supply the new password. You must already have an account on the remote system and know the password for that account in order to change it. If you don't have an account on a remote system that you wish to access, an account must be created on the remote system by the system manager for that system using the NEWUSER program. The NEWUSER program can only be used to create new accounts on the local system.

See the DR NET USER'S MANUAL for more information.

PIP

PIP destination:=source device:, file(s) [options]
PIP destination:=d:file [options]
PIP d:[Gn]=**d:file** [options]
PIP d:newfile[Gn]=**d:file**(s) [options]

The PIP command ("peripheral interchange program") is used to copy files to a destination device from a source device. If you learn enough about PIP to be able to make copies of your files, you will reduce the likelihood of losing important data or programs. PIP typically uses logical drives (hard disks, floppy disks) as the destination and source devices, but PIP can also accept printers, plotters, and terminals as devices.

To use PIP to copy file CAT.TXT from drive A: to drive B: in your current user area, type:

```
8C>PIP B:=A:CAT.TXT[VRW]
```

The [VROW] options cause PIP to verify [V] that the copy matches the original, to copy files regardless [R] of whether they have the DIR or the SYS attribute, and to replace with [W] source files any files of the same names in the destination area, even those set with the RO (read only) attribute. **These options [VRW] should be routinely included** in file transfers, but omit [W] if you do **not** wish to update RO files on the destination drive. An "archival" option [A] allows you to selectively copy only updated files (see **Section 3.8-8**). All 21 PIP letter options are listed later in this description.

PIP can accept wildcards (*,?) in the specifications of files on the source device, so that multiple files can be copied with a single PIP command. For example, to copy all files of type .CMD with two-, three-, four-, and five-letter names that begin with the letters "FR" from drive C: to drive A:, give the command:

```
5B>PIP A:=C:FR???.CMD[VRW]
```

To copy all the files in your user area from A: to B:, type:

```
0C>PIP B:=A:*.*[VRW]
```

PIP can also copy files from one user area to another. If the source or the destination is different from your current user area, you must indicate the user area with the [Gn] ("go to user area **n**") option. If you supply no logical drive or no [Gn] option in a source or in a destination, the PIP program supplies as default values your current logical drive and current user area. Compare, for example, the three commands:

```
 0A>PIP B:[G12]=C:HOW.NOW[G7VRW]
 7B>PIP HOW.NOW[G12]=C:[VRW]
12C>PIP B:=HOW.NOW[G7VRW]
```

All three lines accomplish exactly the same thing, transferring the file HOW.NOW from user area 7 of drive C: to user area 12 of drive B:. The last two lines are shorter, because they omit some user area and drive specifications which match default values implied in the system prompts (respectively, user area 7 of drive B:, and user area 12 of drive C:). You are, of course, allowed to enter default values in PIP, as in all other MC-DOS commands.

PIP can also give a new name to the copy of the file on the destination drive. The following lines rename OLD.DOC as NEW.DOC:

```
0A>PIP B:NEW.DOC=A:OLD.DOC[VRW]
0A>PIP NEW.DOC=OLD.DOC[VRW]
```

In the second line above, the source and the destination are the same, namely drive A:. This is not the same as using the MC-DOS REN command to rename files, since after the last PIP command two copies (OLD.DOC and NEW.DOC) exist in the same directory area. If you copy a file to a destination where a file exists with the same name, PIP writes over the existing file, destroying it.

PIP can combine many source files as one destination file:

```
6B>PIP C:BOOK=CHAP1[G2VRW],CHAP2[G2VRW],A:CHAP3[G2VRW]
```

The command line listed above will create in user area 6 of drive C: a file called BOOK that combines files CHAP1 and CHAP2 from user area 2 in drive B:, and CHAP3 from user area 2 of drive A:. Commas (,) are used to separate the various source files. Option [o] allows ^Z only after the third file. Do not use wildcards (*,?) in PIP commands to concatenate files.

Copying Files to Printers

Although typical PIP operations copy files between logical drives (hard disks or floppy disks), other devices can be used as a source or destination. For example, the PIP command accepts LST: as a destination, allowing you to send files to the printer or plotter to which you are attached. The command:

```
8C>PIP LST:=A:TTYS[G0]
```

will print out file TTYS, if your printer is ready. The LST: option is sometimes helpful in sending to certain printers a file containing special strings of characters to change how the printer operates— for example, switching the pitch to a different number of characters per inch. Another PIP destination option, PRN:, causes a printer to number each line of your file, and to expand tabs to every eighth column. These PIP options offer alternatives to the PRINT command, or to the TYPE command with the ^P control character, or to an applications program, or text editor as a way to print files.

If you type PIP without an argument, the program signs on and gives you a prompt (*). You can then enter a valid PIP command line like those described on the following pages, specifying the desti-nation device and source device, but omitting the keyword "PIP." This interactive usage is called the "multiple command" mode. To return to the operating system, press RETURN at the * prompt.

Selectively Copying Only Updated Files

Since most of the 21 letter options accepted by the PIP command are infrequently required, refer to the next two pages as the need arises, rather than memorize all the letter options. Five of the PIP options, however, are very important for everyday operations: Gn, V, R, W, and A. The first four have already been described; use of the **[A]** option is outlined below:

MC-DOS recognizes an "Archive" file attribute, whose "ON" or "OFF" value can be controlled by the SET or PIP command. If a file has this attribute set ON, the operating system will reset the Archive attribute to OFF if the file is modified. For example, if you edit a text file, or add, delete, or edit records in a data file, or create a new file, the new or modified file will have the Archive attribute set OFF. (The SDIR command lists "Arcv" after any file whose Archive attribute is set ON.)

Using [A] as an option in a PIP command has two effects. First, it tells PIP **not** to copy source files that have the Archive attribute set ON. Second, after transferring any file to the destination, PIP sets the source file's Archive attribute ON. (If the transfer fails-- for example, because a diskette did not have enough space-- Archive is not set ON for the source file.) The Archive attribute is not copied by PIP, so all destination files transferred by PIP have Archive set OFF. The command:

```
0D>PIP D:=A:*.*[VRWA]
```

copies to drive D: all of the files on your directory area that have Archive set OFF, and sets Archive ON for each source file.

The practical value of using the [A] option with PIP is in making backup ("archival") copies of files to removable storage media, such as floppy diskettes. Always using the [A] option in making backup copies saves time and storage space, because it avoids repeatedly copying files that have not been modified since the last backup operation. If you specify [A] in an "archival" PIP command, the Archive attribute is set ON. Only files that have been modified since the last backup will be copied.

Option Codes Accepted by the PIP Command

[A] = Do not copy files with the Archive attribute set ON, and source files are set with Archive ON after copying. If [A] is not specified, all files that match the source are copied, and source files are not set with Archive ON.

[C] = You must confirm (by typing (Y) or (N)) each transfer after a PIP command that has wildcard symbols in the source. Unless [C] is specified, all files are transferred.

[Dn] = Delete characters beyond column **n** in each line. This is useful when your destination is a LST:, PRN:, or CON: device that can print no more than **n** characters per line.

[E] = Echo each transfer on your terminal screen. Used only with character (printable) files, this displays the data that you are copying as it is copied. This option slows PIP.

[F] = Do not copy form feeds. If [F] is not specified, symbols to advance the printer to the beginning of the next page will remain in the destination copy of your file.

[Gn] = "Go to" user area **n.** Both the source and the destination files can have [Gn] options. **This is the only option that can follow a destination specification.** If [Gn] is not specified, PIP takes as its default your current user area. The [Gn] option cannot be used to override any user area access restrictions imposed by the PASSWD file.

[H] = This should be used only in transfering hexadecimal files (with extension .H86 for 16-bit files, or .HEX for 8-bit).

[K] = Suppress terminal display during a wildcard file transfer. If [K] is not selected, the screen will list the names of each source file as it is transferred.

[L] = Change all upper case alphabet characters in the source to lower case letters in the destination; opposite of [U].

[N] = Add line numbers, starting with one, and followed by a colon (:) in the destination file.

[N2] = Added line numbers with single digits should be preceded by a zero (0) and a TAB should be inserted after the line number. If [T] is also specified, Tabs (^I) are expanded. This option can be used when LST: (not PRN:) is the destination device.

[O] = Object file transfers. This causes end of file characters (^Z) to be ignored when copying to a character device such as CON: or LST:.

[Pn] = Sets the page length at **n** lines per page. If [Pn] is not specified or if [Pl] is specified, page length becomes 60. This can be used in conjunction with [F] as source options.

[Qs^Z] = "Quit after **s**" option stops after copying a specified sequence of characters **s**. Type the control character ^Z to indicate the end of sequence **s**. Like [Ss], this option only works in the interactive, multiple command mode.

[R] = Read files that have the system (SYS) attribute set. Unless you specify [R], only DIR files will be copied.

[Ss^Z] = "Start at **s**" option begins copying at a specified sequence of characters **s**. Type the control character ^Z to indicate the end of sequence **s**. This option requires the interactive, multiple command mode, where it can be used with option [Qs] to copy specified parts of files.

[Tn] = Expand TAB characters to **n** spaces. This is used to list programs on printers (destination device LST: or PRN:).

[U] = Change all lower case alphabet characters in the source to upper case in the destination file; the opposite of [L].

[V] = Verify that data was copied correctly after each record is written. This option should only be used when a logical drive is the destination device (as in most PIP commands).

[W] = Write over files that have the read only (RO) attribute. If you do not specify [W] and the destination file already exists with the read only (RO) attribute, then you will be prompted:

DESTINATION IS R/O, DELETE (Y/N)?

If you enter any character other than (Y), then the original file remains. If you enter (Y), then the source file is copied, overwriting the destination file.

[Z] = Used only with printable files, this sets the parity bit of every character in the destination file to zero.

Copying Files To or From Consoles

The CON: option (for a terminal or modem) can also be used in PIP commands as a source device (usually the terminal keyboard, or a modem) or as a destination device (usually the screen). The following example is of a PIP command to display on the terminal screen the file MEMO from user area 4 of drive B:.

```
0A>PIP CON:=B:MEMO[G4]
```

This has the same effect as the following TYPE command:

```
0A>TYPE 4B:MEMO
```

PIP can also create a file by directly entering data from the keyboard or modem. For example, the following command line will save any keystrokes that follow as TEXT.KBD, a file in your current user area. Type (^Z) to end the input and exit from PIP. All characters, including backspaces will be entered in the file.

```
0A>PIP TEXT.KBD=CON:
```

Using PIP with Password Protected Files

The PIP command can be used to copy files protected by file passwords, if you know the password. If the SET command has been used to make the source drive recognize file passwords, the correct password must be presented in the PIP command line:

```
0A>PIP B:=A:DATA.TXT;PSWORD
```

This form copies a password protected file with the name DATA.TXT and the password PSWORD from the default user area 0, drive A: to default user area 0, drive B:. Note that when PIP copies a file, **it does not copy any extended file control blocks (XFCBs)** to the destination. Since file passwords are in the XFCBs, they are not copied, so that if a protected file is copied with PIP, the copy is unprotected. If you want to password protect the copy, you must follow the procedures for file password protection that are described on **page 2.5-SET-1.**

Using PIP on Systems Without Hard Disks

If your computer has no hard disk, your system diskette will normally occupy drive A:. To copy files that are not on the system diskette to another diskette, you must use PIP and the multitasking features of MC-DOS. The following procedure allows you to use PIP and the DSKRESET commands, and different virtual terminals, to copy files from diskette to diskette. For this example, you are in user area 0 of drive A:, and drive B: is empty. To invoke PIP in multiple command mode, type:

```
0A>PIP
```

The program responds with the PIP multiple command mode prompt:

```
*
```

Now switch virtual terminals in the normal way, and log in on a different virtual terminal. Suppose in this example that this is user area 1 of drive A:. When the system shows the prompt for your default directory area, you should reset the disk drives:

```
1A>DSKRESET
```

Now load the source diskette onto drive A:, and the destination diskette onto drive B:. Using the Virtual Terminals procedure for switching screens (usually a special function key), switch back to the virtual terminal where you left PIP running.
In this example, let us suppose that all the files to be copied onto B: have the extension .DOC. Now to transfer your files:

```
*B:=A:*.DOC[VRW]
```

Notice that the first asterisk is the PIP prompt, not a wildcard. The screen will list each of your files being copied. If you change diskettes while at the * prompt, then use Virtual Terminals again to detach the PIP process. At the new virtual terminal prompt, type once more:

```
1A>DSKRESET
```

Mount the diskette in drive A:, and use the Virtual Terminals command again to switch back to the original virtual terminal where PIP was running. The screen will list the files copied, and will show the PIP multiple command prompt:

```
*
```

Now the files *.DOC are on user area 0, drive B:. To exit from PIP, remount your system diskette in drive A: and type ^C.

Copying Diskettes on Single Floppy Drive Systems

You cannot directly copy diskettes to diskettes on systems that support only one floppy disk drive. If you have a hard disk, you can copy all of the files on a diskette to an empty directory area. Then give the DSKRESET command, replace the source diskette with a formatted destination diskette, and PIP the files from the hard disk to the destination diskette.

If no empty directory area is available on the hard disk, set ON the Archive attribute to all files in some directory area of the hard disk, and PIP the source diskette files to this directory area. Give the DSKRESET command. Now you can use PIP with the [A] option to copy the files to a destination diskette. If you do not need copies of the source files on the hard disk, use the ERAQ command to delete them.

Additional Notes on PIP

Besides the standard MC-DOS symbols for logical drives (A:, B:, etc.) and the special symbols LST: and PRN: for your current printer and CON: for your current console, PIP accepts two additional symbols for special "source devices:"

EOF: = An imaginary device that always produces a ^Z to mark the end of a file. This can be concatenated to any file that for some reason lacks an End Of File symbol.

NUL: = Another imaginary device used for paper tape readers, that always produces 40 null symbols (hexadecimal zeros).

Warning

If you are upgrading your current computer system from MP/M 8-16 to MC-DOS it is imperative that you use the new MC-DOS version of the PIP program to copy ALL your new system files onto your hard disk. Make sure that you back up ALL of your files onto floppy diskettes (not just your system files), before you copy your new system files onto the hard disk.

See also the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE for additional information on the PIP command.

PRINT

PRINT d:**file** [option], d:file [option], ...

The PRINT command causes your current printer to print either text or data files directly from the operating system prompt, either locally or over a DR Net network, provided that the physical printer is available. The PRINT command does not recognize formatting paramaters used by WordStar or by other text editors, resembling the TYPE command (with ^P). The PRINT command accepts the following options:

[Gn]: The [Gn] option specifies the user area number of the file to be printed. This lets you print files located outside your current user areas, but this cannot be used to override any user area restrictions of the PASSWD file.

[S]: The [S] option is required if the file to be printed has the SYS attribute (like the [R] option of PIP). Files with the DIR attribute do not require the [S] option.

[Tn]: The [Tn] option expands tab characters (^I) so that the next character is printed in a column exactly divisible by **n**. If **n** is 0, tabs are not expanded. If [Tn] is not specified, PRINT uses a default of **n** = 8 spaces.

A single PRINT command can print a list of files:

```
2B>PRINT *.OLD, C:NEWS[S], ELAINE.REM[G4]
```

This example prints all files with extension .OLD in user area 2 of drive B:, then a system file NEWS in user area 2 of drive C:, and finally the reminder file ELAINE.REM in user area 4 of drive B:. While each of the files specified on the command line above is printing, the PRINT command will display the following messages on your screen:

Printing file: filename.
Press SPACE BAR to stop.

Pressing the space bar causes printing to be interrupted (after the printer's buffer is empty), and the program asks:

Do you want to S(top printing), skip to the N(ext) file, or R(esume printing this file) ?

If you type (S), you exit from PRINT. If you type (N), printing resumes on the next file. If you type (R) printing resumes where it stopped. If the printer is busy with another job, the following message is displayed on your screen:

Printer is being used by another console.
Do you want to (W)ait or (C)ancel?

If you type (W) the program waits until the printer is free, then prints your file(s). If you type (C), PRINT terminates without printing, and returns the system prompt. You must type RETURN after any letter response to a PRINT prompt. (Use the PRINTER command to select a printer other than your current printer before you give the PRINT command, if you want to use a different printer, or if your current printer is not available.)

If the PRINT command encounters any errors, it displays an error message on your terminal screen, and skips to the next file to be printed, if any, that was specified on your command line. When PRINT finishes printing a file, the following message is displayed on your screen:

Printing is completed.
p/c files printed.

Here **p** is the number of files that were successfully printed and **c** is the number of files that were specified on the command line. See also the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE for additional information.

PRINTER

PRINTER n
PRINTER name @node
PRINTER name account **@node**
PRINTER ?

The PRINTER command with no argument displays the printer number assigned to your terminal. If you specify a name or a number, that printer (or plotter) is assigned to your current terminal. The name specified in the command must be in the LPRS file. Different virtual terminals supported by the same physical terminal can select different printers by the PRINTER command. If your system is part of a DR Net network, the PRINTER command can access printers at other systems within the network.

Examples:

The following command will show the correct format for using the PRINTER command:

```
0A>PRINTER
```

The following command assigns a printer, called DRAFT in the LPRS file, to your current virtual terminal:

```
2D>PRINTER DRAFT
```

The command below assigns the printer with the name LPR0 (printer number 0) to your current virtual terminal:

```
4C>PRINTER LPR0
```

The following selects printer number 1 in the LPRS file:

```
2D>PRINTER 1
```

If you type:

```
0A>PRINTER ?
```

THE PRINTER command will display the printer numbers and names that exist in the LPRS file.

Accessing Printers on Other Systems of a Network

The PRINTER command also allows you to select a printer on a different system within a DR Net network. For example:

```
10B>PRINTER DIABLO @MARKETIN
```

This assigns a printer, called DIABLO in the LPRS file of the system called MARKETIN, to your current terminal. For this command to work, however, the account name by which you logged in to your local system must also appear (with the same encrypted password) in the PASSWD file of the system called MARKETIN. The "at" symbol (@) must be prefixed to the system name. If your account name appears in the PASSWD file of the MARKETIN system, but with a different password, the operating system will ask for your password as it appears on the MARKETIN system. The screen will not echo the password that you type.

If you type the password correctly, you will be connected to the DIABLO printer of the MARKETIN system. If you cannot present the correct encrypted password, you cannot connect to the printer of the networked system. If your account name does not appear in the PASSWD file of MARKETIN, you will be asked for the password, but there is no correct answer-- you cannot connect without matching the PASSWD file of MARKETIN.

If the account name by which you logged in does **not** appear on the password file of the networked system whose printer you want to use, you can instead give a PRINTER command that includes a valid account name that appears in the PASSWD file of the other system. The account name appears between the printer name/number and the system name in a PRINTER command like the following:

```
4E>PRINTER 2 ANN @SALES
```

This attempts to assign printer 2 of the Sales Department system to your current virtual terminal. The account name "ANN" must be in the PASSWD file of the system called SALES. If ANN is a "public" account name (one with no encrypted password), or if the account name by which you logged in and ANN have the same encrypted password, the screen will print a message that you are now connected to that printer on the SALES system, and will display the SALES system's MOTD file, if one exists.

Thus, if you need to connect to a printer on a system which does not list your account name in its PASSWD file, you can do so with the command line above by presenting another account name whose encrypted password on the other system either (1) does not exist, or (2) is the same as the password in your local PASSWD file by which you logged in to your local system.

Continuing the same example, if the "ANN" account name that you use in a PRINTER command has an encrypted login password on the remote system's PASSWD file, and if this does **not** match the password of your current account name on your local system, then the operating system will ask you to present the password of ANN on the SALES system's PASSWD file. If you type it correctly, you will be connected to the SALES printer; otherwise you will get a message that your password was not valid. Just as with local printers, connecting to a networked printer will only allow you to use the remote printer if it is available. If a printer is busy, you must wait until work in progress is completed.

The LOGOUT command or rebooting the system reassigns printers that have been selected by the PRINTER command, (but LOGOUT does **not** detach list devices attached by the ^P control character).

The MC-DOS PRINTER command differs from the standard Concurrent DOS PRINTER command in allowing account numbers, and network nodes, and printer names from the LPRS files to be used, as well as numbers, as arguments of PRINTER. See the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE for additional information on the standard Digital Research Inc. PRINTER command.

PWASK.NET

The PWASK.NET file is a security feature used only by the system manager of a system in a DR Net local area network. A remote user who attempts to connect to your node must present a login password that matches the entry for the current login account name in the PASSWD file of your node. If the password is incorrect, or if the PASSWD file does not list the current login name, the remote user cannot connect to your node.

An exception to this rule, if the file PWASK.NET does not exist, is that a remote user does **not** need to present a login password if his or her current login account name and login password exactly match corresponding entries in the PASSWD file of your node. That is convenient in some situations, but it means that an unauthorized person who finds an unattended logged-in terminal could use that terminal to access other network nodes to which the currently logged-in account name is allowed access.

If the PWASK.NET file exists on user area 0 of drive A: of a node, the exception described in the previous paragraph is eliminated: every attempt to connect to a node requires that a login password for the current account name be presented that matches the entry for that account in the PASSWD file of the node being accessed. In the example of the unattended terminal, the person at the keyboard could not access other nodes that has PWASK.NET files unless the correct password were presented.

Like AUTOST, the PWASK.NET file can be empty, since it signals to the network input/output system (NIOS) that a login password must always be presented when someone tries to connect to a node that has a PWASK.NET file. To create PWASK.NET, type:

```
0A>MAKE PWASK.NET
```

Now the operating system will ask anyone attempting to connect to your node for a password. The PWASK.NET file affects only the local node, so that some nodes of a network can implement this protection, while those at other nodes can elect not to do so.

name.REM

The reminder file, **name.REM**, is an optional file which will be displayed when the user to whom the reminders are directed logs in. The file name "name" must match the login account name in the PASSWD file. The reminder files are kept in the default user area and drive of each account.

Use the **name.REM** file to leave yourself a reminder to do something, since you will see the reminder message each time you log in to the system. To make a reminder file, create the file with an editor such as ED or WordStar (in non-document mode) from the default drive and user area of your account. If you created a **name.REM** file from a drive and user area other than the initial, you must copy that file to the initial drive and user area which are specified in the login PASSWD file. For example, if you have logged into the account "jane," and the default drive for the account "jane" is drive D: and the default user area is 9, then from a different drive and user area you would type:

```
3C>PIP D:[G9]=JANE.REM[VROW]
```

Once the file is on user area 9 of drive D:, there is no need to leave it in user area 3 of drive C:. To erase it from user area 3, drive C:, area give the command:

```
3C>ERA JANE.REM
```

You can, of course, leave reminder messages for other users using the above method, but it is more polite to send them mail with the MAIL utility.

REN

REN d:**newfile=oldfile**

The REN command renames files that have the RW (read-write) attribute. The new name must be on the left-hand side of the equal symbol (=), and the old name on the right-hand side.

In the example below, a REN command changes to "ROSE.BUD" a file previously called "ROWS.BUD" in user area 2 of drive A:

```
2C>REN A:ROSE.BUD=ROWS.BUD
```

If you rename a file with a new name that is the same as another file's in the same directory area, the file that first had the "new name" will be deleted. The REN program warns you, and asks you to confirm that you want to go through with this:

```
Not renamed: ROSE     .BUD already exists, delete (Y/N)?
```

If you type (Y), the current file ROSE.BUD will be erased, and the file called ROWS.BUD will be renamed ROSE.BUD. If you press any other key, the program will terminate without doing anything.

Wildcard specifications (*,?) are permitted, but must occur in the same parts of both the new and old names. The command:

```
9A>REN *.OLD=*.NEW
```

changes every file extension .NEW to .OLD in your directory. The REN command cannot rename files that have the RO (read-only) attribute. The REN command does not and cannot change a file's attributes (such as Archive, SYS, DIR, or RW), password, user area, nor logical drive. See also the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE for more examples of the REN command.

RESET

The RESET command allows changes to be made in existing entries in the TTYS and LPRS files without having to shut down and start up the system. When the RESET command is given, the computer executes a SYNC and reads the LPRS and TTYS files. The RESET command cannot be used to add more terminals or printers than are currently described in the files.

To use RESET, first make any changes to the TTYS or LPRS files, then type:

```
OB>RESET
```

The system will make any necessary changes indicated in the files.

SDIR

SDIR d1:,d2:,...,dn: [option(s)]
SDIR [option(s)] d:file(s)
SDIR d:file(s) [option(s)]

The SDIR command displays the names and the attributes of files in directory areas. SDIR is used to locate files, and to display information about files and logical drives. SDIR can accept wildcards (*,?) in file names. The SDIR command is helpful in systems with large hard disks, because it can help you search for a file when you are not entirely certain of the drive, the user area, or file name that were used to store the file.

The SDIR command resembles the DIR[S] command, but has more options, displays file size and attributes, sorts the files alphabetically and displays the date and time of the latest file update and accessing. The default (no options) form of SDIR gives detailed information on every file in your directory area:

0A>SDIR

This displays in alphabetical order the names and extensions of all the files in your current directory area, as well as:

- The logical drive letter of your current directory area
- The user area number of your current directory area
- The size (in 1024-character "kilobytes") of each file
- The number of 128-character "records" occupied by each file
- The Dir/Sys, RW/RO, Archive ON/OFF attributes of each file
- The total number of 128-character records in your directory
- The number of files in your directory
- The total number of 1024-characters blocks in your directory
- The ratio of directory entries to the maximum number allowed

On a logical drive with a label to support extended file control blocks (XFCBs) for passwords, the same command also displays:

- The level of password protection assigned to each file, if any
- Timestamping information, if timestamping has been enabled

The SDIR command accepts the following options and modifiers that specify the files to be listed, or the format in which to display all files that match your command line specifications:

```
SDIR [ATTRIBUTES]     - list the status of user-defined attributes

SDIR [DIR]            - files with directory attribute DIR

SDIR [DRIVE=d]        - files from drive d:
SDIR [DRIVE=ALL]      - files from all connected drives
SDIR [DRIVE=(AB...d)] - files from drives A: B: ... d:

SDIR [EXCLUDE]        - list files that do not match
                        wildcards (*,?) after the [EXCLUDE] option

SDIR [FF]             - add form feed(s) to the beginning of any
                        printed listing

SDIR [FULL]           - default display format (see previous page)

SDIR [LENGTH=n]       - repeat table headings after n lines

SDIR [MESSAGE]        - identify each directory area searched

SDIR [NONXFCB]        - files without extended file control blocks

SDIR [NOSORT]         - sequenced as found, not alphabetically

SDIR [RO]             - files with the read-only attribute RO

SDIR [RW]             - files with the read-write attribute RW

SDIR [SIZE]           - display only names and file size data

SDIR [SYS]            - files with system attribute SYS

SDIR [USER=n]         - files in user area n
SDIR [USER=ALL]       - files in all accessible user areas
SDIR [USER=(0,1,...,n)] - files in user areas 0, 1, ..., n

SDIR [XFCB]           - files that can have passwords
```

Any option (or list of up to 10 options) must appear between a single pair of square brackets ([]) in an SDIR command line, with no more than one option list per command. If an option list combines several options, these must be separated by commas or by blanks. Any list of several user areas or logical drives used as modifiers of a USER or DRIVE option must be enclosed in "()" (parentheses), with the items separated by commas or spaces.

It is not necessary to type more than the first two letters of an SDIR option. The following are completely equivalent SDIR command lines:

```
12E>SDIR *.CMD [DRIVE=B,USER=(2,4,6),SYS,MESSAGE]
12E>SDIR B:*.CMD [USER=(2,4,6), SYS, MESSAGE
12E>SDIR [USER=(2,4,6) SYS MESS] B:*.CMD
12E>SDIR B:*.CMD [US=(2 4 6) SY ME
```

Each tells the system to list in [FULL] format all of the files of type .CMD with the SYS attribute in user areas 2, 4, and 6 of logical drive B:, separately listing such files for each directory area. If any of the three has no file matching the specification, the [MESSAGE] option indicates this on the screen.

The SDIR command can search for a file whose directory area is unknown. The advantage of using SDIR rather than DIR to locate files is that SDIR can examine multiple user areas and multiple logical drives in a single command line. For example:

```
9A>SDIR LOST.DOC [USER=ALL, DRIVE=ALL]
```

will display a directory of all the files called LOST.DOC that are stored on your system, showing their directory areas, size, and attributes (if indeed any file of that name exists).

Every SDIR option list applies to all files that match the specification. If you do not indicate what files you want listed, SDIR will show all the files in the specified directory area(s), as if *.* were your file specification.

For example, the command line with no file specification:

```
6B>SDIR [RO, USER=ALL]
```

lists all files on drive B: that have the read-only attribute. Use ^S and ^Q to stop and to restart long SDIR screen displays.

The default display format is the option [FULL]. This displays one file per line (if the logical drive has a label) or two per line (if it has no label). The [SIZE] option uses a more compact display format with three files per line. For example:

```
4C>SDIR A:, B: [SIZE]
```

This example command would list the names of all files in user area 4 of drives A: and B: as separate directories, and would show how many 1024-character blocks each file occupied. Unlike [FULL], the [SIZE] format does not show the number of records, file attributes, file password protection, nor timestamps.

The [EXCLUDE] option is used to list files that do **not** match a wildcard specification. For example, the command:

```
4C>SDIR A:*.CMD [USER=0, DIR, EXC]
```

will list all files on user area 0 of drive A: that have the DIR attribute, **except** for files that have the .CMD extension.

SDIR is helpful in cataloging files, because a SDIR display can be sent to your printer for output on paper. The [LENGTH=n] and [FF] options can be used with the MC-DOS control character ^P to print out in the [FULL] or [SIZE] formats a listing of the files in any directory areas, or of all the files on a disk.

See also the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE for additional information on the SDIR command.

SET

SET d:**[option(s)]**
SET d:**file [option(s)]**

The SET command is used to specify the attributes of files and logical drives. The SET command can be used to specify whether files can be freely read, copied, modified, renamed, or erased, and to indicate the specific attributes of any files. Wildcard (*, ?) symbols can be used with SET to assign the same attributes to a group of files.

The SET command can assign to any logical drive a **label** that allows the drive to recognize **file passwords** on files. SET can then assign a password to any file, command, or label on that drive, so that only those who give the password are unrestricted in their access to the password-protected files.

SET can also assign a **default password** to a terminal or modem, so that it always presents that password. This security feature can be used with applications programs that have no provision for file passwords, but which must process sensitive data in a multiuser environment, as described in **Section 3.4.** The SET command is so powerful that in some systems the system manager may decide to protect it with a file password. SET also controls the Archive attribute used in backup operations, and the process by which MC-DOS puts date and timestamps on files.

SET can prevent files from being erased. For example, to give the read-only (RO) attribute to logical drive B:, type:

2C>SET B:[RO]

Now files on drive B: can be read, but cannot be created, deleted, modified, renamed, or copied to B:. Substituting RW for RO in the same command line assigns the read-write (RW) attribute to the drive, so that files on drive B: can be freely manipulated, unless restrictions are put on individual files of B:. Typing a ^C or DSKRESET will clear the RO status back to RW status.

Setting an entire **logical drive** to read-only (RO) is not recommended for most systems as a procedure for protecting files, since (1) no one can write on a RO drive, and (2) another user might reset your drive to RW, unprotecting your files. It is listed here, however, as one capability of the SET command.

An individual **file** (or with wildcards, a group of files) can be SET with RO or RW attributes, and can be given the system attribute (SYS) or directory attribute (DIR). For example:

```
2B>SET *.CMD [SYS,RO]
```

This gives SYS and RO attributes to every file of the type .CMD in user area 2 of drive B:. This differs from the command on the previous page (to make drive B: RO) in that:

1. Only files, not a drive, can have the SYS attribute.

2. A PIP command with the [W] option can override the RO status of a file, but not of a logical drive with the RO attribute.

3. Users can write on drive B: after the command on this page, but not after the previous command.

4. The first command affects all files on all user areas of B:; the second only affects files of type .CMD that were in user area 2 of drive B: when the command was given.

5. A ^C or DSKRESET command gives drives the RW (read-write) attribute, but does not change file attributes.

You can invoke command files with SYS and RO status in user area 0 of drive A:, regardless of your current drive and user area, and regardless of whether the PASSWD file allows you to access user area 0. Certain files, such as some .OVR (overlay) files, must exist in user area 0 of your current logical drive, or in your current directory area, in order for programs that use them to execute properly. To erase a RO file, you must first use SET to give the file the RW attribute.

Attributes that the SET command controls are specified by the options and modifiers listed below. Those in lower case letters are variable names that the user must supply.

ARCHIVE=OFF	Enables a PIP[A] command to copy a file
ARCHIVE=ON	Prevents PIP[A] from copying a file
DIR	Non-system file attribute, opposite of SYS
RO	Read-only status (ERA, ERAQ, REN disabled)
RW	Read-write status (ERA, ERAQ, REN will work)
UPDATE	Turns ON UPDATE timestamps on the default disk drive
CREATE	Allows time of file creation to be recorded
ACCESS	Turns ON the ACCESS timestamps on a disk drive
SYS	System file attribute, opposite of DIR
DEFAULT=password	Makes your terminal always present a password
PASSWORD=password	Requires a password to modify a label
PASSWORD=(RETURN)	Removes password protection of a label
NAME=label.typ	Creates or renames a label on a logical drive
PROTECT=ON	Enables a drive to recognize file passwords
PROTECT=OFF	Prevents a drive from recognizing passwords
PROTECT=DELETE	Password required to erase or rename file
PROTECT=WRITE	Password required to modify, erase, rename file
PROTECT=READ	Password required to do anything to file
PROTECT=NONE	File has no password protection
F1,F2,F3,F4=ON	Compatibility file attributes; not for ordinary use
F1,F2,F3,F4=OFF	(See page 2.12 of DRI's Concurrent CP/M Programmer's Guide)

The Archive Attribute

SET accepts a file attribute option called archive, which can be displayed (as "Arcv") by the SDIR command. The [ARCHIVE] option of SET can be used to control which files are copied by PIP (or by similar tape backup commands) when files are backed up. The example command line below sets the archive attribute ON for any file whose extension is .BAK in your current directory.

```
4B>SET *.BAK [ARCHIVE = ON]
```

This command would **prevent** a PIP command with the [A] option from copying any .BAK file from your current directory, since PIP's [A] option makes PIP copy only files whose archive attribute is OFF.

PIP also sets ON the archive attribute of any source files that are copied by a PIP [A] command. If your files of type .TXT had their archive attribute set ON, but you wanted them recopied,

```
3A>SET *.TXT [ARCHIVE = OFF]
```

would cause PIP with [A] to copy all files with extension .TXT from your current directory.

Drive Labels

A **label** can be placed on a logical drive or a diskette by the SET command, allowing files to be marked with **timestamps** or set with **file passwords.** The label itself can be protected with a password, to prevent modification of the label. For example, to put a protected label called "BELLA" on drive E: so that logical drive E: recognized file passwords, give the SET commands:

```
2E>SET [NAME=BELLA]
2E>SET [PROTECT=ON]
2E>SET [PASSWORD=XYZZY]
```

After each of these three commands, MC-DOS would indicate the status of the drive label with a screen display, as shown in **Section 3.4** but omitted here. The first command creates a label called "BELLA," the second enables the drive to recognize file passwords, and the third prevents the label from being modified unless the password XYZZY is presented. The system manager cannot modify a drive label if its password is forgotten.

Files in all user areas of a logical drive are affected by the attributes of the drive label. Once a drive has a label, the only way to delete the label is to reformat the drive.

File Passwords

If a logical drive has a label that has the PROTECT attribute ON, then the SET command can be used to put a password on any file on that drive. A password can be up to eight characters long. The SET command does not distinguish between upper case and lower case letters in passwords.

Passwords can specify various levels of protection: **none** (anyone can do anything to the file), **delete-protect** (the file can be read or modified without presenting the password, but cannot be erased), **write-protect** (the file can be read or copied, but not modified or deleted), or **read-protect** (no one can do anything to the file without first presenting the password). If you do not specify any PROTECT mode when you set a file password, the operating system selects the READ mode, the highest level of protection. For example, to put the read-protect password "HIDDEN" on the file "FINANCE.DOC," type:

```
0A>SET FINANCE.DOC [PASSWORD = HIDDEN]
```

This prevents the file from being read, copied, erased or modified. To put a write-protect password on the file in the last example, you would type:

```
0A>SET FINANCE.DOC [PASSWORD = HIDDEN, PROTECT = WRITE]
```

If you decide to use file passwords, it may be efficient to assign the same password protection to groups of files that are used in the same way. A default password on a terminal can simplify data processing procedures when protected files have a common password. By defining groups of file names in the usual wildcard notation (*, ?) of the PIP, DIR, and SDIR commands, you

can give a group of files the same protection level or a common password with a single command. For example, to protect from erasure all .DBF files whose names begin with "B" on drive A: (if A:'s label has the PROTECT attribute ON) you could type:

```
0A>SET B*.DBF [PASSWORD = INDELIBL, PROTECT = DELETE]
```

This will require the password "INDELIBL" to erase any of the .DBF files beginning with the letter B, such as:

```
0A>ERA  BOX1917.DBF;INDELIBL
```

Timestamping Files

As an option, MC-DOS can record the time and date when each file on a logical drive was created, accessed, or modified. Only a logical drive that is SET with a label can support date and timestamping of files (**Section 3.8**). The INITDIR command must have previously been invoked, with no other process active, to format the directory of a logical drive to accept timestamps.

```
4B>SET [ACCESS = ON]
```

The example SET command above causes a drive with a label to record in the directory the most recent time and date when each file was accessed. Alternatively, the time of creation can be recorded by the option [CREATE = ON]. The [CREATE] and [ACCESS] options are mutually exclusive-- you cannot have both. The SDIR command displays the date and timestamps.

A timestamp that shows the last time a file was modified is activated by the command:

```
4B>SET [UPDATE = ON]
```

Once you turn a timestamping attribute ON, it is in effect for any files that you access or add to that logical drive, unless another SET command is used to turn it OFF. If the drive label has a password, a user cannot change the timestamping options of a drive without giving the label password. Since time and date information occupy space in a file directory, you cannot have as many entries in the directory areas of a drive with timestamping attributes set ON as on a drive with timestamping OFF.

See **Section 3.4** for more on file passwords. See the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE for additional examples of using the SET command.

SETMEM

SETMEM hhhh

The SETMEM command sets an upper limit on the amount of RAM allocated for programs. This is useful if you intend to create large spreadsheets with SuperCalc 86 and need to have more RAM than the default value of 128 kilobytes (128K). The command is typed with a hexadecimal ("hex") number to specify the number of **paragraphs** of RAM that any program can use as a transient program area. A "paragraph" is 16 bytes (characters), written **10** in hex.

When using SETMEM, keep in mind the amount of RAM you have in your computer, the size of the operating system, the number of users on the system, and the types of programs users normally run. Programs which run on the 8-bit processor require just under 70K (the exact value in hex is 110C). This includes RAM for the program and RAM for SW. Programs which run on the 16-bit processor require anywhere from 16K to 1 megabyte of RAM, depending on the nature of the program.

To use SETMEM, type the command and the hex value for the amount of RAM to allocate for programs. Use the following table as a guide:

Hex paragraphs	**Kilobytes (K = 1024 decimal)**
400	16K
110C	69.5K (minimum allowable size for 8-bit programs)
2000	128K
3000	192K
4000	256K
6000	484K
8000	512K

If RAM space is limited on your computer and you want to create large spread sheets, you might consider running SuperCalc 86 under single user CP/M-86, which requires less RAM than MC-DOS.

Note that SETMEM affects an entire MC-DOS computer, not just a drive or terminal. Each node of a DR Net network can have a differentvalue for SETMEM. An example of a SETMEM command is:

```
0A>SETMEM 110C
```

This sets the memory allocation to 69.5K, the minimum size required to run 8-bit programs.

```
0A>SETMEM 4000
```

This sets the memory allocation to 256K. This is a practical size for large SuperCalc spreadsheets.

SETNET

The SETNET command is used by the system manager in a DR Net network to configure the local system to the specifications that are designated in the CONFIG.NET file. The SETNET command is not used in MC-DOS systems that are not in local area networks. The CONFIG.NET file is supplied on your networking diskette. Some of the lines of the CONFIG.NET file are static, and can only be changed when the system is rebooted. These are:

node number
number of servers
number of requesters
number of message buffers
number of requester configuration tables

If you have changed any of these lines in the CONFIG.NET file, you must reboot your system for the changes to be implemented into the operating system.

The rest of the lines in the CONFIG.NET file can be changed dynamically using the SETNET command. These are:

default user password
timeout
list of private drives
mapping of: drives
printers
queues

If you have changed any of these lines in the CONFIG.NET file, you can implement these changes into the operating system by typing the following command:

0A>SETNET

For a more detailed discussion on the SETNET command, read Gifford's DR NET USER'S MANUAL.

SHOW

SHOW d: option

The SHOW command tells you about the logical drives of your system. SHOW is used to see if enough space is available on your disk, to look at the label or physical format of your drives, and to list the directory areas that are being used to store files.

The SHOW command accepts any one of four options described below: SPACE, DRIVES, USERS, and LABEL. These describe any drive specified before the option. If no drive is specified, USERS and LABEL describe **your current logical drive**, but SPACE and DRIVE describe **all of the drives** that users have accessed since your system was last rebooted (with exceptions, such as drives from which users detached with ^C or ABORT). A fifth option, HELP, displays a brief summary of SHOW syntax.

SPACE This displays how many 1024-character blocks remain on a logical drive. This is also the default SHOW display, if no option is given. A typical display of SPACE is:

```
A: RW, Space:     2,388K
B: RW, Space:     4,336K
D: RW, Space:       896K
```

Here drive C: is not shown, since no one has accessed it. If you specify a drive before SPACE, only the space remaining on that drive will be displayed.

DRIVE Displays the disk format. A sample DRIVE display is:

```
     A: Drive Characteristics
86,912: 128 Byte Records
10,864: Kilobyte Drive  Capacity
 1,024: 32 Byte  Directory Entries
     0: Checked  Directory Entry
   256: Records / Directory Entry
    32: Records / Block
    64: Sectors / Track
     7: Reserved  Tracks
```

USERS This displays your current user area number, the user areas on the drive where files exist, and the number of files in each user area. A typical USERS display is:

```
B: Active User : 10
B: Active Files:  0  2  4  7  8 10 12 14
B: # of files  : 42 13 26 34 15108 24 76
```

Here there are eight active user areas, with the largest number of files (108) in user area 10. This option is helpful in backing up files to diskette or tape, to make sure that you copy all the files that are on a drive.

LABEL This option produces an error message if the logical drive has no label (see the SET command). If a label exists, the LABEL option displays the label (with any timestamps), and the ON/OFF status of the attributes PROTECT, CREATE or ACCESS, and UPDATE.

The following command would show the label (if any) of A:

```
11D>SHOW A:LABEL
```

The next command shows how many kilobytes of unallocated space remain on each of the logical drives currently active:

```
0A>SHOW
```

The directory areas of drive C: are summarized by:

```
3B>SHOW C:USERS
```

The first letter of any SHOW option can be substituted for the complete option. See the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE for additional examples.

SPACE

SPACE d1: d2: ... d:

The SPACE command is used to show how much space remains on logical drives. This command is helpful when you begin work on a long file, or before you use PIP or TYPE to copy a file or group of files to another drive, since insufficient available space on a destination drive causes a BDOS error message. In estimating space requirements when using a text editor, remember that most make backups (files of type .BAK) that double storage requirements if (as recommended) you intermittently save work-in-progress. To use the SPACE command, type the keyword SPACE. For example:

```
12C>SPACE
Drive    Max     Used    %      Free
A:      8860     7672   86%     1188
B:      8860     3656   41%     5204
C:      8860     8072   91%      788
       -----    -----   ---    -----
       35440    26220   73%     9220
```

This displays the storage capacity in kilobytes (Max), the space currently occupied (Used), the proportion of capacity occupied (%), and the unoccupied space in kilobytes (Free). The bottom line shows the aggregated figures for all the currently logged-in logical drives of your virtual terminal. You are **logged in** to the drive named in your MC-DOS prompt, and to the system drive (A:). You are also logged in to any other drives that you have accessed, unless a ^C control character or an ABORT or DSKRESET command has logged you off a drive.

With a drive name (or list of drives, separated by blank spaces) after the keyword SPACE, only the logical drives that you list will be displayed. You do not need to type the colon (:) in a drive name. In a DR Net network, SPACE will identify with the @ symbol and node name any drives on remote nodes to which you are logged in. The SHOW command gives a less detailed display of the available space on logged-in logical drives.

SPOOL

SPOOL PRINT **file** file file ... [option, option, ...]
SPOOL HELP
SPOOL DELETE n
SPOOL START n n n ...
SPOOL command

The SPOOL command is used to send files to printers. If a system (or a network) has multiple printers, the SPOOL command can select from among up to 16 different printers to print a file. Printers accessible by the SPOOL command can be attached to different systems within a DR Net network. Only ASCII (character) files can be printed out by the SPOOL command.

DSP and SPL are background processes that are run automatically by the SPOOL program. You must have the DSP and SPL files on user area 0 of drive A: to run the print spooler system. Files with a .DSP filetype are created after the printer spooler system is started. These files are created for internal program maintenance purposes. Never edit any files with the .DSP filetype, and do not erase them. Under no circumstances should you invoke these programs.

The SPOOL program sets up **queues** that list the files waiting to be printed by each printer. One advantage of using the SPOOL command, rather than sending a file directly to a printer, is that your terminal is free after SPOOL sends your file to a queue, regardless of how long the printer takes to complete its output, or whether the printer is busy with other jobs.

Setting up the **spooler system** that supports the SPOOL command is the responsibility of the system manager, involving procedures outlined later in this description and in **Chapter 3.** Unless those steps are followed, the SPOOL command cannot print files. The next page lists several commands and options that are accepted by SPOOL. A HELP command displays a summary of the usage of SPOOL commands and options:

```
9B>SPOOL HELP
```

Using the SPOOL Command

The command PRINT, followed by a file name, sends the file that you specify to the queue for your current printer:

```
0A>SPOOL PRINT MEDIA
```

This example command line sends the file called MEDIA in user area 0 of drive A: to the printer attached to your terminal. Since the default command for SPOOL is PRINT, the line:

```
0A>SPOOL MEDIA
```

is completely equivalent. Nine different modifiers are accepted by the SPOOL PRINT command. These follow the file in square brackets ([]), and are listed below. Underlined letters show the number of characters required for MC-DOS to identify each name:

[COPIES=n]	Print **n** copies. If not specified, program will print one copy.
[FORMAT]	Divide into 55-line pages on 66-line forms. Used with text that lacks line and page breaks.
[FORMSIZE=n]	Formfeeds every **n** lines. Default is 66. Specify if 66 lines do not equal one page.
[MARGIN=n]	Starts printing **n** spaces from left-hand margin. Default = 8. Range: 0 to 50.
[NOFEED]	Prevents initial formfeed; no effect on those in your text. Default: start on new page.
[NUMBER]	Numbers pages every 66 lines (or to match specified [FORM],[PAGE]). Default: no numbers. Specify if 66 lines do not equal one page.

[PAGESIZE=n]	Divide into **n** line page. Default = 55. Range of ([FORM]-[PAGE]): 2 to 100.
[PRINTER=n]	Selects printer **n** from file LPRS. If not specified, current printer is used.
[TABS=n]	Replaces tab (^I) characters with spaces, so that next character is printed in a column exactly divisible by **n**. Default = 8.

Note that [PAGE=n] resembles the [Pn] option in the TYPE and PIP commands, [PRINTER=n] resembles the PRINTER command, and [TAB=n] resembles the [Tn] option of the PIP and PRINT commands.

A single SPOOL PRINT command line can send several files to the printer, with blank spaces separating the various files in a list. Multiple modifiers are also accepted, and are applied to all the files in the command line. All of the modifiers must be enclosed by a single pair of brackets ([]), and must be separated by commas. For example,

```
4C>SPOOL TIC TOC TOE [MAR=10, PAGE=41, NUM, PRINTER=1]
```

This command prints three files (TIC, TOC, TOE) from the current user area on printer 1, with page numbering, with 41 lines per page and setting a margin ten characters to the right of the left-hand margin of the text. Note that "PRINT," the default SPOOL command, is omitted without ambiguity between SPOOL and the list of files. Other default values are 66 lines per page [no FORM], tab expansion to the eighth column [no TAB], begin each file after a formfeed [no NOF], one copy of the printout [no COP], and follow the linefeeds and pagebreaks that are indicated in the text [no FOR]. If the files to be printed were prepared on a word processor, these default format values are safe to use. If the height of your paper is different from 11 inches, or if your printer gets something other than six lines per inch vertically, you must do some calculating before you specify the [FORM] and [PAGE] values of a SPOOL PRINT command.

When SPOOL commands send several files to the printers, they are processed in the order that the queue for each printer received them. Another SPOOL command lets you look at the status of the spooler system:

```
6A>SPOOL STATUS
```

The screen will list the files being printed, jobs waiting to be printed, the file size, number of copies, and printer number.

The last SPOOL command that most users need to know about can remove a file from the queue before it is printed. The form of this SPOOL command is:

```
6A>SPOOL DELETE 10
```

This deletes from the queue of files not yet printed the job that the SPOOL STATUS command refers to as 10. If this job were being printed when you gave the SPOOL DELETE command, the printing would stop as soon as the printer buffer was empty. To avoid deleting the wrong job, it is good practice to give the SPOOL STATUS command before attempting to delete anything.

Two other commands, TERMINATE and RESET, have the effect of disabling the SPOOL command, and should be reserved for use by the system manager. The TERMINATE command saves a record of what was in the queues, so the spooler system could be restored after some problem was resolved that had provoked a TERMINATE command. The RESET command, however, obliterates the spooler system, so that any jobs waiting for printers are unrecoverably lost.

Installing the SPOOL Command

The system manager must follow certain procedures after the computer is rebooted for the SPOOL command to work properly. The SPOOL command is part of Digital Research Inc.'s **spooler system** for MC-DOS and DR Net that allows files to be sent to printer queues, rather than requiring that they be printed directly by other system commands (like PIP, PRINT, or TYPE) or by applications programs. The spooler system is normally configured at the time of system startup by commands within the SYSINIT.SUB file that the system manager creates. If the spooler system does not seem to be functioning, the system manager should type:

```
0A>SDIR ???.CMD
0A>SDIR SPOOL.CMD
```

to make sure that the files DSP.CMD, SPL.CMD, and SPOOL.CMD are all present on user area 0 of drive A:, and that these files have the SYS and RO attributes. If these files are not present, copy them (with the PIP command) from the system diskette, and if necessary SET them as SYS and RO files. Do not invoke the DSP or SPL commands, which are reserved for use by the spooler system. Do not erase any files with extension .DSP, which the spooler system creates and uses to manage the queues.

The SYSINIT.SUB file should include a command equivalent to:

```
0A>CHSET DSP.CMD [SHARED=ON]
```

to pass the [SHARED=ON] argument from the SUBMIT command line to a CHSET line in the SYSINIT.SUB file, so that all users can access a single copy of the DSP.CMD file. Otherwise the spooler system will use RAM memory inefficiently.

Initializing the spooler system requires a command like:

```
0A>SPOOL START 0  4  1
```

where after START is a list of the numbers (from the LPRS file) of the printers that will be accessible via the spooler system. Printer numbers range from 0 to 15. The above command means that printers 0, 1, and 4 can be accessed by the SPOOL command. These can include numbers corresponding to printers on other systems that will be accessible to users on your system via the network. A system must be configured for a number of printers within the range of numbers that follow your SPOOL START command. (Since the numbers start at zero, note that a "5-printer system" has printer numbers ranging from 0 to 4, not 5). The system will indicate on the screen if the START command is successful.

In the example above, printers 2 and 3 will not be supported by the spooler system. If you later decided to add more printers to the spooler system, you must first give the command:

```
0A>SPOOL TERMINATE
```

You can then repeat the SPOOL START command with a corrected list of printers. The SPOOL TERMINATE command interrupts all jobs in the spooler queues. If you give a SPOOL START command after a SPOOL TERMINATE command, the spooler system reprints from the beginning any interrupted job, and restores to the queues any files that were waiting to be printed when the TERMINATE command was given. You should issue a SPOOL TERMINATE command before you shut down the system, otherwise you will lose all of the contents of the queue.

Stopping the Spooler System

If a printer is printing useless output, a SPOOL DELETE **n** command will stop print job **n** (after a SPOOL STATUS command, to make sure that you specify **n** correctly). The SPOOL TERMINATE command shuts down the whole spooler system, but lets you restart it, with the queues intact, by a SPOOL START command and printer list. The SPOOL RESET command is a drastic way to stop the spooler system. Since it also erases all the queues, use this only if you had somehow lost control of a spooler system, and had no hope of successfully restarting it. Since any command that brings the spooler system to a halt tends to disrupt other users who are waiting for their jobs to print, users other than the system manager should probably restrict themselves to the SPOOL DELETE command, and should be very careful when using that.

Using the SPOOL Command with DR Net

Like a logical drive or a printer, a spooler system can be a network resource that all of the systems linked in the network can share. A system in a network can also have a spooler system that is not shared, but is accessible only to users on that system. Each system in a network can have only one spooler system, but if the system manager designates a spooler system as a network resource, users on other systems who connect to it can use SPOOL commands to send their files to its printers.

Individual users can modify the network by connecting or disconnecting spooler systems and their terminals. To do this, use the NET menu and select option 7 (to change spooler configuration). Equivalently, the NET SPOOL command can be used to attach a remote spooler system. After either procedure, SPOOL commands from your terminal to print files are executed by the remote spooler system.

Besides connecting your terminal to a remote spooler, DR Net also allows you to connect the printer of a remote system to your local spooler. To do this, use the NET menu and select option 1 (to change device assignment). Equivalently, the NET DEVICE command can be used to attach a remote printer. After either procedure, SPOOL commands from your terminal to print files are executed by the printer at the remote node. The system manager

has the option of using the CONFIG.NET file or network passwords to facilitate or to restrict the ability of network users to access other spooler systems or other printers.

Each virtual terminal in a DR Net network can be connected to no more than one spooler system. But different virtual terminals connected to different spooler systems can be supported by the same physical terminal. Provided that the system manager permits access rights to remote network resources, and provided that the operator can remember which virtual terminal is attached to which spooler system (the NET STAT display helps), a DR Net network linking several systems that each have spooler systems can provide great flexibility and scope to an operator who needs to produce printed output.

The menu-driven NET program is among several Gifford enhancements to MC-DOS that simplifies changing DR Net printers, queues, and spooler systems. See the DR NET USER'S MANUAL for details on the menu-driven NET program, accessing remote queues, the ONFIG.NET file, network security, and spooler systems within DR Net.

SUBMIT

SUBMIT d:**file** parameters

The SUBMIT command puts forth a SUBMIT file for batch processing of MC-DOS commands and programs. A SUBMIT file must have .SUB as its file type. The .SUB extension of a SUBMIT file is not entered in a SUBMIT command line.

SUBMIT files are created with a text editor, such as ED or WordStar in non-document mode. A SUBMIT file contains one line for each MC-DOS command with its possible options. A SUBMIT file on user area 0 of the system drive can be called by any user.

Examples:

3C>SUBMIT A:BILLING

This executes a SUBMIT file on user area 3, drive A: with the file name BILLING.SUB.

0A>SUBMIT WRKFILE

This executes a submit file on the current drive and user area with the file name WRKFILE.SUB. If a command line calls on an 8-bit program, then the SW (or SW86, on CPU 8086 systems) command must precede the 8-bit program file name. For example, if you want to run an 8-bit program called F80 on a file called BILLING.FOR, and then erase the file called BILLING.LST, your SUBMIT file would have the following lines:

```
A:SW F80 BILLING.FOR
ERA BILLING.LST
```

Up to ten (10) optional parameters can be included in a SUBMIT command line. These are arguments that are substituted in the command lines of your SUBMIT file, where they must be written as:

$0 $1 $2 $3 $4 $5 $6 $7 $8 $9

The symbol $0 is replaced by the name of the SUBMIT file itself. The symbols $1 through $9 are replaced by the first through ninth parameters respectively. For example, suppose that you had a file called AUTOPIP.SUB that consisted of the commands:

```
DSKRESET
PIP $1:[g$2]=*.$3[vrow]
ERA *.BAK
```

If you entered the SUBMIT command:

```
0A>SUBMIT AUTOPIP  F  6  DOC
```

then the first line of your SUBMIT file would be interpreted as

```
PIP F:[g6]=*.DOC[vrow]
```

On execution of the SUBMIT command above, all files of type .DOC would be copied to user area 6 of drive F: (provided that drive F: were ready, and that you had permission to access user area 6, and all of the files of type .BAK in your current user area would be erased. Similarly, the SUBMIT command:

```
12C>SUBMIT AUTOPIP  G  7  TXT
```

would copy files of type .TXT to drive G: user area 7. The $ notation lets you supply parameters to a general-purpose SUBMIT file, so that routine tasks can be accomplished with less effort than if each command were manually entered from the keyboard.

The MC-DOS control character ^P cannot be used in a .SUB file to attach or to detach a printer, but in a .SUB file you can reassign your current user area, printer, and/or logical drive. You can specify whether a change in assignment persists

after the SUBMIT file has executed, or reverts to your pre-SUBMIT values. To keep the new assigments, the .SUB file should include the special SUBMIT variable **$GLOBAL**, as a line **prior** to any commands that change current values. For example, suppose that you call a SUBMIT file that includes the lines:

```
$GLOBAL
C:
PRINTER 0
```

After this SUBMIT file has run, you will have C: as your current drive, and printer 0 as your list device, regardless of what your drive and printer were before the SUBMIT command. Any changes in user area, logical drive, or list device will revert to previous values if you type **$LOCAL** in place of $GLOBAL. Unless you specify $GLOBAL, $LOCAL is the default value of reassignment statements in a SUBMIT file.

A SUBMIT file can be "nested" within another SUBMIT file, but the word "SUBMIT" cannot appear within a SUBMIT file command line. To call a .SUB file from within a .SUB file, substitute the special command **$INCLUDE** for the word SUBMIT. For example, within a SUBMIT file the line:

```
$INCLUDE MARINE.SUB
```

is equivalent to typing SUBMIT MARINE at the system prompt, and causes the system to execute the commands in file MARINE.SUB.

The following is an example of a SUBMIT file. This example selects printer 1, reads a KEYS file called GCS950.KEY to optimize the terminal for spreadsheet use, selects user area 4 of drive B: as the current directory area, and displays a file called SPREADSH.EET that provides the user with information on what to do next. For this SUBMIT routine to execute correctly, it is necessary that the KEYS.CMD and SC950.KEY files be on user area 0 or 4 of drive B:, that the text file SPREADSH.EET be in user area 4 of drive B:, and that the user's account be authorized in the PASSWD file to access user area 4. The lines beginning with semicolons are comments that are not executed.

```
; Example of a SUBMIT file:
$GLOBAL
USER 4
B:
; User will see 4C> prompt after SUBMIT is completed.
PRINTER 1
; This selects printer 1 as the current list device.
KEYS SC950
; This file loads the special function keys of the console.
TYPE SPREADSH.EET
; This displays on the screen a file called SPREADSH.EET
```

Commands in a SUBMIT file are executed in the order in which they appear, "top to bottom." A command that cannot be executed will halt execution of the commands in the SUBMIT file. No more commands will be read from the SUBMIT file.

In some applications the user may wish to use the dollar sign ($) symbol as an actual command argument, rather than to symbolize a parameter supplied in the SUBMIT command line. To represent a "real" dollar sign in a .SUB file, type two dollar signs ($$). The SUBMIT program will interpret this as a single dollar symbol, and as unrelated to any SUBMIT command parameters.

The SUBMIT file can include file passwords (see the SET command). Notice, however, that including file passwords in a SUBMIT file would enable someone who could read the SUBMIT file to learn the passwords. To maintain security in this situation, the SUBMIT file could be kept in a protected user area, or a password could be put on the submit file itself, or passwords could be supplied as parameters in the SUBMIT command line, and symbolized only as $1, $2, etc. in the SUBMIT file.

Besides enabling the user to automate frequently used sequences of commands by SUBMIT files, the MC-DOS operating system recognizes the files SYSINIT.SUB, USERINIT.SUB, LOGIN.SUB, LOGOUT.SUB, and DOWN.SUB as sequences of commands to be automatically executed at certain times, if files with those names exist. See the separate descriptions of these files for more detail on when they are executed, and in what directory areas they should be located. All of these are .SUB files that the system manager or the individual user must create, rather than files distributed with the MC-DOS operating system.

SW

SW file

The SW command is used in SUBMIT files to process 8-bit programs. The SW program is the most innovative part of Gifford Computer Systems' enhancements to Concurrent DOS. This program enables an MC-DOS computer that has as its CPU (central processing unit) a Macrotech MI-286 or a CompuPro CPU 8085/88 dual processor to run 8- and 16-bit programs at the same time. The program SW.CMD must be on user area 0 of drive A:.

When you type in an MC-DOS command, the system first searches the directories for a Concurrent DOS (16-bit) program with a .CMD extension (see the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE for a discussion of the search order for files). If a .CMD file is not found, the shell automatically runs SW, which searches for a CP/M-80 (8-bit) program with the .COM extension.

If the .COM file is found, SW allocates a contiguous 64K block of memory. This block is loaded with the switching code (about 3K), and the .COM program. The program is then run in this transient program area of RAM.

If the .COM file is not found, SW gives the message:

Program file cannot be opened

If the system cannot find the SW.CMD file on user 0 of drive A:, it gives the message:

?Can't find command

When running CP/M-80 programs, SW processes all system calls in the 16-bit environment. It simulates CP/M-80 system calls so that programs run exactly as if they are in a CP/M-80 environment, except that the system calls are executed much faster. SW also handles MP/M II queue calls.

The SW command has to be specified in a submit file if the file is to call on 8-bit programs. If you want to run an 8-bit program called F80 on a file called BILLING.FOR, your SUBMIT file would have the following line:

```
A:SW F80 BILLING.FOR
```

The SW program enables the system to support 8-bit compatible CP/M programs, and also 8-bit MP/M II compatible programs. SW enables MP/M II queue calls, which differ from the queue calls of Concurrent DOS.

SW86

SW86 command arguments

The SW86 command is used in command lines and SUBMIT files to process 8-bit .COM programs on systems having the CPU 8086/87. The SW86 program causes the 8086 to function as a Z80 emulator. This program enables the system to run both 8- and 16-bit programs at the same time with a 16-bit only processor. The program SW86.CMD must be on user 0 of drive A:.

In order to run 8-bit programs on an 8086 processor system, you must type SW86 followed by the name (but not the .COM extension) of an 8-bit command file. The shell searches for a CP/M-80 (8-bit) program with the .COM extension. For example:

```
3B>SW86 WS
```

This allows an 8086 system to load and execute the 8-bit command file WS.COM, if this file is found in user area 0 of drive A: with the SYS and RO attributes. If the .COM file is found, SW86 allocates a contiguous 64K block of memory. This block is loaded with the switching code (about 3K), and the .COM program. The program is then run in this transient program area of RAM.

If the .COM file is not found, SW86 gives the message:

Program file cannot be opened

If the system cannot find the SW86.CMD file on user 0 of drive A:, it gives the message:

?Can't find command

When running CP/M-80 programs, SW86 processes all system calls in the 16-bit environment. It simulates CP/M-80 system calls so that programs run exactly as if they are in a CP/M-80 environment, except that the system calls are executed much faster.

The SW86 command has to be specified in a submit file if the file is to call on 8-bit programs. If you want to run an 8-bit program called F80 on a file called BILLING.FOR, your SUBMIT file would have the following line:

```
A:SW86 F80 BILLING.FOR
```

The SW86 program enables the system to support 8-bit compatible CP/M programs, and also 8-bit MP/M II compatible programs. SW86 enables MP/M II queue calls, which differ from the queue calls of Concurrent DOS.

SYNC

The SYNC command flushes the cache buffer onto the hard disk. This prevents loss of data in case of a power failure or a system reset. MC-DOS automatically flushes the cache buffer every 30 seconds, so under normal circumstances it is unnecessary to run SYNC.

SYSGEN

The SYSGEN program copies the system tracks from one 8-inch floppy diskette (the "source) onto another diskette (the "destination"). Unlike the CPY program, which must have physically identical source and destination diskettes, SYSGEN will accept different formats in the source and destination drives. Double-sided or single-sided diskettes can be used for source or destination, but both source and destination must be double-density (1024 byte sectors).

Gifford's MC-DOS operating system requires a **system diskette** to be in the left-hand floppy disk drive (if your system has two floppy drives), when the system is powered up, like the ignition key of a car. To avoid potential problems if a system diskette is damaged, SYSGEN or CPY should be used to make backup copies of the system diskette.

If your MC-DOS software is modified (to accommodate changes in your hardware configuration, or to install updates of the operating system software) the SYSGEN utility can be used to create new system diskettes. A special portion of every diskette is reserved for the "system tracks," which cannot be read or modified by any program or command except SYSGEN, CPY, and the destructive FORMAT program. SYSGEN and CPY are able to write or copy files on the reserved part of a diskette.

SYSGEN can be used to copy the system tracks from a system diskette to another diskette; or to copy the system tracks to memory, so that a computer with only one floppy diskette can create a system diskette by copying the system tracks back from memory to a diskette. Readers with only one floppy disk drive cannot use the CPY program to create a system diskette.

To use SYSGEN, place a system diskette that you intend to use as the source in the left-hand floppy drive, and put a newly formatted blank diskette in the right-hand floppy drive, if you have two floppy drives.

Type:

0A>SYSGEN

and the program responds with the following:

GIFFORD COMPUTER SYSTEMS
System Track Maintenance Program Rev 1.1
Command Menu Options:

C = Sysgen disk to disk.
O = Single drive sysgen.
F = Sysgen file to disk.
X = Exit to the system.

Option:

If you have a system with two floppy disk drives, type (C) (without RETURN). If you have a system with only one 8-inch floppy drive, type (O) (with no RETURN).

If you selected option C, the program responds:

Enter source drive:

Type the logical drive designation for the source drive (without pressing RETURN) and the program will respond:

Enter destination drive:

Type the logical drive designation for the destination drive (without pressing RETURN) and the program will respond:

Press any key to continue sysgen or control-C to abort.

If you type any key (other than ^C) to continue, the program then copies the system tracks onto the destination diskette. When the process is complete, the program responds:

Function complete.
Press any key to continue.

Use option F to use the boot file and the loader file to generate system tracks. Type (F) without pressing RETURN from the command menu and the program responds:

Enter boot filename:

The answer to this prompt should be BOOT.LDR. The program then prompts:

Enter loader filename:

There are two possible loader file names: DISK1.LDR and DISK1A.LDR. Your response to this prompt depends on the type of floppy disk controller that your system supports.

The program responds:

Enter destination drive:

Type the drive destination for the destination drive without pressing RETURN and the program will respond:

Press any key to conntinue sysgen or Control-C to abort.

If you type any key to continue, the program then copies the system tracks onto the destination drive. When the process is complete, the program responds:

Function complete.
Press any key to continue.

If you select option O for a single drive SYSGEN, the program responds:

Enter source/destination drive:

Type the drive designation for the source/destination drive (without pressing RETURN) and the program will respond:

Press any key to continue sysgen or control-C to abort.

If you type any key to continue, the program responds:

Exchange disks, press RETURN to continue or control-C to abort.

The program then copies the system tracks onto the destination diskette. When the process is complete, the program responds:

Function complete.
Press any key to continue.

The system tracks have been copied, but the file MC.DOS must be on the diskette before it can be used to boot the system. Use PIP to copy MC.DOS and any additional files that you want to add to the new system diskettes.

SYSINIT

SYSINIT.CMD is an optional program file which will run when you start up the system. This program file can be used to customize your system by executing a program which must be part of your startup procedure. SYSINIT.CMD has limited application on most systems. The file SYSINIT.SUB will have greater value to most users. Refer to **Section 2.5-SYSINIT.SUB.**

The maintenance of this file falls under the jurisdiction of the system manager. It is discussed again in **Chapter 3.**

SYSINIT.BAN

SYSINIT.BAN is an optional banner file. The contents of the file are displayed above the Name: prompt before a user logs in to the system. The banner appears again after the user logs out of the system. The SYSINIT.BAN file must exist in user area 0 of drive A: for the banner message to be displayed.

You can create your own company banner using ED or WordStar in non-document mode. An example banner file might look like:

-[Y

GIFFORD COMPUTER SYSTEMS WOULD LIKE TO
WELCOME YOU TO THE WORLD OF THE
MC-DOS OPERATING SYSTEM

The -[Y characters on the first line of the file are commands to clear the video screen. This allows the message to appear at the top of the screen. You should refer to the documentation on your terminals to see what escape sequence is used to clear the screen.

The maintenance of the SYSINIT.BAN file falls under the jurisdiction of the system manager. It is discussed again in **Chapter 3.**

Each computer within a network can have a different SYSINIT.BAN file. If a NETMOTD file exists on user area 0 of drive A: of the system node, that banner is displayed above the SYSINIT.BAN message on all terminals on the network.

SYSINIT.SUB

SYSINIT.SUB is an optional file containing commands which are to be executed as part of the startup procedure. This will be a useful file if you have initialization procedures you want to execute on startup. SYSINIT.SUB should be located on user area 0 of drive A:.

The following example shows how you might use SYSINIT.SUB to initialize the spooler system for printers 0 and 1. The contents of the file are as follows:

```
; Sample SYSINIT.SUB file for initializing the spooler.
SPOOL START 0 1
```

It is not necessary to include CCPMINIT in the SYSINIT.SUB file to implement virtual terminals. This is because CCPMINIT is run automatically if it exists on user area 0 of drive A: when an MC-DOS system is powered up or reset.

The maintenance of SYSINIT.SUB falls under the jurisdiction of the system manager. It is discussed again in **Chapter 3.** See the description of the SUBMIT command for details on .SUB files in MC-DOS.

SYSTAT

SYSTAT [Option]
SYSTAT [Option C ##]

The SYSTAT command displays information about the internal status of your MC-DOS system. SYSTAT is used to identify what system and user processes are running, what physical terminals, virtual terminals, and printers are attached to them, their current state of activity, their size and location in memory, their foreground or background status, what queues exist, and other characteristics of the system. Type SYSTAT and you see:

```
Which Option ?

        H Help
        M Memory
        O Overview
        P Processes - All
        Q Queues
        U User Processes
        C Consoles
        X Exit

      ->
```

Like AP, CLOCK, NET, NEWUSER, and the TIMELOG utility, SYSTAT can function as an interactive program whose menu prompts you to indicate by a keystroke the type of information that you seek. Like NET it can also function in command mode, so that the experienced user can give unprompted options and modifiers in a single command line that produces the desired display.

The SYSTAT display provides a much more detailed description of the system than do the CONSOLE, PRINTER, or WHO utilities, but for most users the SYSTAT description is so voluminous and technical that it may not seem very informative. In fact, SYSTAT is not an ordinary system command intended for non-technical users, but a programming and diagnostic tool like the ASM86 and DDT86 routines. It is not discussed in the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE (where the system commands are

described), but in the CONCURRENT CP/M PROGRAMMER'S REFERENCE GUIDE, which is not light reading. Why did Digital Research Inc. write a menu-interface for this command, so that novice users who are struggling to use PIP can easily use SYSTAT?

In a word, diagnostics. The right SYSTAT command can sometimes be very informative when something seems to be wrong. For example, suppose that a printer suddenly won't work. If it is not out of ribbon or paper, try looking at the problem with SYSTAT. Type the "SYSTAT" keyword, and when the menu is displayed, select option C ("Consoles"). Press RETURN when the program asks if you want a continuous display, since that will not be necessary. The screen will show a display that begins:

PHYSICAL CONSOLE	VIRTUAL CONSOLE	NAME	STATE
00	00	Shell	BACK,BUFF
00	01	WS	FORE,BUFF
00	02	Login	BACK,BUFF
00	03	WS	BACK,BUFF
01	04	Shell	BACK,BUFF,^P
01	05	dBASE	FORE,BUFF
01	06	Login	BACK,BUFF
01	07	Shell	BACK,BUFF
02	08	SYSTAT	FORE,BUFF
02	09	Shell	BACK,BUFF
02	0A	Login	BACK,BUFF
02	0B	Shell	BACK,BUFF
03	0C	WS	FORE,BUFF
03	0D	Shell	BACK,BUFF
:	:	:	:

The mystery is solved: the number 4 virtual console has attached itself to a printer with a control character ^P (shown at the far right), and then switched to background mode to allow dBASE to run from the same physical console. To free the printer, the operator must switch virtual console 4 to the foreground and type ^P again to toggle it OFF. Now other users can access that printer.

To use SYSTAT in command-driven mode, supply the same letter prompts that appear in the menu, but enclose them in a pair of square brackets ([]). Several options can also accept a "C" argument, to refresh the display every ## seconds, for ## some hexadecimal number. SYSTAT in command mode seems to be the only MC-DOS command that cannot be executed if the user fails to supply a matching right bracket (]) as the last character in the command line.

Besides the special uses of SYSTAT already described, SYSTAT may be helpful if you ever experience trouble with your system, since your dealer may ask you to select SYSTAT options and describe the display. Readers who use their system primarily for software development should study the CONCURRENT CP/M OPERATING SYSTEM PROGRAMMER'S REFERENCE GUIDE for more information on the SYSTAT command, whose features and applications are beyond the scope of this manual.

TIME

TIME command
TIME SW command

The TIME command measures how long it takes for a program to run. TIME can run with any program or SUBMIT file. For example, to time the file BILLING.CMD, give the command:

```
2B>TIME BILLING
```

The TIME program will display on your screen the elapsed time:

```
Elapsed time=00:04:29
```

To time an 8-bit program, you must precede it with an "SW" on systems based on the CPU 8085/88 or MI-286 dual processors), or with an "SW86" for systems based on the CPU 8086/87. Ask your system manager if you do not know what kind of CPU your computer has. In a DR Net local area network that includes several types of computers, the choice of SW or SW86 depends on whether your local node has an 8088. For example, suppose that you are working on a computer with an 8085/88, and you want to find out how long it takes the 8-bit program DBASE.COM to run the program FILER.CMD. You should enter the command line:

```
2B>TIME SW DBASE FILER
```

The TIME utility is helpful in "benchmarking" programs. It can assist the system manager in determining how different MC-DOS configurations or different operating conditions affect the speed of the system.

TIMELOG.CMD

The TIMELOG.CMD file allows you to use dBASE II to report the access of specific accounts and terminals. To use this program, you must be in user area 0 of drive A:. Give the command:

```
0A>DBASE TIMELOG
```

The screen displays a menu of reports that you can produce:

```
Here are your options for timelog reports:

 1 = Report access by a specific account and specific device
 2 = Report access by a specific account, all devices
 3 = Report access by a specific device, all accounts
 4 = Report access by all accounts & devices, subtotal by device
 5 = Report access by all devices & accounts, subtotal by account
 6 = Return to the operating system

Input your choice : :
```

Account names are the login names that appear as the first entry in the password file. To specify a device, give a device name that appears in the fifth field of the TTYS file (usually 0 through 6). Enter a number (1 through 6) to select your option.

Each time you run a report, TIMELOG asks you for a file name for the report. Once the report is finished, you can TYPE the report on the screen, send it to the printer with the PRINT command, or edit it with ED or WordStar.

TIMELOG.SUB

The TIMELOG.SUB file is used to maintain the TIMELOGX.DBF database. TIMELOGX.DBF is a dBASE II database used to prepare records based on the file WTMP. Use TIMELOG.SUB to create the database or to append the database if it already exists. Give the command:

```
0A>SUBMIT TIMELOG
```

This program copies all entries from the WTMP file, writes them to the database, erases the WTMP file, and restarts the WTMP file. Perform this step when there are no other users on the system, so that your database is complete.

Use dBASE II to view the TIMELOGX.DBF file. You can use the utilities in the dBASE II command file called TIMELOG.CMD to prepare TIMELOG reports. To use dBASE II on TIMELOGX.DBF you must be in user area 0.

TTYS

The TTYS file describes characteristics of each terminal or modem used on the system.

An example of the TTYS file is:

```
0:9:0:82n:console:System support
1:9:0:82n:tty1,front:Interfacer 3 port 0
2:9:0:82n:tty2,sales:Interfacer 3 port 1
```

Entries in the file have the following format:

```
DEV #:BAUD RATE:HANDSHAKING:PROTOCOL:TTY NAME:COMMENTS
  \        \           \          \       \        \
   1        2           3          4       5        6
```

1. **Number of the device,** corresponding to the connector label on the back panel. Numbers begin with 0.

2. **Terminal baud rate;** the default is 9600. Baud rates correspond to the following:

   0 = 9600
   1 = 110
   2 = 300
   3 = 600
   4 = 1200
   5 = 1800
   6 = 2400
   7 = 4800
   8 = 9600
   9 = 19200
   10 = automatic baud rate detection for modems.

The automatic baud rate detection for modems only works for 300 and 1200 baud. To use this feature, pin-20 on the RS-232 connector must be wired to the speed detect pin on the modem.

3. **Optional RS-232 hardware handshaking.** Hardware handshaking is not required for most terminals; the default is 0. If handshaking is necessary, enter the number corresponding to the type of handshaking needed from the following:

   0 = No handshaking
   1 = Hardware handshaking (DTR) (RS-232 pin-20)
   2 = Software handshaking (XON/XOFF)
   3 = Reserved for custom application

4. **Optional communications protocol.** This is a three character entry. The default protocol is 8 data bits, 2 stop bits, and no parity, which are the settings for most terminals. If you need to change this, select the configuration you need from the table below and enter the three characters without any spaces. For example, if your terminal requires a protocol of 7 data bits, 2 stop bits, even parity, enter 72E in field 4.

Data bits	Stop bits	Parity
8	2	N (none)
7	1	O (odd)
		E (even)

5. **Terminal name** of up to 8 characters with no spaces. The terminal name is often entered as "console" for the system console, "ttyl" for terminal 1, "tty2" for terminal 2, etc. This field is used to identify the location on the system of a user in many MC-DOS utilities. Many terminal names can be listed, separated by commas with no spaces.

6. **Optional comments field.** The comments field can give information of potential value to users, such as the location and RS-232 port.

In the example TTYS file on the previous page, terminal 0 (the system console) communicates at 19200 baud, uses the standard communications protocol (8 data bits, 2 stop bits and no parity), and has the terminal name "console." You could change the names to correspond with the physical location of the terminals. This is handy if you have to use the ABORT process to free a "locked" terminal.

The TTYS file can be modified while the system is running in multiuser mode. Once the file has been modified, type:

```
0A>RESET
```

This will reset the system to the new specifications without the need to shut down the system.

The number of terminals and modems that can be supported by an MC-DOS system is specified in the XIOS (extended input/output system). If a system is reconfigured to support a greater number of terminals and modems than was known to Gifford when the system was ordered, it may be necessary to install a larger XIOS. Read the Gifford publication CUSTOMIZING YOUR MULITUSR CONCURRENT DOS SYSTEM for details on how to modify the XIOS.

The maintenance of this file falls under the jurisdiction of the system manager. It is discussed again in **Chapter 3.**

TYPE

TYPE d:**file**, d:file, ..., d:file [PAGE]
TYPE d:**file(s)** [Pn]
TYPE d:**file(s)** >d:file
TYPE d:**file(s)** **@node** >d:file

The TYPE command displays ASCII (printable) files on your terminal screen. TYPE can be used to look at the contents of text files or data files, either locally or across a DR Net network. With the > option, TYPE combines a list of files without displaying them, creating a file whose name follows the > symbol. Used in conjunction with the MC-DOS control character ^P, it can send files to your current printer. For example, the command:

```
0A>TYPE VTTYS
```

followed by the control character ^P before pressing RETURN, lists your VTTYS file on the terminal screen and also causes your current printer to print the VTTYS file. This procedure will not take advantage of special printer control commands (such as pitch control or boldface) offered by word processing programs such as WordStar. The printer will continue to echo the screen of your terminal until you type another ^P.

You can TYPE a file (or a list of files) that is on another logical drive. For example:

```
7B>TYPE A:PART1.TXT, A:PART2.TXT, C:PART3.TXT, PART4.TXT
```

This example will display (without pausing between files) the files PART1.TXT and PART2.TXT from user area 7 of drive A:, the file PART3.TXT from user area 7 of drive C:, and the file PART4.TXT from user area 7 of drive B:. The commas in the example above are optional, but a blank space must separate each file name in a TYPE command to type a list of files.

To display files in other user areas, specify the user number either before the colon (:) or before the drive letter. Thus, "5C:" or "C5:" denotes "drive C:, user area 5." For example,

```
7B>TYPE A0:PASSWD
```

displays the PASSWD file on user area 0 of drive A:.

TYPE with the > symbol can be used instead of PIP to copy a file or list of files to a file whose name follows the > symbol. This option does not display files on the screen. For example,

```
5C>TYPE DASHER DANCER DONNER BLITZEN >B:REINDEER.DOC
```

creates a file called REINDEER.DOC in user area 5 of drive B: that combines four files from the current user area in the order listed (DASHER first, BLITZEN last). This has the effect of:

```
5C>PIP B:REINDEER.DOC=DASHER DANCER DONNER BLITZEN
```

Wildcard characters (*, ?) can be used in the file specification of a TYPE command line to display all the files that match the specification. Wildcards can also be used with the > option to combine files. To display all files of type .DOC in user area 6 of drive B:, you can use the command line:

```
6A>TYPE B:*.DOC
```

Besides the > symbol, TYPE accepts two options, [PAGE] and [Pn]. If neither is specified in a TYPE command line, the display scrolls continuously. The scrolling can be stopped with ^S, continued with ^Q, and abandoned with ^X or ^C.

The [PAGE] option displays 24 lines, and then waits until any key is pressed to continue. The [Pn] option displays **n** lines at a time before pausing for a keystroke to continue. There must be a space between the option and the last file name. If a TYPE command to display a list of files ends with either option, the option applies to every file on the list. For example:

```
8C>TYPE DRAFT.DOC FINAL.DOC [P48]
```

This displays DRAFT.DOC and FINAL.DOC from the current user area, pausing every 48 lines for a keystroke to continue.

Displaying Files on Other Nodes of a DR Net Network

To use the TYPE command to display a file over a DR Net network, your terminal must be connected via the network to the remote drive whose file you intend to display. For example, suppose that you want to display a file called TEST.DOC which is located on user area 2 of drive C: on the SALES system. You must first connect to drive C: on the SALES system by typing:

```
0A>NET CONNECT @SALES
```

You can now use the TYPE command to display the file called TEST.DOC that is stored on user area 0 of drive C: of the SALES system by typing:

```
0A>TYPE TEST.DOC@SALES
```

Note that the TYPE command with MC-DOS control character ^P cannot be used to display a file on a printer attached to a different system within a DR Net network.

The TYPE command described in the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE is different in several ways from TYPE in Gifford's MC-DOS.

UPTIME

UPTIME @node

The UPTIME command displays the amount of time that a system has been up and running since last being rebooted. The UPTIME command can be run locally and across the network.

If you type:

0A>UPTIME

from the local system called "service," the system responds with:

Service has been up for 1 + 02:10

That is, "one day, two hours, and ten minutes."

To check the uptime for a particular node across the network, type UPTIME @, followed by the name of the remote node. For example, if you type:

0A>UPTIME @SALES

The system might respond with the following display:

Node sales has been up for 03:48

To get a listing for all of the systems attached to the network, type the following:

0A>UPTIME @

An example of a possible display from this command is presented below:

```
Node SALES-2  down     21:26
Node ACCOUNTI down   3+07:22
Node MIS        up     02:47
Node EDUCATIO   up     10:15
Node SERVICE    up     12:24
Node MARKETIN   up   2+02:07
Node SALES      up     03:50
```

The UPTIME command provides a tool with which the system manager can monitor system operations.

USER

USER n

The USER command will change the current user number to **n**, where **n** is an integer with value between 0 and 15. For example:

```
0A>USER 6
```

This returns the prompt:

```
6A>
```

If you type the keyword **USER** without indicating a number, the system will prompt:

```
Enter new user number:
```

The USER command cannot be used to access user areas that are not specified in your login account in the PASSWD file. For example, if you try to access restricted user area 6 from unrestricted user area 5, the screen will display:

```
You don't have permission to access user area 6.
User number = 5
```

The USER command can be avoided by typing the new user number followed by a colon. For example, to move from user area 6 of drive A: to user area 5 of drive A:, type the following command:

```
6A>5:
5A>
```

USERINIT

USERINIT.CMD is an optional program which will run when a user logs in to the system. This program file can be used to customize your system by executing a program which must be part of the login procedure. USERINIT.CMD has limited application on most systems. The USERINIT.SUB file will have greater value to most users.

The maintenance of this file falls under the jurisdiction of the system manager. It is discussed again in **Chapter 3.**

USERINIT.SUB

USERINIT.SUB is an optional file containing commands which are to be executed as part of the login procedure. A sample file, USERINIT.SAM, is supplied with the system. The sample file has the following entries:

```
; Sample USERINIT.SUB
; This SUBMIT file runs automatically each time a user logs in.
; This file types out mail (if any), and lists the .BAK files
; to remind users to delete them.
MAIL -A
DIR *.BAK
```

The maintenance of this file falls under the jurisdiction of the system manager. It is discussed again in **Chapter 3.**

UTMP

The UTMP file is automatically created when you boot the system. Booting the system also automatically puts a special password on the UTMP file. Only the operating system itself can use this password, which prevents any person or program but the operating system from modifying the UTMP file.

If you do not wish to restrict access to the UTMP file, simply use the SET command to turn PROTECT OFF on the label of drive A:. This will disable password protection on all of the files of drive A:. Once in use, the UTMP file has the format:

ttyl 2david 12 16642132392034 7221982

The first three fields are the physical terminal name, the relative virtual terminal number, and the account name. The other fields are:

```
12 1664 21 32 39 203 4 7 22 1982
  \\    \  \  \  \   \\
   1 2    3  4  5  6   7 8
```

1. **Virtual** terminal number.

2. **Days** since January 1st, 1978.

3. **Hour** of the day (24 hour clock).

4. **Minute** of the hour.

5. **Second** of the minute.

6. **Julian day** of the year (1-366)

7. **Day** of the week (Sunday = 0, Monday = 1, Tuesday = 2, etc.)

8. **Date** in month-day-year format.

The UTMP file is discussed in more detail in **Chapter 3.**

VCMODE

VCMODE
VCMODE HELP
VCMODE BUFFERED
VCMODE SIZE=n
VCMODE DYNAMIC
VCMODE FLUSH

The VCMODE program allows you to change dynamically certain background operating characteristics of your virtual terminals. The VCMODE command only affects the virtual terminal that is in the foreground at the time when you give the command, but it specifies how the operating system will treat output from that virtual terminal if it is later switched to the background (that is, when it is not attached to your physical screen).

The VCMODE program allows you to select the way that each of your virtual terminals will handle characters output by background processes (programs attached to virtual terminals that you are not currently monitoring). The VCMODE command allows you to specify that output from a background virtual terminal is treated in any of three ways:

(1) BUFFERED, or saved in a disk buffer for display when you reattach the virtual terminal to the physical screen;
(2) DYNAMIC, or displayed on your screen as it is produced, while another process runs in the foreground;
(3) FLUSH, or discarded.

The BUFFERED mode saves the first **n** kilobytes (**n** times 1024 characters) that any application program attempts to output to a background screen, and stores those characters in a temporary buffer area of your hard disk. You can specify the maximum size of the buffer area with the SIZE=n option of the VCMODE program. To cause the current virtual terminal screen to operate in BUFFERED mode, type the command:

```
4B>VCMODE BUFFERED
```

To set the maximum size of the buffer to 16 kilobytes, type:

```
4B>VCMODE SIZE=16
```

If you invoke a program and then switch your screen to another virtual terminal, up to 16K of background output (in this example) could be sent to the disk buffer. If this buffer fills up, your background program will pause as soon as it tries to send an additional character to the buffer. When you switch back to a BUFFERED virtual terminal that has sent screen output to the disk, any characters stored in the disk buffer are immediately sent to your screen, and the program resumes execution. The temporary buffer files are automatically erased from the disk after BUFFERED output is displayed to the screen. The MC-DOS control characters ^S and ^Q, described on **page 2.3-2**, can be used to control the scrolling of screen output, or the ^P control character can be used to list the output on your current printer.

You must be in BUFFERED mode to use the SIZE option, and there must be enough space on logical drive A: for the buffer. (Note that the virtual terminal buffers controlled by the VCMODE program are not directly related to the MC-DOS cache buffer.) The default mode for output from background virtual terminals is BUFFERED, the mode appropriate for most business applications.

In some programming and technical applications, however, it is helpful to see output in real time from a background process. The DYNAMIC option of VCMODE allows you monitor screen output as it is produced by a detached program, without concern that the background process will pause if your disk buffer becomes full. A background virtual terminal in DYNAMIC mode sends character output directly to the screen, rather than to a disk buffer. The following example shows how to select DYNAMIC mode for your current virtual terminal:

```
9A>VCMODE DYNAMIC

Background Mode for Virtual Terminal n set to Dynamic with Output
```

If you invoke a program from a virtual terminal whose VCMODE status is "DYNAMIC mode with output," the character output will be displayed on screen, even if you switch to another virtual terminal. That is, even though your keyboard and screen are attached to a different foreground virtual terminal, a program running in the background on a DYNAMIC virtual terminal can still write to your physical screen.

Since output from two programs can be confusing if you are running a visually-oriented foreground program like a spreadsheet or word processor, VCMODE also provides a FLUSH option to suppress screen output from a background virtual terminal. For example:

```
5C>VCMODE FLUSH

Background Mode for Virtual Terminal n set to Dynamic with Flushing
```

This differs from the SUSPEND option of the CHSET command (which makes a background process pause until reattached to the screen) in that FLUSH allows a background process to run to completion.

With the FLUSH option, characters sent to the screen by a background process go neither to the screen nor to a disk buffer, but are discarded, and go nowhere. The FLUSH mode prevents a background virtual terminal from sending characters to the screen, even when you bring that virtual terminal back to the foreground. When you return to a virtual terminal that is operating in FLUSH mode, the screen will not display any output, but will only show what was on the screen prior to switching to the background mode.

The HELP option of VCMODE displays information about the VCMODE program. If you type:

```
6B>VCMODE HELP
```

your screen will display a general description of the VCMODE command, including all of VCMODE's primary command arguments.

If you type the keyword VCMODE without an argument, as in:

```
8B>VCMODE
```

your screen will tell you the background mode of your current virtual terminal. If its status is BUFFERED, for example, the program will show the size of the disk buffer:

```
Background Mode for Virtual Terminal   0 is Buffered
Maximum file size =     10K
```

Because each virtual terminal is independent, a VCMODE command given from one virtual terminal cannot change the status of another virtual terminal, even if both are supported by the same physical terminal.

When you use VCMODE to select the background screen output mode of a virtual terminal, that mode persists until you change it with another VCMODE command, or until the MC-DOS system is reset or powered down. Neither switching the electrical power of a physical terminal OFF nor entering a LOGOUT command will restore a DYNAMIC virtual terminal to the default BUFFERED mode.

The VCMODE command in MC-DOS has DYNAMIC and FLUSH functions that are different from those of the standard Digital Research VCMODE command. The description of the VCMODE DYNAMIC option in the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE does not apply to your MC-DOS system.

VT950.KEY

VT950.KEY is a sample file included on the Virtual Terminals Master Diskette that allows you to program function keys for use with the VT program. VT950.KEY is a function key template to switch VT screens for TeleVideo 950 or Freedom 200 terminals that have been upgraded with extra pages of internal RAM. The VT950.KEY file that is on your master diskette is written to program function keys one (1) through four (4) to switch directly to the first through fourth VT screens respectively.

The summary below describes the function keys that the VT950.KEY file programs to cause VT to switch screens.

VT950.KEY: KEYS template for TeleVideo 950 and Freedom 200

Function Key	Character Sequence	VT Function
F1	(^_) 0	Switch to first VT Screen
F2	(^_) 1	Switch to second VT Screen
F3	(^_) 2	Switch to third VT Screen
F4	(^_) 3	Switch to fourth VT Screen

To program your TeleVideo 950 or Freedom 200 function keys using this template, type:

```
0A>KEYS VT950
```

After you enter the above line from your terminal, pressing key F1 will switch you to the first VT screen, F2 to the second, F3 to the third, and F4 to the fourth. If for any reason your terminal is turned OFF, your TeleVideo function keys will be cleared and you must run the KEYS program again. Rebooting the system does not by itself require you to run the KEYS program again unless your terminal was turned OFF in the process.

See Section 3 of the VIRTUAL TERMINALS USER'S MANUAL for additional information.

VTGCS80.KEY

VTGCS80.KEY is a sample file included on the Virtual Terminals Master Diskette that allows you to program function keys for use with the VT program. VTGCS80.KEY is a function key template to switch VT screens for the GCS-80 (CIT-80) terminal that has been upgraded with extra pages of internal RAM. The VTGCS80.KEY file that is on your master diskette is written to program the top four keys on the 18-key pad, left to right, to switch to virtual terminal screens one through four respectively.

The summary below describes the function keys that the VTGCS8.KEY file programs to cause VT to switch screens.

VTGCS80.KEY: KEYS template for GCS80 terminal

Function Key	Character Sequence	VT Function
COL MODE	(^_) 0	Switch to first VT Screen
COPY BLOCK	(^_) 1	Switch to second VT Screen
WRITE BLOCK	(^_) 2	Switch to third VT Screen
BLOCK DEL	(^_) 3	Switch to fourth VT Screen

To program your GCS-80 function keys using this template, type:

```
0A>KEYS VTGCS80
```

After you enter the above line from your terminal, pressing the COL MODE key will switch you to the first VT screen, COPY BLOCK to the second, WRITE BLOCK to the third, and BLOCK DEL to the fourth. If for any reason your terminal is turned OFF, your function keys will be cleared and you must run the KEYS program again. Rebooting the system does not by itself require you to run the KEYS program again unless your terminal was turned OFF in the process.

See the VIRTUAL TERMINALS USER'S MANUAL for additional information.

VTTYS

The VTTYS file is an ASCII file, much like the TTYS file, that tells the Virtual Terminals program what terminals on your system can support virtual terminals and describes some technical characteristics of these terminals. The VTTYS file contains a series of lines, each of which describes one physical terminal intended for use as virtual terminals. Every physical terminal with additional screens must have an entry in the VTTYS file in order to be supported by virtual terminals. If the terminal does not have an entry in the VTTYS file, that terminal will be treated as a standard, single-page terminal. The VTTYS file is read by the CCPMINIT.CMD program during the boot sequence, and must reside on user area 0 of drive A: in order for Virtual Terminals to operate.

The VTTYS file must conform to the physical terminals on your system. If you add or switch physical terminals, your VTTYS file must be modified to reflect the change.

The master diskette contains a VTTYS.SAM sample file which contains lines that can be used as a model in the initial creation of the VTTYS file. See the VIRTUAL TERMINALS USER'S MANUAL, Section 1, for additional information.

WHO

WHO @node
WHO accounts
WHO -option ...
WHO ~terminal
WHO !program

The WHO command lists the account names of users currently logged in to each system attached to the network. This list of account names also includes the terminal name, the virtual terminal number ("con"), the current program, the time that each account logged in, the amount of time that each user has been logged in to the system, and the amount of time that has elapsed since the last keystroke was made on the foreground virtual terminal. For the WHO command to work, the UTMP file must be on user area 0 of drive A:.

If you type WHO, a listing similar to the one below will be displayed on your screen for the accounts currently logged in to the local system:

Wed Jun 20, 1984 2:54 pm
Total of 5 users logged in

name	terminal	con	program	time	period	idle
nancy	console	00	*	9:22 am	5:31	0:02
tom	tty1	04	*WS	9:36 am	5:17	
trish	tty2	09	*WHO	9:01 am	5:52	

Listed below is a definition of each of the columns of information displayed above:

name: The account name used to log in to the system.

terminal: The physical terminal's name from the TTYS file.

console: The virtual terminal number. This number is required by the ABORT program.

program: The program that the user is currently running. The asterisk that is displayed before the program name indicates that this is the foreground terminal. If there is no program name next to this asterisk, this means that the user is currently not running any program. The terminal is most likely at the MC-DOS prompt.

time: The time of day that the user logged in to the system.

period: The amount of time that the user has been logged in to the system. If this field is empty, the WHO program was unable to calculate this time period.

idle: The amount of time that has elapsed since the user last entered a keystroke on the terminal. This is useful for checking to see if someone is at their terminal before you use the WRITE program to send them a message. To check for users currently running a program if you plan to shut down the system. If the idle field is empty, this means that the user has been idle for less than one minute.

There are five options that can be used in conjunction with the WHO program. They are:

1. @ node
2. accounts
3. - option
4. ~ terminal
5. ! program

These commands and the arguments that can be used in conjunction with them are described below.

1. @ node

The at-sign (@) can be used to display information about any system that you specify after you type (@) on the command line. For a listing of timelog information for all accounts currently logged in to all systems attached to the network, type WHO @ without specifying a system.

To find out information about who is currently logged in to a particular system, type (@) followed by a system name. Make sure that you don't enter a space between the "@" and the system name that you intend to use. For example, to find out information about the Sales Department, type:

```
0A>WHO @SALES
```

and you will get a listing of all accounts currently logged in to the SALES system.

2. accounts:

You can also use the WHO program to display information about a particular account by entering WHO and the name of the account that you intend to monitor. For example, if you type:

```
0A>WHO NANCY
```

the screen will display a list of the various terminals, consoles, and programs which that account has accessed. This display will also give the time these were accessed, as well as the duration of the access.

To find out information about several accounts on several systems, you can use the @ option in a command line. For example, to find out information about dale, janet, and pat on the systems named DALE, SALES, and SERVICE, type:

```
0A>WHO DALE JANET PAT @DALE @SALES @SERVICE
```

The WHO program will look for the account names that are listed on all the systems that are listed. A listing of all the computer activity for those accounts will be displayed on your screen.

3. - options:

Options are used to change the way that a program processes information. For the WHO program, single-character options can be used to change the display format. Program options will be read by the WHO program in the order that they were entered on the command line. Listed below are the various options that can be used to change the display format of the WHO program:

-a (for ALL virtual terminals)
-d (for DATE of login time)
-h (24-HOUR clock, instead of 12)
-r (REVERSE the order of the listing)
-s (for time in SECONDS, as well as in minutes and hours)
-v (VERSION number of your WHO program)

-a all:

By typing WHO -a, you will get a listing for every virtual terminal that each account is currently logged in to, instead of just the foreground process.

The foreground processes are marked by the asterisk that precedes the name of the program that each account is running. This option is useful in a DR Net network if you are shutting down the system, and you want to make sure that all users on the system have logged out of each of their virtual terminals.

-d date:

This option will allow you to list the date when each account logged in to the system, as well as the time. This would be useful if someone regularly forgets to log out of their system, and you want to determine the last day that they logged in. In order to display this information type:

```
0A>WHO -d
```

-h hours:

The WHO program allows you to list the login time for each account in a 24-hour clock format, instead of the 12-hour clock format (which uses am and pm) to which the WHO program defaults. To do this type, WHO -h, and this information will be displayed on your screen.

-r reverse:

The WHO program normally displays its information in alphabetical order, based on the account name. The -r option reverses the order of this display.

-s seconds:

By typing WHO -s, all of the timelogging categories (time, period, and idle) will list the seconds, as well as the hours and the minutes.

-v version:

This option will print out the current version number of your WHO program, along with the normal WHO information for the local system.

Listed below are the various options that can be used in conjunction with the WHO program to change the sorting format of the WHO program. The display is normally sorted by account name only. You can cause the display to be sorted by up to three categories by specifying one of the sort options below:

-@ (sort by system name)
-i (sort by IDLE time)
-l (sort by LOGIN time and date)
-n (sort by system number)
-p (sort by PROGRAM name)
-t (sort by TERMINAL name)
-u (sort by USER name)

-@ system name:

This option can be used to sort the WHO program display information alphabetically using the system "name" category.

-i idle:

This option can be used to sort the WHO program display information using the "idle" time category. This means that you will get a list of logged in accounts with the user whose keyboard has been the least idle at the top of the list, and the user whose keyboard has been idling the longest at the bottom of the list.

-l login:

You can sort WHO program information by the login time for each account attached to the network. The account that logged in first will be positioned at the top of this list, and the last account to log in will be positioned at the bottom of this list.

-n system:

The WHO program enables you to sort the timelog information by DR Net node numbers, in ascending order, that are attached to the network. By typing WHO -n @, this information will be displayed on your screen.

-p program:

The "program" sorting function of the WHO program enables you to list the various accounts logged in to the network by sorting the programs that they are currently running in alphabetical order. By typing WHO -p, the listing is sorted by program.

-t terminal:

By typing WHO -t, you will see a list of the various accounts attached to the network, sorted by the number of their foreground virtual terminal screen. This list will position the "console" at the top of the list and then the virtual terminal screen numbers in ascending order.

-u name:

The "name" option can be used in conjunction with the WHO program to list the accounts logged in to the network in alphabetical order. This is the sorting function that the WHO program will default to if no sorting option is specified on the command line. By typing WHO -u, the listing is sorted by account.

Each of the sorting and format options listed on the previous pages can be entered as a string on the command line. For example, suppose that you want to list WHO information for all of the virtual terminals on the network (not just the foreground screens), you want to list the seconds as well as hours and minutes in the timelogging categories, and you want to sort the list by whose terminal has been idle the longest. To do this, you must type the following command:

```
OA>WHO -R -I -A -S
```

The -r option is the first one that the WHO program uses to sort the information since this is the first option that was entered on the command line.

4. ~ terminal:

The tilde (~) is used to indicate that the following characters represent the name of a terminal attached to a system on the network. This command can be used to search for a specific terminal number across the network. For example, suppose that you want to list all of the accounts on the network whose terminal number is 1. To do this, type:

```
OA>WHO ~TTY1 @
```

and a table of WHO program information listing all accounts currently logged in to a network systems whose terminal number is listed as TTY1 in the TTYS file will be displayed on your screen.

5. ! program:

The "program" sorting function enables you to list WHO program information according to the program that each account is currently running. By typing (!), you are alerting the WHO program that the subsequent characters should be interpreted as the name of a program. Suppose that you want to find out who on the network is currently running WordStar. To do this, type:

```
0A>WHO !WS @
```

to list all WordStar (WS) programs running in the network.

WRITE

WRITE ALL
WRITE name tty# @node

The WRITE command is used to send a message to any account currently logged in to your system (or to any system in a DR Net local area network). This means that you can instantly send a message to any foreground virtual terminal screen.

The WRITE command must be followed by an account name, or you can send your message to all of the accounts logged into the local system using the ALL argument. The ALL argument, however, cannot be used across the network. The "@node" and "tty#" arguments are optional. The @node argument can be used to send a message to an account by specifying the system name, after the account name on the command line. The tty# argument can be used if you intend to send a message to an account that is logged in to more than one physical terminal. Here "tty#" is the **name of the terminal**, as it appears in the fifth field of the TTYS file.

The message is typed by the person who invoked the WRITE program. The source of the message will be shown on all terminals receiving it. The receiver of the message will see one line of text appear on the terminal screen each time that the sender presses the RETURN key. The WRITE program is terminated with a ^Z or with a period (.) in column one of a blank line. Users can simultaneously send messages back and forth.

Messages that are sent with the WRITE command do not affect a file being processed, but they do affect the screen display. WRITE messages will appear in the midst of text, spreadsheets, or programs which are being displayed on the receiving terminals. The WRITE program will first ask you if you want to interrupt the process that is running on the receiver's terminal. The WHO command will display the login account names and the terminal names of all currently logged-in operators and terminals.

The **name** option can be used to send a message to a terminal whose operator is currently logged in as that account name, as it appears in the first field of the PASSWD file. For example, suppose that an account called "rich" is logged in to your system, and you want to send him a message. To do this, type:

```
0A>WRITE RICH
```

If "rich" is currently running a program, such as WordStar, the following prompt will be displayed on your screen:

```
WARNING: User is running program "WS"
Do you want to continue?
```

Type (N) to interrupt the WRITE program, or (Y) to continue writing your message. If you type (Y), the following message will be displayed on your screen:

```
Ready
>
```

You can now begin typing your message to the screen of the operator logged in to the account called "rich."

The WRITE ALL utility can be used to send your message to all users currently logged in to the local system. Suppose that you want to send a message to everyone currently logged in to the system. To do this, type:

```
0A>WRITE ALL
```

After a WRITE ALL command, the WRITE program will interrupt all foreground programs currently running on the other terminals without issuing any warning to you. If you logged in as "mary," the following message will appear on all logged-in terminals:

Broadcast message from mary on tty2...

Now type the message to be sent. For example:

The computer will be shut down in 5 minutes.
Please save your files and log out.

The sender must enter ^Z, or a line with a period in column 1, followed by RETURN, to end the message.

The TTY# option must be used if you want to send a message to an account that is logged in on more than one physical terminal. Suppose that you use the WHO program to determine that there is an account called "john" that is logged in on terminals tty2 and tty3. You know that John is physically located at the tty3 terminal, and you want to send him a message right away. To do this type:

```
0A>WRITE JOHN TTY3
```

If John is not running a program on the foreground virtual terminal screen for that console, you will be prompted to begin typing your message.

Using WRITE in a DR Net Network

The WRITE program can also be used to send a message to any designated account currently attached to a DR Net local area network. The NET STATUS NODES and WHO @ commands should be used first to learn which systems are currently attached to the network, which accounts are currently logged in to those systems, and which programs are currently being run by those accounts.

The remote system to which you send a message, however, must be up and running. You can use the NET STATUS NODES command to determine the status of each of the systems in the network. For example, suppose that you want to send a message to the account called "pat" on the system named SALES. You must type:

```
0A>WRITE PAT @SALES
```

Provided that the account called "pat" is not currently running a program, you will be prompted to begin typing your message.

The TTY argument can also be used across the network. For example, if you want to send a message to the account called "john" at tty2 on the system named SALES, type the following command:

```
0A>WRITE JOHN TTY2 @SALES
```

The WRITE facility is intended as a convenience to users of a MC-DOS system or DR Net network, since it allows real-time communication by users who might be separated by large distances, or whose telephones are busy. A remote user can use WRITE over a modem. Excessive or frivolous use of WRITE can be a distraction to users, however, and if the WRITE command is abused, the system manager should protect it with a password and limit its use to urgent messages for all users on the system.

WTMP

The WTMP file contains a list of entries corresponding to login and logout. The file is used by the TIMELOG files for time accounting purposes. Entries in the file have the following form:

```
tty2   ldavid    12 16641016212034 7221982 125
tty2   l         12 16641326222034 7221982 126
```

In this example, "david" logged in at 10:16:21 and logged out at 13:26:22.

The first three fields are the physical terminal name, the relative virtual terminal number, and the account name. The other fields are:

```
12 1664 21 32 39 203 4 7 22 1982 125
 \  \    \  \  \  \   \\          \
  1  2    3  4  5  6   7 8         9
```

1. **Virtual** console number.

2. **Days** since January 1st, 1978.

3. **Hour** of the day (24 hour clock).

4. **Minute** of the hour.

5. **Second** of the minute.

6. **Julian day** of the year (1-365).

7. **Day** of the week (Sunday = 0, Monday = 1, Tuesday = 2, etc.).

8. **Date** in month-day-year format.

9. **Program file link number.** Each entry in the file contains the number of records in the file.

The WTMP file is covered again in **Chapter 3.**

2.6 System Security Overview of MC-DOS

Your approach to system security depends on how the system will be used, and on the sensitivity of the programs and data stored in the system. This section will illustrate four areas of system security: **login security, monitoring and special internal system security, standard Concurrent DOS security options,** and **DR Net security.** Each facet of system and file security can be established independently at a level appropriate to the perceived risk, ranging from a maximum level in which user activities are highly restricted, to a minimal level in which the security options are not used to control access to system resources. The purpose of this section is to present the range of options available in MC-DOS and DR Net security.

Because some security options globally affect all users, all files, all drives, or all systems in a network, while others selectively restrict only a few users, files, or devices, MC-DOS security options provide great flexibility to the system manager in matching security procedures to the current needs of the system or network. The system manager may choose, for example, to uphold the maximum level of login security but the minimum levels of internal system security, standard Concurrent DOS security, and DR Net security. In some situations it is neither necessary nor desirable to maintain a high level of system security. In other situations it is imperative to maintain the highest level of system security possible.

1. Login Security

Login security is controlled by the PASSWD file. The login PASSWD file contains descriptions of account login names, account login passwords, default user areas, default drive designations, optional terminal assignments, optional printer assignments, optional default program assignments, and accessible user areas. The PASSWD file entries for a system with a minimum level of login security differs from the PASSWD file entries for a system with a maximum level of login security. (Here "minimal login security" does not mean to delete the PASSWD file, since doing so would disable the appointment calendar, electronic MAIL, reminder files, WHO, WRITE, and other MC-DOS and DR Net programs.)

A. Minimum Level of Login Security

1. Account login names listed in PASSWD.
2. Login passwords are not assigned to any accounts.
3. Default user drive is specified.
4. Default user number is specified.
5. Unrestricted terminal access.
6. Unrestricted modem access.
7. Unrestricted or restricted program access. (Accounts with an entry in the optional default application program field of the PASSWD file are automatically entered into the program specified. It may be convenient to assign certain accounts to a default program if they always use the same program.)
8. Unrestricted user area access.

B. Maximum Level of Login Security

1. Account login names listed in PASSWD.
2. Login passwords for all accounts, to prevent unauthorized access to the system.
3. Default user drive is specified.
4. Default user number is specified.
5. Restricted terminal access for accounts.
6. Modem access restricted to accounts with default programs.
7. Restricted program access (default programs).
8. Restricted user area access. For practical reasons, there are exceptions to user area restrictions. Mail files stored in user area 15 of drive A: can be sent or received by all users, regardlesss of restrictions from user area 15. Command files SET to system (SYS) and read only (RO) status in user area 0 can be accessed by all users, even those restricted from user area 0.

2. Special MC-DOS System Security and Monitoring

To enforce user area restrictions and other PASSWD file controls, and to monitor system usage, MC-DOS has special system security features which can be enabled by a system manager who desires a maximum level of special internal system security and monitoring. The system manager satisfied with a minimum level of internal system security need not enable the special features.

A. Minimum Level of Special Security and Monitoring

1. All users can modify the PASSWD file with the NEWUSER utility by logging in as "system." All users can modify the PASSWD file with a word processor, but they should leave the password field empty, since a word processor cannot enter an encrypted password.
2. The special internal system passwords are not enabled if there is not a label with PROTECT=ON on drive A:.
3. Timelog reports are rarely, if ever, reviewed.
4. The optional HISTORY file is not created.

B. Maximum Level of Special Security and Monitoring

1. The PASSWD file can only be modified with the NEWUSER utility by the system manager who logs in to the "system" account. Only the system manager should know the login password for this account.
2. The special internal security feature of the operating system assigns secret passwords to the UTMP and PASSWD files so that no one can tamper with these files. There must be a label with PROTECT set ON on drive A: to enable password protection. Put a password on the label to prevent unauthorized users from changing the protect attributes.
3. The optional HISTORY file is created.
4. Timelog reports are regularly reviewed by the system manager.
5. Supplementary file encryption hardware is installed to create diskettes whose files cannot be read without similar equipment and knowledge of the encryption code.

3. Standard Concurrent DOS Security Features

Standard security features devised by Digital Research Inc. as part of the Concurrent DOS operating system include the option to restrict access to files by file passwords, and to mark each file's directory entry with the most recent time and date when the file was accessed or modified. The system manager has the option of implementing time and date stamping of files, and of setting passwords on selected files so that only users who can present the password can freely manipulate the files. Minimum and maximum levels of control are described.

A. Minimum Level of Timestamping and Password Protection

1. If you do not have a label on a logical drive, then files on that drive cannot have password protection or timestamping.

B. Maximum Level of Timestamping and Password Protection

1. Timestamping requires that the INITDIR program be run to reformat the directory of any logical drive whose files will have date and timestamps.
2. Give each logical drive that will contain files protected by file passwords a label with PROTECT set ON to enable file password protection.
3. Use the SET command to activate date and timestamping on labelled drives whose files are to be stamped.
4. Password protect each drive label with the SET command (refer to **2.5-SET-1**) so that the label cannot be modified.
5. Use SET to assign a file password to files that should be protected with passwords, using as appropriate the DELETE, WRITE, or READ level of protection (see **Section 3.4**).
6. Password protect powerful commands and programs such as SET and DDT86, and other utilities and languages that can directly address RAM locations.
7. Do not assign passwords to the following files:

NEWUSER	PASSWD	.MBX	MOTD
.REM	UTMP	TTYS	AUTOST

4. DR Net Local Area Network Security Features

In addition to the MC-DOS security features described earlier, MC-DOS supports security options for DR Net networks. These features can be established to minimally restrict access to files across the network, or to reduce or eliminate the extent to which some users, terminals, and systems can access the programs, files, and devices within the DR Net network, according to the way in which the system manager configures the network. Just as the PASSWD file is the keystone of login security, the CONFIG.NET and NETUSERS files control the access rights of network users.

A. Minimum Level of DR Net Network Security

1. Set up CONFIG.NET to support multiple "servers" and "requesters," so that several processes can access your system from other linked machines, or access other systems from yours.
2. Do not designate any drives as "private."
3. Do not use CONFIG.NET to set a node password on your system.
4. Have a PASSWD file on your system that lists all of the account names in the network, with the same passwords that they have on their local systems, and do not run PWASK.NET.

A. Maximum Level of DR Net Network Security

1. Run the PWASK.NET routine, so that every user who accesses your system from another system in the network will have to present a login password matching your PASSWD file's password for the account name that they present.
2. Use CONFIG.NET to set a password on your system. Only users who correctly present it can access your node's files or printers.
3. Have a PASSWD file on your system that does not list the account names of other network users, or that lists the account names with different login passwords than the passwords on their local PASSWD file.
4. Designate logical drives as "private," so that they will not be network resources.
5. Set up CONFIG.NET to support zero "servers" so that no users on remote systems can access your system from other linked machines via the network. Set "requesters" at zero if you want to keep the terminals of your system from accessing other parts of the network.

CHAPTER 3: MANAGING THE SYSTEM

Chapter 3 is a guide for the system manager describing how to install the MC-DOS operating system as a customized software environment. Sections of this chapter discuss system security, monitoring system usage, file maintenance, and backing up files. The last part of the chapter describes how to install MC-DOS, updates, or hardware upgrades. Many of the procedures required for system management refer to commands and files described in **Section 2.5** or in the separate Gifford publications CUSTOMIZING MULTIUSER CONCURRENT DOS USER'S MANUAL, VIRTUAL TERMINALS USER'S MANUAL, or the DR NET USER'S GUIDE.

CONTENTS

CHAPTER 3: MANAGING THE SYSTEM

CHAPTER 3: MANAGING THE SYSTEM

3.1 The System Manager

It is essential that someone be available at all times to give on-site system support. A system manager's responsibility is to insure efficient operation by monitoring the system and its use, and by encouraging users to cultivate good work habits.

Everyone using the computer should be aware that a multiuser system is a shared resource. Unlike a personal computer, a multiuser computer (or a DR Net network) brings many people together in a common work environment. How users manage their computer time and computer space affects other computer users. Every user should be considerate of others.

On-Site Support

To provide on-site support, the system manager should:

- Read this manual as well as the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE. These manuals provide the primary references for using the system. As the system manager, you should be familiar with the material in these manuals in order to know where to go for answers and where to direct users who need documentation. Keep these materials in a location where they will be convenient to users, together with the manuals of applications programs such as SuperCalc and WordStar.

- Maintain a written record of all passwords. This is important in the event that someone else has to take over responsibility for the system. To avoid being locked out of your own system, write down your passwords and keep them in a safe place. Keep track of account login passwords, drive label passwords, terminal passwords, and file passwords.

- Maintain a map of the system showing the location and identity of terminals, printers, and cables, like the map in **Appendix M.**

- If you have terminals with programmable function keys, these can be programmed with the KEYS utility and specific .KEY files to match the needs of individual users or tasks. The procedure is described in the description of the KEYS command in **Section 2.5,** and in Section 3 of the VIRTUAL TERMINALS USER'S MANUAL.

- Be aware that many applications programs such as WordStar can be customized for individual needs. This is often done with the assistance of an **install** program. With WordStar, for example, you can set the default help level, margins, word wrap, hyphen help, or printer commands to meet specific needs. You can create custom versions of WordStar and rename these for specific individuals, departments, or specific tasks. If you do this kind of customization, you only need to rename the WS.COM file to distinguish one custom version from another. Thus, the copy of WordStar installed to drive a draft quality printer might be called WSDRAFT, while a letter quality version might be called WSLET. Do not change the names of the overlay files. (See also **Appendix H.**)

- Be prepared to contact your dealer for support if your system seems to operate improperly. This manual provides a limited troubleshooting guide. Review **Chapter 4** to understand what information to have ready if you need to call for support.

- Keep the system current. The **GIFFORD OBSERVER** newsletter announces new releases of MC-DOS when they become available. You can obtain new MC-DOS releases and manuals at a nominal cost. Changes that Gifford makes in the system are customer driven. Give us your suggestions for improving our product. We value your input and we are responsive to it.

- Make sure that all terminals and printers connected to the system are turned ON. If they are turned OFF, their cables should be disconnected from the back panel of the computer enclosure (not just the device end).

- The system manager should oversee shutting down the system for servicing. Improper shutdown may result in loss of data.

- Make sure that backups are done at regular intervals. It is extremely important to make backup copies of files. Backing up files is your ultimate protection against loss of valuable data.

- Maintain an adequate stock of materials, including spare printer elements, ribbons, paper, and blank diskettes, so that work will not be constrained by a lack of supplies.

CHAPTER 3: MANAGING THE SYSTEM

Establishing Efficient User Habits

In addition to providing the on-site support outlined above, the system manager should outline and enforce efficient habits among users. Recommended practices include:

- Use file names and file extensions that are self-documenting. Make the habit of always using the file extension. You might call a temporary file something like DOCUMENT.TMP, or a laundry list LAUNDRY.LST. Make the file names helpful in locating data in your directories, and make them document what you are doing. This will make managing file directories an easy chore.

- Have users back up important files on their own, independent of the system backup procedures that you use. This is additional insurance against the loss of valuable files. Have the users do this during the workday, after lunch for example. If users wait until the end of the day, they may sometimes neglect this task. Use procedures that do not copy over the last backup.

- Have users erase WordStar's .BAK files, or similar temporary backup files. These files tend to accumulate and take up valuable space, and make directories harder to read. Keep work areas clean by eliminating non-essential files.

- Have users keep listings of program, data structure files, and dBASE II index keys. These items are essential for documenting work or reconstructing files, should the need arise.

- Make sure that users log out when they are not using the system. This makes it easier to isolate system problems and gives you the ability to maintain accurate timelog reports.

- If programs (or the ^P control character) are used that do not release the printer after a print job, make sure that users release printers as soon as possible after printing. For example, people should not go to lunch while printing documents on a letter quality printer. As an additional courtesy, have users set the paper in a tractorfeed printer at the start of a page when they remove their output. This way the printer is always ready for the next user without requiring adjustment.

CHAPTER 3: MANAGING THE SYSTEM

3.2 Creating a Customized Multiuser Environment

In order to understand how to customize the system for your specific needs, you need to know what files to modify and how to modify them. The following section documents the operations of starting up, logging in, logging out, and shutting down the system. The points at which these MC-DOS operations can be customized are indicated, and the files and commands necessary to customize them are identified.

Description of a System Startup

When the RESET button is pressed (or the computer is powered up), the operating system does not exist in RAM (memory) and must be loaded from the system diskette. The computer starts up by going through the following operations:

1. A floppy diskette with the MC-DOS system on it is placed in the left-hand floppy drive. The computer instructs the disk drive to go to the system tracks, read the file MC.DOS from the left-hand floppy drive, and load the operating system into memory. From this point, the computer only looks in user area 0, drive A: for the system files described here.

2. If the system has an MDRIVE/H RAM disk emulator it executes the file BUFFER.CMD to initialize the hard disk cache buffer.

3. The computer runs the CCPMINIT program, which reads the VTTYS file. VTTYS describes the virtual terminals on your system, and must be in user area 0 of drive A: for the system to support virtual terminals. (If VTTYS is later modified, the system must be rebooted to read the new VTTYS file.)

4. If the system is in a DR Net local area network, the SETNET program loads and reads the CONFIG.NET file, initializing the internal tables of the network input/output system (NIOS).

5. The resident system processes called "shells" start up, and read the TTYS and LPRS file. TTYS describes the physical terminals and modems on your system, LPRS the printers and plotters. Both files must be in user area 0, drive A: for the system to support multiple users and list devices. (If

your terminals or printers are later changed, the TTYS or LPRS files that specify the baud rate, handshaking, etc., must be modified accordingly. Reboot the system or type RESET so that the system reads the modified TTYS or LPRS.)

6. The computer runs an optional program, SYSINIT.CMD, which can be any program that you rename SYSINIT.CMD so that it will be part of the system startup process. Use REN or PIP to rename the chosen program SYSINIT.CMD, and place it on user area 0 of drive A: to be run automatically at startup.

7. The computer searches for an optional file SYSINIT.SUB, a SUBMIT file whose lines are a list of commands and programs that you wish to run as part of the startup process. The SYSINIT.SUB file is optional for customizing the startup process. **Section 2.5-SUBMIT** describes SUBMIT files.

8. The computer looks for a file called AUTOST. If AUTOST is present on user area 0 of drive A:, the initial user is automatically logged out, and all of the terminals will display the **Name:** prompt. If AUTOST is not found, the computer remains a single-user system, and only the system console can communicate with the computer until the initial user types LOGOUT. If you want to inhibit the automatic jump to multiuser mode, you can erase AUTOST. AUTOST does not need to contain anything, and can be created by typing:

```
0A>MAKE AUTOST
```

9. When the system enters multiuser mode, it automatically erases the UTMP file, if it exists, and then creates a new, empty UTMP file. The system then puts a special write protect password on the UTMP and PASSWD files.

10. The system looks for the optional file SYSINIT.BAN. If this is not found, the default banner **Gifford Computer Systems Multiuser Concurrent DOS** is displayed. If the file is found, its contents are displayed on all terminals in place of the default banner. Any terminal that is waiting for a user to log in will display the banner. This completes the startup.

CHAPTER 3: MANAGING THE SYSTEM

Once the system is up, an automatic RAM test continually cycles through the RAM, and the system automatically executes a SYNC command every 30 seconds. If the optional HISTORY file exists, the system will record all the commands entered, the terminals on which they were typed, and the account name giving the command.

Description of User Initialization

The following steps take place when a user logs in at a terminal. All the files used during user initialization are optional.

1. The system looks for a file called PASSWD. (If PASSWD is not found, the user initialization process is complete, and the **0A>** prompt is displayed, skipping all of the steps that follow.) Once PASSWD is found, the computer displays the name prompt (**Name:**). The user responds by typing an account name.

2. Next the computer looks for an optional password in the PASSWD file. If the login account name entered during step 1 has an associated password, the **Password:** prompt is displayed. The user types a password, which the screen does not display. If the entry does not match the password in the PASSWD file, an error message is displayed and the name prompt is repeated.

3. If the password matches, the system checks whether the terminal being used is contained in the selective terminal access field of the PASSWD file. If the terminal does not have access rights to the login account, an error message is displayed, and the name prompt is repeated. If the terminal has access rights to the login account, the system reads from the PASSWD file the default work area and printer.

4. If the files UTMP amd WTMP exist, the system writes the account name, terminal, and login time in these files.

5. If the message of the day file (MOTD) exists, its contents are displayed on the user's screen. If the network message of the day (NETMOTD) exists in systems on a network, then its contents are displayed on the screen of each networked terminal.

6. The system checks to see if the account has any mail. To check for mail, the system looks for the file **name**.MBX in user area 15, drive A: of the system (or in a DR Net network, on that directory area of the system node, where the MAIL facility resides). Here **name** is the account name from the PASSWD file (or in a network, from the NETUSERS file that lists the account names from the PASSWD files of all the systems in the network). If **name**.MBX is found, the following is displayed:

You have mail.

7. The system checks the account's initial default directory area (specified in the PASSWD file) for a reminder file with the file designation **name**.REM (matching the account name). If this file is found, its messages are displayed on the screen.

8. In a DR Net network, the system looks in the initial default directory area for an optional file name that is the same as the account name, and with **.NET** as the file extension. If the file is found, the system maps and connects the virtual terminal to the network, according to the specifications in the **name**.NET file, as if the user had typed the command:

7B>NET CONFIG **name**

The file **name**.NET can be created when a user who is already connected to the network gives the command:

7B>NET MAKE

The effect is to copy the current network mapping to a file called **name**.NET. (The system warns the user if a file of that name already exists in the same directory area, and asks for confirmation to overwrite the old **name**.NET file. See the DR NET USER'S MANUAL for more detail on the NET command.) If the file **name**.NET does not exist, the virtual terminal is not automatically connected to the network.

9. The system looks for USERINIT.CMD on user area 0 of drive A:. If found, the program is run. USERINIT.CMD can be any program file which is to be included as part of the login routine for all users, and renamed USERINIT.CMD by the REN or PIP command.

10. The system looks for the optional file USERINIT.SUB, and processes it if found. The file USERINIT.SUB can contain any commands which are to be run as part of the login routine for all users. Gifford Computer Systems supplies a sample file USERINIT.SAM which shows how a SUBMIT file can be set up to display mail automatically and type a message.

11. The system looks for the optional program LOGIN.CMD on the drive and user area assigned to the account name by the PASSWD file and runs the program if it finds it. LOGIN.CMD can be any program file which is to be included as a part of the login process, renamed by the REN or PIP command. A user might, for example, rename the appointment calendar program (AP.CMD) as LOGIN.CMD, to check for any reminders.

12. The system looks for the optional file LOGIN.SUB on the drive and user area assigned to the account name by the PASSWD file. If found, the file is submitted for processing. LOGIN.SUB can be any SUBMIT file which is to be included as a part of an individual user's login routine.

13. If the PASSWD file specifies a default applications program, the program is run. When this program is exited, the user is automatically logged out of the system. If no default program is specified in the PASSWD file, the MC-DOS prompt is displayed, indicating the default user number and drive designation.

Description of a Logout

1. When a user gives the LOGOUT command, the system looks for the optional file LOGOUT.SUB on the current drive and user area. If found, the file is submitted for processing. LOGOUT.SUB can be any SUBMIT file which is to be included as a part of an individual user's logout routine.

2. In a DR Net network, a LOGOUT command causes the system to disconnect the current virtual terminal from the network.

3. The system looks for the UTMP file. If UTMP is found, the system removes the account name, terminal number, and login time from the file, and updates the WTMP file if it exists. The system then returns with the name prompt, so that on systems with a PASSWD file, an account name and an optional password must be presented to access the system through any terminal.

Description of a System Shutdown

When the system is intentionally shut down by a user at the system console, a number of important processes occur. If the system is shut down either by interruption of the power supply, by typing the DOWN command, or by pressing the RESET button without first giving the proper sequence of commands, files may be lost or damaged. The operations that should occur during an intentional shutdown include:

1. All users except the system console give the LOGOUT command. This closes all files and displays the banner and name prompt at all the other terminals.

2. The user at the system console should exit from any programs and terminate or complete any processes attached to the console. The system console will display the MC-DOS prompt.

3. The user at the system console types WHO -A and the system displays a list of the virtual terminals still on the system. If users are still on the system, each should give the LOGOUT command from each virtual terminal. The WRITE command can be used to advise users whose virtual terminals are marked by WHO with an asterisk (*) to exit from any programs and log out of the system.

4. The user at the system console types SYNC. This command flushes the cache buffer onto the hard disk.

5. The user at the system console types DOWN. This turns OFF all other terminals, and puts the system into singleuser mode.

6. The system looks for a file called DWN.CMD, and runs the program if it is found. DWN.CMD can be any program that is meant to run as part of your system shutdown routine.

7. The system looks for DOWN.SUB, and if found, submits the file for processing. The file DOWN.SUB can be used as the counterpart to the startup command SYSINIT.SUB. Here is an example of the contents of a DOWN.SUB file, whose command saves the contents of the spooler queue.

```
SPOOL TERMINATE
```

Like the other customization SUBMIT files, DOWN.SUB can be created by a text editor like ED or by WordStar in non-document mode.

8. After DOWN.SUB has been executed, the system can be shut down by pressing the RESET button while turning OFF the power switch.

Summary of MC-DOS Customization Files and Commands

Files and commands that can be used to customize MC-DOS at startup, login, logout, or shutdown include:

AP.CMD	LOGOUT.SUB	SETNET.CMD
AUTOST	LPRS	SYNC.CMD
BUFFER.CMD	MAIL.CMD	SYSINIT.BAN
CCPMINIT.CMD	MC.DOS	SYSINIT.CMD
CONFIG.NET	MOTD	SYSINIT.SUB
DOWN.CMD	NETMOTD	TTYS
DOWN.SUB	**name**.MBX	UIMP
DWN.CMD	**name**.NET	VTTYS
HISTORY	**name**.REM	USERINIT.CMD
KEYS.CMD	NET.CMD	USERINIT.SUB
LOGIN.CMD	NETUSERS	WHO.CMD
LOGIN.SUB	PASSWD	WRITE.CMD
LOGOUT	RESET	WIMP

3.3 Setting Up the System Management Files

The following section provides the information to assist the system manager in modifying or creating system management files tailored to the needs of a specific system. This discussion will begin with the LPRS, TTYS, VTTYS, and PASSWD files. These files configure the system's printers and plotters (LPRS), modems and terminals (TTYS), multitasking Virtual Terminals (VTTYS), and the account names and access rights (PASSWD) of operators who can use a Gifford Multiuser Concurrent DOS system. Since these four files are ordinarily delivered with new MC-DOS system, the system manager does not need to create these files, unless they have been accidentally deleted.

The system manager should read this section to understand how to change LPRS, TTYS, and VTTYS, if these files were incorrectly installed, or if your dealer did not know what terminals and printers you planned to use with your system, or if in the future you add or substitute different input or output devices.

The PASSWD file generally **must** be modified by the system manager at installation and from time to time, since the version delivered with a new MC-DOS system only contains the "system" account of the system manager. By learning how the PASSWD file works and how the NEWUSER utility maintains it, the system manager can specify the account names that operators will use to access the system, and install appropriate security restrictions.

The LPRS File

The LPRS file describes the characteristics of each printer or plotter attached to the system. Each line of LPRS describes one output device (a "list device") attached to your system, in data "fields" each separated by colon (:) symbols. If your list devices were known prior to shipment, you may not need to change the LPRS file. An example of an LPRS file is:

```
0:8:1:82n:0,lpr0:System printer, Texas Instruments
1:8:2:81n:1,h-p:H-P graphics plotter
```

The LPRS information is in the following format:

```
DEV #:BAUD RATE:HANDSHAKING:PROTOCOL:LPR NAMES:COMMENTS
    \      \          \          \         \         \
     1      2          3          4         5         6
```

1. **Number of the list device,** corresponding to the named connection on the back panel of the computer enclosure. (This value is used by the RESET command. Printer 0 is the default printer when any account logs in, unless the PASSWD file specifies another printer or plotter.)

2. **Printer baud rate,** defining the speed at which the printer or plotter communicates with the system. The same baud rate must be set both on the physical list device and on the line in the LPRS file describing that device. The default is 9600 baud. The LPRS file uses the following codes to describe baud rates:

0 =	9600	5 =	1800
1 =	110	6 =	2400
2 =	300	7 =	4800
3 =	600	8 =	9600
4 =	1200	9 =	19200

3. **Optional RS-232 hardware handshaking.** "Handshaking" refers to procedures by which the list device indicates that it is ready or not ready to receive data from the computer. The default is 0. If your device requires handshaking, enter the type of handshaking needed, using the following codes:

   0 = No handshaking
   1 = Hardware handshaking (DTR) (RS-232 pin-20)
   2 = Software handshaking (XON/XOFF)
   3 = Reserved for custom application

4. **Optional communications protocol.** This describes the format in which information is transmitted by your system to the printer or plotter. Leaving this field blank selects the default protocol of 8 data bits, 2 stop bits, and no parity, which are the settings for most printers.

The LPRS file uses the following communications protocol codes:

Data bits	Stop bits	Parity
8	2	N (none)
7	1	O (odd)
		E (even)

If you need to change the communicatons protocol, select the new values that you need from the table above and enter the three characters in the fourth LPRS field without any spaces. For example, if your printer requires a protocol of 7 data bits, 2 stop bits, even parity, enter 72E in field 4.

5. **Printer names.** In addition to the "list device number" (also called "printer number") from the first field of LPRS, you should also assign a **name** or a list of names to each printer or plotter. Some users find it easier to remember, for example, that a device is called "DIABLO" than whatever its list device number may be. The printer names listed here are separated by commas (,) and are used in the PASSWD file to assign an initial default list device, and by the PRINTER command to display or change the current printer.

   Gifford recommends that the first name listed here be the list device number from the first LPRS field. In the example LPRS file on **page 3.3-1**, which lists "1,h-p" in the fifth field as names of list device number 1, users could make this plotter their current list device by either of the following commands:

```
2B>PRINTER 1

2B>PRINTER H-P
```

6. **Optional comments field.** This allows you to make notes about each device, such as its location or model number.

Read the manufacturer's manual to learn the baud rate, hardware handshaking, and communications protocols supported by your printers and plotters. The physical list devices must be set to match the codes in the LPRS file for these values.

The LPRS file can be modified while the system is running in multiuser mode, using a text editor like ED or WordStar in non-document mode. After the LPRS file has been modified, type:

```
0A>RESET
```

This will reset the system to the new LPRS specifications. Do **not** confuse this command with pressing the red RESET button on the computer, which aborts all of the processes running on the system, causing users to lose any work not saved to the disk.

The maximum number of list devices that your operating system can support is specified in the extended input/output system (XIOS) of MC-DOS. You cannot install printers or plotters beyond the XIOS limit merely by adding lines to the LPRS file. Read the Gifford publication CUSTOMIZING YOUR MULTIUSER CONCURRENT DOS SYSTEM for information on how to modify MC-DOS to change the XIOS limit on the maximum number of list devices.

The TTYS File

The TTYS file provides your Multiuser Concurrent DOS operating system with characteristics of each terminal or modem attached to the system. A TTYS file is included on your system diskette. Each line in the TTYS file describes one terminal device on the system, in the same format as the LPRS file, starting with device 0, the "system console." A colon (:) separates each field of TTYS. Here is an example of a TTYS file:

```
0:9:0:82n:console:System support
1:9:0:82n:tty1,front:Interfacer 3 port 0
2:9:0:82n:tty2,sales:Interfacer 3 port 1
3:10:0:81n:tty3,modem0:Interfacer 3 port 2, VenTel 212
```

If your terminal arrangement was known when you purchased your machine from Gifford Computer Systems, you will not need to change this file, unless you later add or substitute terminals.

The format of TTYS closely resembles that of LPRS:

```
DEV #:BAUD RATE:HANDSHAKING:PROTOCOL:TTY NAMES:COMMENTS
   \     \          \           \         \         \
    1     2          3           4         5         6
```

1. **Device number of the terminal,** corresponding to the named connector on the back panel. Numbers begin with 0.

2. **Terminal baud rate,** whose default is 9600. As with the LPRS file, the same baud rate must be set both in the file and on the physical device, or else the device will not function correctly. The TTYS file uses these baud rate codes:

0	=	9600	6	=	2400
1	=	110	7	=	4800
2	=	300	8	=	9600
3	=	600	9	=	19200
4	=	1200	10	=	automatic baud rate detection for modems
5	=	1800			

   The automatic baud rate detection for modems only works for 300 and 1200 baud. To use this feature, pin-20 on the RS-232 connector must be wired to the speed detect pin on the modem.

3. **Optional RS-232 hardware handshaking.** Hardware handshaking is not required for most terminals; the default is 0. If handshaking is necessary, enter the code number corresponding to the type of handshaking needed from these codes:

   0 = No handshaking
   1 = Hardware handshaking (DTR) (RS-232 pin-20)
   2 = Software handshaking (XON/XOFF)
   3 = Reserved for custom application

4. **Optional communications protocol.** This is a three-character entry that describes the format of information sent to or from your terminal device. The default protocol is 8 data bits, 2 stop bits, and no parity bit, which are the settings for most terminals.

If you need to change this, use the following codes to select the configuration that you need:

Data bits	Stop bits	Parity
8	2	N (none)
7	1	O (odd)
		E (even)

The communications protocol is entered as three characters without intervening spaces. For example, if your terminal requires a protocol of 7 data bits, 2 stop bits, even parity, enter **72E** in field 4 of TTYS.

5. **Terminal names** of up to 8 characters with no spaces. Many terminal names can be listed, separated by commas (,) with no spaces. If you enter "console" as the first terminal name of the system console, "ttyl" for terminal 1, "tty2" for terminal 2, etc., then your notation will correspond to that used by Gifford in the examples that document the HISTORY, UTMP, TIMELOG, WHO, WRITE, and other MC-DOS utilities that identify a physical terminal by the first name in this field.

6. **Optional comments field.** This allows you to make notes about each terminal device, such as its location or model number.

In the first line of the example TTYS file on **page 3.3-4**, terminal 0 (the system console) communicates at 19200 baud, uses no handshake, uses the standard communications protocol (8 data bits, 2 stop bits and no parity), and has the name "console."

The TTYS file can be modified while the system is running in multiuser mode. After the TTYS file has been changed and the modified file saved in user area 0 of drive A:, give a RESET command. This will reset the system to the new specifications without the need to shut down the system. (Do not confuse giving the RESET command, which does not disrupt MC-DOS operations, with pressing the RESET button, which aborts all current programs!)

Modifying TTYS cannot increase the number of terminals past the maximum number specified in the extended input/output system (XIOS) of MC-DOS. Read Gifford's CUSTOMIZING YOUR MULTIUSER CONCURRENT DOS SYSTEM manual on changing your XIOS.

The Multitasking VTTYS File

The Multiuser Concurrent DOS operating system can support "multitasking" operations in which any properly equipped physical terminal can support up to four simultaneous programs. Special buffer areas within the RAM memory, called "Virtual Terminals," keep track of these processes, so that the physical screen can resume the display when the operator wants to monitor a specific program.

This multitasking feature of MC-DOS is controlled by an ASCII file called VTTYS. Like LPRS or TTYS, this file must be located in user area 0 of drive A:, and can be modified by a text editor like ED or by WordStar in non-document mode. The VTTYS file describes your virtual terminals to the operating system. For VTTYS to work correctly, the physical terminals must be equipped with supplementary display memory, and the CCPMINIT file must exist on user area 0 of drive A:. The MC-DOS command VCMODE specifies how programs running on virtual terminals that do not control the physical screen (called "detached processes" or "background processes") produce output.

If complete information on the virtual terminals configuration for your system was known to Gifford when your system was ordered, then a VTTYS file is included on your system diskette. Otherwise a file called VTTYS.SAM is included, showing examples of VTTYS lines. Like the TTYS file, each line of the VTTYS file describes one physical terminal. A TTYS line has only only six (6) fields, while a VTTYS line has sixteen (16) fields. Since the VTTYS file contains so much information, if you need to create your own VTTYS file it may save you time to make a copy of the sample file VTTYS.SAM, and then edit and rename the copy.

An example of a VTTYS file is shown below:

```
0:::::4:4:/^_:/^N/Eg::/EK/0/0:/EJ/0/0:/Eg/Ef/EG4:/R: Console
     /V | /p0|/p1|/p2|/p3|pr=/L /e/s/o:Fr-200 TV-950 4 pages
1:::::5:4:/^_:/^N/Eg::/EK/0/0:/EJ/0/0:/Eg/Ef/EG4:/R: Console
     /V | /p0|/p1|/p2|/p3|pr=/L /e/s/o:GCS-80 CIT-80 5 pages
2:::::2:2:/^A:/^N/Eg::/EK:/EJ:/Eg/Ef/EG4:/R: Console
     /V | /p0|/p1|/p2|/p3|pr=/L /VT=/v:TeleVideo 925 2 pages
```

Here each line is divided to accommodate the page width of this manual; in a real VTTYS file the lines extend undivided to whatever width is required. In the example on the previous page, the first line describes a terminal like a Freedom 200 (or any other terminal that can emulate a TeleVideo 950 with four pages of display memory). The second line describes a GCS-80 or C.Itoh CIT-80 terminal with five pages of display memory. The third line describes a TeleVideo 925, with two pages of display memory. Note that the caret symbol (^) in this example represents an actual caret (^) keystroke, which is the shifted numeral 6, in a VTTYS file, rather than meaning that you should somehow enter a control character in the file. (It happens that the sequence "/^" in a VTTYS file tells MC-DOS that "the next character should be interpreted as a control character.")

Using the first line of the example VTTYS file as an illustration, the sixteen fields of a VTTYS file are defined as shown below. (Fields 2 through 5 are reserved for future use.)

```
0:::::4:4:/^_:/^N/Eg::/EK/0/0:/EJ/0/0:/Eg/Ef/EG4:/R: Con
 \     \\\\  \     \\       \       \           \   \
  1     6 7 8   9    10 11      12      13         14  15

    sole /V | /p0|/p1|/p2|/p3|pr=/L /e/s/o:Fr-200 TV-950 4 pages
                                                \
                                                 16
```

1. Physical terminal number, the same as the first TTYS field.

2-5. Reserved for future use by Gifford Computer Systems.

6. The number of pages of display memory in the terminal.

7. The number of virtual terminals supported by the device. This can be no greater than four or the previous field.

8. The "leadin" character used to switch the screen. It is convenient to program function keys with the KEYS program to produce this sequence and the screen number, so that you can change virtual terminals with a single keystroke.

```
0:::::4:4:/^_:/^N/Eg::/EK/0/0:/EJ/0/0:/Eg/Ef/EG4:/R: Con
 \     \\\\  \     \\       \       \          \   \
  1     6 7 8   9    10 11     12      13         14  15

    sole /V | /p0|/p1|/p2|/p3|pr=/L /e/s/o:Fr-200 TV-950 4 pages
                                                \
                                                 16
```

9. The initialization sequence that MC-DOS sends to the terminal when CCPMINIT installs virtual terminals at system startup. In the example (which is repeated above) a line feed (/^N) enabling the DTR handshake, and an ESC g sequence to display the 25th (status) line, are sent at startup.

10. The sequence to directly switch to another screen. Since the TeleVideo 950 lacks this feature, the field is empty, and screen switching is controlled by the next two fields.

11. The sequence to switch to the next screen (for example, from screen #2 to screen #3), here ESC K, with two zeros (/0/0) to make sure that the status line is updated when you switch to a new virtual terminal screen.

12. The sequence to switch to the previous screen (for example, from screen #2 to screen #1), here ESC J, with two zeros (/0/0) to make sure that the status line is updated when you switch to a new virtual terminal screen.

13. Escape sequences to turn the status line ON, enable load status line, and display the status line in reverse video.

14. The carriage return sequence to disable the load status line.

15. This field defines the information displayed on the status line. The example specifies the word "Console" and the current "relative terminal" (screen within display memory), the names of processes running on each virtual terminal of this physical terminal, the symbols "pr=" followed by the printer number, and any ^P, ^S, or ^O control characters that currently affect the output of that virtual terminal.

16. A field for comments or notes by the system manager.

The eight VTTYS fields 8 through 15 specify sequences of characters that MC-DOS uses to control the terminal in multi-tasking operations. These control sequences and escape sequences are specific to each type of terminal, and are defined in the manufacturer's documentation for each terminal. The following notation is used to define the special character sequences in fields 8 through 14 of the VTTYS file:

:	Begin a VTTYS field
/	Interpret what follows as a special code function, rather than as a printable ASCII character
//	One printable ASCII slash (/) character
/:	One printable ASCII colon (:), rather than a field boundary
/^	Interpret the next character as a CONTROL character
/0	Send a null character (0H) for a very short "padding" delay
/0/0	Updates your status line to reflect new virtual terminal screen
/C	The logical terminal number (the same value that the CONSOLE command displays as the "virtual terminal number")
/D	Causes a longer delay (= 1/60 of a second) than /0
/E	The ESC key (1BH), used to initiate escape sequences
/l	Display the current leadin character (for switching screens)
/N	Line feed (0AH)
/P	Physical terminal device number (the first field of TTYS)
/R	Carriage RETURN (0DH)
/V	Relative virtual terminal number (the page of display memory in the physical terminal), whose value can be 0, 1, 2, or 3

The fifteenth field, which defines the display of the status line at the bottom of the screen, recognizes additional symbols:

/L	Number of the current list device (the first field of LPRS)
/e	Display ^P if printer is toggled ON by a ^P
/m	Display the VCMODE program's background output option
/n	Display in right-hand corner the foreground program name
/o	Display ^O if output is suppressed by a ^O
/p#	Display the name of the program running on relative virtual terminal number # (you must substitute 0, 1, 2, or 3 for #)
/s	Display ^S if scrolling is interrupted by a ^S
/v	Same as (/V + 1); value can range from 1 up to 4

See also Gifford's VIRTUAL TERMINALS USER'S MANUAL for more detail on the VTTYS file and on the VCMODE and KEYS programs.

Each line in a VTTYS file refers to the physical terminal whose device number (the first TTYS field) matches the first field of VTTYS. Only physical terminals that will support virtual terminals need to be described in a VTTYS file. The number of virtual terminals supported by a physical terminal (field 7 of VTTYS) can be less than the available pages of additional screen memory (field 6 of VTTYS), but should not be greater than the value of field 6 or 4. In systems with programmable function keys, the KEYS program sould be used to implement screen switching sequences.

Terminals without additional display memory can have their status line configured by VTTYS, and can even specify screen switching sequences to detach a process from the screen and attach the screen to a process that was in the background. Without additional display memory, however, the previous screen display of the background process cannot be instantly restored to the screen. Unless the operator asks the program to supply a new display (by scrolling forward in a word processor, for example, or by jumping to a distant cell in a spreadsheet) a terminal that has no extra memory presents an incomplete display immediately after attaching a background process.

The XIOS (extended input/output system) of your MC-DOS must specify a number of physical and virtual terminals equal to or greater than what your TTYS (physical) and VTTYS (virtual) terminal configuration files specify. If you modify your VTTYS file in a way that increases the total number of physical and virtual terminals, you may need to use the GENCCPM command to remake MC-DOS with an XIOS of a larger capacity.

Note that the operating system can support virtual terminals because it reserves space in RAM to keep track of what each virtual terminal is doing. When you replace your current XIOS with an XIOS of larger capacity, you also increase the size of your operating system, and leave less space in RAM available for user programs. Select the smallest XIOS that can support the virtual terminals that you require. Similarly, programs running on virtual terminals occupy RAM, so a large number of active virtual terminals can exhaust all the available RAM in your system. Heavy use of virtual terminals may require that you add more RAM to your system, but 1024K (a megabyte) is the upper limit on the total RAM that Concurrent DOS 3.1 can utilize.

The PASSWD File and the NEWUSER Command

The PASSWD file lists the login account names that can be used to sign on to the system, and records an optional encrypted login password required to log in for each account. The PASSWD file also defines for every account name an initial default drive, user area, printer, and optional default applications program. The PASSWD file plays a central role in maintaining system security, since the system manager has the option of using PASSWD to specify optional restrictions on the user areas within a system or network in which the account is allowed access to files, and on terminals and modems from which the account can log in to the system. The PASSWD file also supplies each account with an initial directory area and printer, so that at login, each operator can be assigned to a part of the system where only the files and programs necessary for that operator's work are stored. This speeds and simplifies use of the system.

The PASSWD file treats each account independently, so that a PASSWD file can simultaneously perform the following functions:

- It can restrict login access to the system to operators who present a valid account name and the corresponding password.

- If each operator (or at least each project team) uses a unique account name, the names can provide unique identifiers for sending electronic mail and reminders, maintaining appointment calendars, sending messages to terminal screens, monitoring system usage, individualizing network assignments, and maintaining timelog information for accounting purposes.

- It can assign each account at login to a unique directory area, so that files created by each operator tend to be in the same directory areas. This reduces the hazard of different operators simultaneously working in the same directory area.

- It can restrict login access of accounts to specific terminals and modems, even if the correct password is given.

- It can restrict file access of accounts to specific user areas. In systems that implement a high level of file security, only the "system" account can access areas 0 or 15.

- It can restrict any account to a specific applications program, so that any operator logging in with that account name cannot access other programs or files, nor give MC-DOS commands.

- It can assign any account to a specific hardware and software environment, so that any operator logging in with that account name is immediately running a default program, without needing to invoke the program, nor to log out after exiting.

NEWUSER

The NEWUSER program is a security utility that must be used by the system manager to maintain the PASSWD file. To use NEWUSER, the system manager must first have logged in as "system." If someone who has not logged in as "system" attempts to invoke NEWUSER, the system responds:

```
Sorry but the 'newuser' program can only be run from
the 'system' account. Please check with the system
manager.
```

To invoke NEWUSER, typing the keyword NEWUSER causes the main NEWUSER menu to appear, as shown in the following dialogue:

```
0A>NEWUSER

          Gifford Computer Systems - newuser
                  Maintenance Package
        Copyright (C) 1984 by Gifford Computer Systems, Inc.
          Version 2.0 of newuser updated on Aug 3, 1984

     A) Add a user.
     B) Brief listing of users.
     C) Change a user's parameters.
     D) Delete a user.
     E) List users in detail.

     X) Exit.
    ^C) Abort, make no change.

Option:
```

The "A" Option

If you type (A) to add an account, the program responds:

Enter account name:

Type the account name with which a user will log in to the system. The name can have up to eight (8) letters. The name entered here will also be used by the AP, MAIL, WHO, WRITE, and reminder (.REM) programs, by the WTMP, UTMP, and HISTORY files, and by the NET MAKE and NET CONFIGURE commands in DR Net. If you enter project names as account names, the WTMP and TIMELOG utilities will keep records on individual projects, but MAIL, HISTORY, and password encryption will have little value. Most system managers use real personal names as account login names.

Each account must have a unique name. If you enter an account name that already exists on the system, the program will respond with the following message:

The name you have selected is already in use.

When a new account name is entered, the program responds:

Password:

Each account name should have a password. An account without a password will allow easy access to the system and defeat the purpose of the password security feature.

The next two prompts request an initial default drive, and an initial user area number (together specifying an initial directory area) where the new account will be after login:

Select default drive:

Enter a valid drive letter, without a colon (:), such as A, B, C, etc. After you enter the initial default drive specification, the program responds:

Enter default user number:

Enter a default user number between 1 and 14. In selecting the initial default directory areas, most system managers give each account a unique combination of user area number and logical drive. In this way a DIR command shows only the files of that account, and there is less risk of users interfering with one another by running in the same directory area programs that try to create intermediate files with the same name. Users can change directory areas by the **d:**, USER and PIP commands to access all drives and user areas from which they are not restricted.

It is recommended that you **not** make user areas 0 nor 15 of drive A: default work areas. User area 0 of drive A: should be reserved for system files, and user area 15 of drive A: should be reserved for MAIL and calendar (AP) files, rather than used as work areas. NEWUSER allows accounts to share the same default drive and user area, but warns if this is the case.

The system next lists all the valid names of printers (and plotters, if any) from the LPRS file, and prompts:

Select the default printer:

Enter a number or name (the first or fifth fields from lines of the LPRS file, as reproduced above the prompt) to identify the initial default printer for the account. The program will not allow you to enter a printer number that has not already been designated in the LPRS file. If you press RETURN without specifying a printer, printer 0 is the current printer when the account logs in to the system. Printer 0 is the printer attached to printer port 0 on the back of the main enclosure.

The program next prompts you for a list of terminals:

Enter terminal access restrictions or return for no restrictions:

This is an optional specification. If no terminals are specified, the account can log in from all terminals and modems. Otherwise the account can only log in from physical terminals specified here, identified by their device numbers or names (as they appear in the first or fifth fields of the TTYS file).

Enter a single terminal name, a list of terminals separated by commas, or list of accessible terminal numbers with commas and hyphens. The following example allows the account to log in from physical terminals 1, 2, 3, and 6 but **not** from 0, 4, 5, or 7:

1-3, 6

The next prompt requests user area restrictions:

User areas:

This optional specification can restrict accounts to certain user areas. If you do not specify any user areas (by pressing RETURN at the prompt), the account can access **all** user areas. To impose user area restrictions, list only those areas to which the account will have access. Separate the accessible user areas by commas, or designate consecutive accessible user areas with hyphens. For example, to restrict an account to user areas 3, 4, 5, 6, 7, and 14, respond by typing:

User areas: 3-7,14

If that account tried to access user area 8, an area not included in this example, the following message is displayed:

You don't have permission to access user area 8.

The next prompt is:

Enter default program to run:

This optional entry can be used to restrict an account to a specified program. This can simplify use of the system for someone who always uses the same program. If you assign a default program to an account name, the account enters that program automatically at login, and is automatically logged out of MC-DOS when the program terminates. If you enter WS, the account will be restricted to WordStar. SC restricts an account to SuperCalc, and dBASE restricts the user to dBASE II. Users with assigned default programs can also be restricted to specific data files. For example, to restrict an account to editing the WordStar file called EXAMPLE.TXT, type:

Enter default program to run: WS EXAMPLE.TXT

At login, EXAMPLE.TXT is automatically loaded for editing. Accounts with default programs have no direct access to the operating system. To let an account access MC-DOS, press RETURN without entering a default program. NEWUSER will respond:

Comments:

This is an optional comments field, where you can enter up to 100 characters to explain your entry in the PASSWD file.

The "B" Option

If you selected (B) for a brief listing of account names, the screen will display an alphabetic list of the current account names. Here is a sample listing of account names in PASSWD:

```
betty  dan   harry  james  janet  larry  les   linda
mike   nate  pat    tom    trish
```

The "C" Option

If you selected (C) to change an account's PASSWD file entry, the program requests an account name (as in option (A)). After you specify an account name, this menu of options appears:

```
a) name
b) password
c) drive
d) user
e) printer
f) terminal access
g) user area access
h) application program
i) comments

j) no more changes for this user.

What to change?
```

Specify the letter corresponding to the option of your choice. The NEWUSER program will display the PASSWD listing for the account that you specify, so that you can modify anything that should be changed. When you have finished making changes for this account, press (J) to return to the main menu.

The "D" Option

If you choose (D) to delete an account, you will be prompted to supply the account name that you wish to delete. If you supply an account name that has not already been created on the system, the NEWUSER program will display the following error message:

```
No such user
```

Suppose that you want to delete "james" from the PASSWD file. After you have supplied the account name, NEWUSER will ask you to confirm the deletion:

```
Deleting james  OK? (Y/N)
```

Type "Y" to verify that you want to delete this account name.

The "E" Option

If you choose (E) for a detailed list of accounts that are allowed access to your system, you will be supplied with a list of account names, their default drive and user area, and all of the restrictions, if any, that have been imposed on them. Here is a sample detailed list of account names:

```
Name     Dir    Printer Terminal Access  Application Comment
betty     c14      0          3-7              ws
dan       b2       0        2,4-9              sc
janet     b7       0
system    0a
```

The "X" Option

This option will write any modifications that you have made back to the PASSWD file on user area 0 of drive A:.

If you want to abort NEWUSER without saving any changes that you may have made to the PASSWD file, enter ^C.

If someone happens to be running NEWUSER at another terminal of the same system, or if someone is using PASSWORD to change a login password in the PASSWD file just as you run NEWUSER, the following error message will appear on your screen:

The password file is busy, try again later.

This message will appear if NEWUSER finds that the PASSWD file is already open.

PASSWD

The PASSWD file has the following format:

```
NAME:PASSWORD:DRIVE:USER:PRINTER:ACCESS:PROGRAM:USER AREAS:COMMENTS
  \    \       \    \    \       \      \       \          \
   1    2       3    4    5       6      7       8          9
```

1. **Account login name** (maximum length is 8 characters).

2. **Login password** (maximum length is 8 characters).

3. **Initial default drive** (letter designation of logical drive).

4. **Initial default user area** (ranges from 1 to 14).

5. **Initial default printer** (printer name from field 5 of the LPRS file; RETURN = printer 0).

6. **Optional selective terminal access** This is a list, separated by commas, of the terminal names (from TTYS, field 5) from which the account can log in to the system. If this field is empty (RETURN), the account can log in from any terminal.

7. **Optional default application program.** Accounts with an entry in this field are automatically entered into the program specified. The system searches for the program as if the operator had typed it at the system prompt. The account is automatically logged out after exiting from the program. The extension (file type) is not included with the program name. If this field is empty (RETURN), the account is unrestricted.

8. **Optional user area access.** This is a list of numbers, separated by commas, or pairs of numbers separated by a hyphen to designate consecutive numbers, listing the user areas that an account is allowed to access. If no assignment is made in this field (RETURN), the account has access to all user areas. If an account tries to access a restricted user area, the system will deny permission to access that user area.

9. **Optional comments field.** Up to 100 characters can be entered, showing remarks of the system manager regarding an account.

As in the LPRS, TTYS, and VTTYS files, colons (:) are used to separate the fields of the PASSWD file. The PASSWD file must not be empty. If the PASSWD file is empty or damaged, the system will lock up on startup, and you will have to contact your dealer for assistance. Your system is shipped with only one valid account name, "system," in the PASSWD file.

Here is an example of a PASSWD file:

```
guest::c:14:epson:1:submit demo:14:run demo.sub only
nancy:qJNrC8Fby3Mn:c:1:ti::::no access restrictions
modem:yT5Mo9b34Axl:b:8::7:message:8:modem bulletin board
rhonda:j9hPI92fzl9j:c:10:0:2,3:ws:10:correspondence only
sales:7BoJqvNtCEpd:d:4:ti:4,5,6:dbase prospect:12:
steve:dCTPcl54h9On:a:5:ti:2:dbase mail:5:mailing list
system:fYh63Krl2gTc::diablo:::::no restrictions
```

Note that the "guest" account in the example above has no password, but that anyone logging in by that account name is immediately entered into a SUBMIT file called DEMO.SUB. When this program is completed, the "guest" account is automatically logged out of the system. Similarly, the "modem" account is restricted to terminal device 7 and to a program called "message" that supports a bulletin board on user area 8 of drive B:. The characters that appear in the second field of the example are not actual login passwords, but the encrypted forms created by NEWUSER. Entering the characters that appear in the encrypted form of a password would not result in a successful login. This feature protects a MC-DOS system from being accessed by an unauthorized person who has seen a listing of the PASSWD file.

The UTMP and WTMP Files

The UTMP file keeps a record of which accounts are currently logged in to the system. It can be read by the WHO utility. UTMP is created automatically when the system enters multiuser

mode. WTMP is used by the TIMELOG and HISTORY accounting programs to maintain records of system use. For WTMP to keep an active record, both the PASSWD and UTMP files must exist.

To create WTMP, use the MAKE command by typing:

```
0A>MAKE WTMP
```

This will create empty files to which the system can add information as it is generated. Once in use, the contents of the WTMP file will look like the following example, which records that "david" logged in from virtual terminal #13 at 10:39 am on July 5th, 1984, and logged out at 12:18 pm the same day.

```
tty3    ldavid    13  23781039 21874 7 51984 1124
tty3    l         13  23781218471874 7 51984 1125
```

The first three WTMP fields are the **name of the physical terminal** (from the fifth field of TTYS); the **relative virtual terminal number** (equivalent to /V in the 15th field of the VTTYS file, designating which page of display memory controlled the screen at login); and the **login account name.** The other fields (with blanks inserted for clarity) are described below:

```
13   2378   10   39   2   187   4    7 5 1984   1124
 \    \      \    \    \   \    \    \          \
  1    2      3    4    5   6    7    8          9
```

1. **Virtual terminal number** (The value displayed by CONSOLE)
2. **Days** since January 1st, 1978
3. **Hour** of the day (24 hour clock)
4. **Minute** of the hour
5. **Second** of the minute
6. **Julian day** of the year (1-366)
7. **Day** of the week (Sunday = 0, Monday = 1, Tuesday = 2, etc.)
8. **Date** in month-day-year format
9. **Program file link number** (A serial number, showing the number of records in the file after the login or logout recorded in the line.) As each new records is added to WTMP, this number in the first record is updated to show the new total number.

The WTMP file consists of entries that can be paired to correspond with login and logout. The account name is given in the login entry, but not in the logout entry. Use the terminal numbers as the key to "matching" user login and logout times.

```
console 0Boot     0 1674 916212130  8 11982  346
console           0 1674 916412130  8 11982  347
```

In this example, the system was booted (reset or powered up) at 9:16:21 am, on August 1, 1982, and the system entered multi-user mode twenty seconds later. When for some reason a system is reset before all the users log out, there will be unmatched login records, but the next "Boot" entry shows that any activities of such users had ended by the time that the system was rebooted.

The UTMP file has the same format as WTMP, but it does not include the 9th WTMP field, and it only records logins. The UTMP file is created automatically, and it records the account names that are currently logged on the system. When an operator gives a LOGOUT command, the login entry for that operator is deleted from the UTMP file.

The CONFIG.NET File

The CONFIG.NET file is needed only if a MC-DOS system is part of a DR Net network linking two or more MC-DOS systems. **You do not need to read this section** unless your Multiuser Concurrent DOS system is part of a DR Net local area network. A CONFIG.NET file is provided on the DR Net Networking diskette.

Just as the LPRS, TTYS, VTTYS, and PASSWD files are used to configure an individual Multiuser Concurrent DOS system, the CONFIG.NET file provides information to the network input/output system (NIOS) of a DR Net network that specifies what resources of a MC-DOS system are available to a DR Net network of multiple systems. CONFIG.NET can be created by a text editor like ED, or by WordStar in non-document mode. The CONFIG.NET file should exist in user area 0 of drive A: when the system is powered up or reset. A network utility called SETNET reads the CONFIG.NET file and makes its information available to the NIOS.

CHAPTER 3: MANAGING THE SYSTEM

Although creating CONFIG.NET requires the system manager to learn some additional terms and concepts, it greatly speeds and simplifies the tasks of creating or reconfiguring a network.

The CONFIG.NET file is used to customize the relationship of a MC-DOS system (called the "local node") to the other parts of a DR Net local area network. The system manager can use the CONFIG.NET file to define a network configuration and to specify **default resource mapping.** The CONFIG.NET file can be used to modify certain features of the initialization process. A sample CONFIG.NET file is provided with the DR Net system diskette, that can be modified with a text editor.

Listed below is a sample CONFIG.NET file:

```
password    PASSWORD              ;Node password, also default password
node        9                     ;Local node number
default     0                     ;Default node number
servers     8                     ;Number of server processes
requester   8                     ;Number of requester processes
buffers     3                     ;Number of message buffers
timeout     5                     ;Seconds till timeout
rct         8                     ;Number of RCT's
private     f: g:                 ;Private drives
drive       l: = a:               ;Map a drive
drive       m: = b:
drive       n: = c:
drive       o: = d:
drive       p: = e:
printer     5  = 0 0              ;Map a printer
queue       MXSPL = MXSPL 02      ;Map a queue
queue       SPLIN = SPLIN 02
queue       SPLOUT = SPLOUT 02
```

Network parameters of the local node are listed on the left. Their values are assigned in the middle column. The right-hand column provides comments describing the function of the data in the other two columns. For more detailed information about the CONFIG.NET, read Gifford's DR NET USER'S MANUAL.

Message Files: SYSINIT.BAN, MOTD, NETMOTD, name.REM

If a file called SYSINIT.BAN exists in user area 0 of drive A:, MC-DOS will display it above the name prompt. This "banner" can be edited with ED or WordStar (in non-document mode). An example of the contents of a banner file appears below:

```
-[Y
                GIFFORD COMPUTER SYSTEMS WOULD LIKE TO
                WELCOME YOU TO THE WORLD OF THE
                MC-DOS OPERATING SYSTEM
```

The -[Y characters on the first line above clears the screen of a TeleVideo 950, so that the message appears at the top of the screen. The manufacturer's documentation of your terminals lists the escape or control sequences to clear the screen, make words blink, reverse video, and other effects. (WordStar can write some control characters if you first type ^P in the file.)

MOTD

At every login, the system types the "message of the day," if a file called MOTD exists on user area 0 of drive A:. This file can contain anything you wish, including escape sequences to control the terminal's screen. The system manager can create and update the MOTD file with an editor such as ED or WordStar (in non-document mode).

NETMOTD

If your MC-DOS system is part of a DR Net network, an additional message file called NETMOTD is transmitted over the network when an operator logs in whose account name appears in the NETUSERS file, prior to the display of the local MOTD. For this to happen, the NETMOTD file must exist in user area 0 of drive A: of the system node. The system manager can use a text editor like ED or WordStar (in non-document mode) to create and modify a NETMOTD file.

name.REM

Each time an operator logs in, MC-DOS looks in the account's default drive and user area for a file called **name.REM**, where **name** is the login account name. If the system finds the file, it types it on the user's screen at login. These reminders are not the responsibility of the system manager, but the system manager should understand how to create them, so that others can be assisted in using this helpful feature of MC-DOS.

This is an easy way to leave yourself a reminder to do something, since you will see the reminder message each time you log in to the system. To make a reminder file, create the file with an editor such as ED or WordStar, in non-document mode, then move it to the initial default drive and user area of your login account. For example, if you use account "jane" and your initial default directory area is user area 5 of drive C:, you could type:

```
3C>PIP C:[G5]=JANE.REM[VROW]
```

Remember that if a reminder file for your account already exists in that user area, the PIP command will replace the old reminder file with the new, since in MC-DOS two files with identical names cannot coexist in the same directory area. You can also use the ERAQ command to delete a reminder file that is out of date. For example:

```
5C>ERAQ JANE.REM
C: JANE     .REM?
```

By typing (Y) or (y) the file will be deleted, returning the system prompt. You can, of course, leave reminder messages for others by using this feature (if you know their initial default directory areas), but it is more polite to send them mail with the MAIL utility.

Additional Customization Files

Additional files used to customize MC-DOS are AUTOST and various SUBMIT and renamed command files that are executed at system startup, login, logout, and system shutdown.

The AUTOST file is used during system startup as a flag to jump automatically into multiuser mode, if AUTOST exists in user area 0 of drive A:. Otherwise the system remains in single-user mode until the user at the system console gives the LOGOUT command. The system manager will find that many administrative functions, such as setting passwords, are easier to perform when MC-DOS is in single-user mode. The AUTOST file does not need to contain any information, and can be created by a MAKE command:

```
0A>MAKE AUTOST
```

To inhibit the automatic entry into multiuser mode, you can erase the AUTOST file:

```
0A>ERAQ AUTOST
A: AUTOST.  ?Y
```

The remaining customization files include SYSINIT.CMD and SYSINIT.SUB for system startup, USERINIT.CMD and USERINIT.SUB for login of all users, LOGIN.CMD and LOGIN.SUB for initial login routines, and LOGOUT.SUB for logout of individual users, DWN.CMD and DOWN.SUB for shutdown, all of which are described in **Section 2.5.** The system master diskette contains LOGIN.SAM, LOGOUT.SAM, SYSINIT.SAM, and USERINIT.SAM as examples of how SUBMIT files can be used. The command files can be used for whatever program you might want to run. For example:

```
0A>PIP LOGIN.CMD = WS.CMD
```

The system would automatically load WS at login.

3.4 System Security

Your MC-lDOS system is a shared resource, in which a community of users has access to the computer, the memory, the disk drives, the commands, the programs, the data files, the printers, and so forth. This ability to share data and equipment is a major advantage of MC-DOS, since it offers both increased functionality and cost savings compared with multiple personal computers.

In some situations, however, it may be prudent or necessary to restrict some or all users from access to parts of the system. A hospital or clinic may have medical records whose confidentiality is mandated by state or federal statute. An educational institution may seek to avoid having the modem used as an entry point by students experimenting in telecommunications. Most businesses have trade secrets, customer data, personnel files, or sensitive financial information on file that must be protected from casual, curious, unskilled, or malicious intruders.

MC-DOS has built into it optional **system security features** which are designed to prevent unauthorized users from accessing, copying, modifying, or destroying information. Some were devised by Digital Research Inc. as part of Concurrent DOS, the operating system which is the nucleus of MC-DOS. Others have been developed by Gifford Computer Systems as proprietary enhancements to MC-DOS. Together these security features provide the system manager with great flexibility in selecting whatever degree of security is appropriate to a given system. The system can be set up as a completely unrestricted resource, in which anyone can do anything; or some users can be given wide scope, while others are confined to specified parts of the system. At another extreme, every user can be permitted access only to part of the system, with most system resources restricted from use. This section outlines the security features and procedures available for Gifford MC-DOS systems.

Additional security features of DR Net networks provide flexibility in specifying the access rights of remote users to files on your drives. The system manager can dispense with any or all security options, if an unrestricted operating environment is more appropriate for your system. **Section 2.6** contains an overview of the security options available to the system manager.

An optional HISTORY file monitors all commands received by the system. This file records the terminal, login name, date, time, and command, for every command that anyone gives. The HISTORY utility makes it very difficult to probe system security without leaving evidence that may provide a basis for corrective or legal action.

HARDWARE SECURITY

Physical Location

Keep the computer in a room that can be securely locked. Never leave any system diskette in the floppy drives or accessible to unauthorized persons. Write down the passwords that protect drive labels, files, and commands, to avoid locking everyone out of the system. Never leave a record of any password, or diskettes or other media with programs or sensitive data where an intruder can find them. Never leave a terminal unattended without first exiting with the LOGOUT command.

Most MC-DOS systems are left running all the time. A user at an inactive terminal should always type LOGOUT, so that an intruder cannot use the terminal. After the LOGOUT command is given, only a user who supplies a valid account name and the unique password for that name can talk to the operating system.

The list of valid account names is kept in a file called PASSWD which is maintained by the system manager. The password associated with each name is up to eight characters long. As a user, you can change your own password, but only the system manager can initially put your name on the PASSWD file. The passwords are **encrypted**, so that they cannot be read by

displaying the PASSWD file with any text editor, TYPE, or PRINT command. It would be tedious for an intruder to attempt to discover the password for any name by entering random characters, because the number of different passwords available in MC-DOS is greater than 10,800,000,000,000,000 (ten quadrillion).

If the TIMELOG option is used, the system manager should periodically review the TIMELOG records, to make sure that users log out, rather than carelessly leave terminals logged in overnight or over weekends. TIMELOG will also show any unusual after-hours activity, or users who log in via modem, which may reflect unauthorized attempts to access files. The WHO command allows the system manager to monitor system usage in real time.

Restricting Access to Terminals

An efficient line of defense against intruders is available by limiting access to physical devices. Gifford's MC-DOS offers the system manager the option of restricting access to terminals by the login PASSWD file, which specifies the terminals that each account can use. The operating system will not allow a user, even one with a valid password, to log in from a terminal or modem unless the PASSWD file specifies that the user's account name has access to that device. A user can be given access to many or all terminals, but only the system manager can modify the terminal access authorization of any account name.

Modem Security

If a modem is permanently attached to the system, the login accounts authorized to log in via modem should be restricted to user areas where only non-sensitive files are stored. Any "public" login name valid on a modem has the least potential for trouble when a default program is also specified in the PASSWD file.

SOFTWARE SECURITY

The PASSWD file

The login PASSWD file provides a number of options which relate to system security. These options include:

1. Login account names to restrict access to the computer.

2. Encryption of login passwords to increase security.

3. Assignment of default drives and user areas to enforce the management practice of keeping users in separate work areas.

4. Assignment of default terminals, and restriction of accounts to specific terminals.

5. Assignment of default printers.

6. Assignment of default programs to restrict users to specific applications programs.

7. Restriction of any account to specified user areas.

The system manager must identify any groups who can share the same user areas, so that the PASSWD file can assign them common work areas.

The primary means of maintaining security in an MC-DOS system is through the login PASSWD file. Every account name **must** have a password, which is automatically encrypted. The PASSWD file must be kept on user area 0 of drive A:, which must have a protected label that sets PROTECT = ON. The system manager can assign to any login name a default program, to which a user is restricted. The PASSWD file can restrict the access of any account to specific terminals and user areas, and can restrict users who log in via modem to a default program. The NEWUSER and PASSWORD utilities are used to maintain the PASSWD file. Do not select passwords that can be easily guessed, nor leave written copies of passwords accessible.

Keep in mind that even these advanced features offer only limited security. They will keep honest users out of one another's files, but like physical security systems, they cannot offer absolute safety from the most sophisticated and determined intruder. The procedures in this section, however, will defeat most attempts to violate security that would have succeeded against many other computers.

If login passwords are used for security reasons, every account should have an encrypted password. An account without an encrypted password could easily become a free key to the system, since the PASSWD file cannot be read protected. By following the procedures described on page **3.4-9**, only the system manager can modify the PASSWD file.

If you believe that system security is being attacked, use the HISTORY file to see every command from every terminal. This is a drastic procedure, which significantly slows system speed.

Putting a Label on a Logical Drive

This section describes a procedure that is **mandatory** for system security. The SET command of Digital Research's Concurrent DOS permits a **label** to be put on any logical drive. The drive label is a directory entry, which like a file can be password protected. This label tells the operating system that a drive has special attributes. The attribute that is necessary for system security is the PROTECT option, which causes drive A: to recognize passwords on files. (Page 3-58 of the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE provides a detailed discussion of drive labels.)

The system manager must do the following three things:

(1) Put a label on drive A:.
(2) Put a password on the label.
(3) Set password protection ON for drive A:.

Suppose that PRIVATE will be your password on a label called "TAG." Each step can be accomplished by SET commands:

```
0A>SET [NAME=TAG]

Label for drive A:

Directory          Passwds  Stamp    Stamp    Stamp
Label              Reqd     Create   Access   Update
-----------------  -------  -------  -------  -------
A:TAG      .       off      off      off      off
```

```
0A>SET [PROTECT=ON]

Label for drive A:

Directory          Passwds  Stamp    Stamp    Stamp
Label              Reqd     Create   Access   Update
-----------------  -------  -------  -------  -------
A:TAG      .       on       off      off      off
```

The "PROTECT=ON" option causes MC-DOS to look for file passwords that the system automatically puts on the PASSWD and UTMP files on user area 0 of drive A: at startup. This enables drive A: to recognize passwords on any file. A third SET command is needed, however, to prevent the label from being modified:

```
0A>SET [PASSWORD=PRIVATE]

Label for drive A:

Directory          Passwds  Stamp    Stamp    Stamp
Label              Reqd     Create   Access   Update
-----------------  -------  -------  -------  -------
A:TAG      .       on       off      off      off

Password = PRIVATE
```

Do not write this procedure into an SYSINIT.SUB file, or anyone who looks at SYSINIT.SUB can learn the password on your drive A: label. A label on a logical drive of a hard disk or floppy diskette persists until the disk is reformatted.

A password **must** be set on the drive A: label to keep anyone but the system manager from setting the PROTECT attribute OFF. To remove file protection (for example, to make an archival backup), type:

```
0A>SET [PROTECT=OFF]
```

The system will prompt:

```
Password?
```

Now type PRIVATE, the password of the drive label. Your keystrokes will not appear on the screen, which shows:

```
Label for drive A:
```

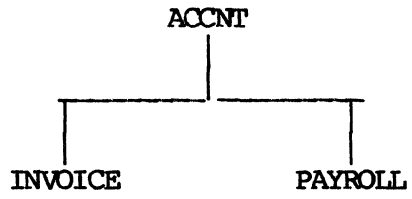
Directory Label	Passwds Reqd	Make XFCBs	Stamp Create	Stamp Access	Stamp Update
A:Label	off	off	off	off	off

This command does not erase the label. There is no reduction in system speed if a logical drive has a label with all attributes OFF. Neither ERA nor ERAQ can delete a label.

Note that "PRIVATE" would be an unwise choice of password for your actual label, since anyone who finds this manual might read these pages, test PRIVATE as the password, and be able to access protected files on drive A: by setting protection OFF.

Protecting Programs

Most application programs that call on files do not know how to use passwords. You might decide, for example, to protect with a file password parts of an accounting software package, or all of the programs, so that unauthorized users cannot see or change information. Suppose that processing your accounts requires you to invoke a program called ACCNT, which asks some questions, and then runs other programs called INVOICE and PAYROLL. When the sub-programs finish, they run ACCNT again.



For this example, assume that INVOICE and PAYROLL must be run by the ACCNT program, and cannot be run directly. To prevent some users from running any of the programs, put a password on ACCNT (such as LARK). To run ACCNT, then, you must give the password on the command line:

```
0A>ACCNT;LARK
```

To allow a user to run INVOICE but not PAYROLL, put a password on the PAYROLL program that is different from the one you put on the ACCNT program. The ACCNT program then cannot run PAYROLL unless you set your default password to match the password on the PAYROLL program. For example, if the password on PAYROLL is ROBIN, and the password on ACCNT is still LARK, use the following two steps to run PAYROLL:

```
0A>SET [DEFAULT = ROBIN]
Default Password set to ROBIN

0A>ACCNT;LARK
```

When the ACCNT program calls the PAYROLL program, it matches the password on the program with the default password. If a user tries to run PAYROLL from ACCNT without the correct default password, the system gives a BDOS error message and aborts the program. Default passwords vanish when a system is reset, but the file passwords remain as long as the XFCB and the file label support them.

File Security

Files can be given any one of three levels of protection by assigning a password. A **read restriction** can be assigned where a password must be given to read or write to a file. A **write restriction** can be assigned so a file has unlimited read status but restricted write status. And a **delete restriction** can be assigned so that the file can be modified, but the name cannot be changed and the file cannot be deleted.

Use file passwords to protect data or program files from being read or updated by unauthorized users. File password protection is discussed in Section 7.2 of the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE and page **3.4-7** of this chapter. Read also the SET command in **Chapter 2.**

Before you assign passwords to files, you should be aware that your system, as well as most applications programs, does not know how to use files with passwords on them. You cannot specify the password for a file when you run WordStar, unless you specify a default password for your terminal.

Applications programs can read password protected files if you assign a default password to a terminal. For example, to edit the file called ACCOUNTS.TXT with WordStar, if the password on the file is XACTO, give the commands:

```
0A>SET [DEFAULT = XACTO]

Default Password set to XACTO

0A>WS ACCOUNTS.TXT
```

Since you can have only one default password at a time, you should use the same password on all of your data files. This eliminates the need to change your default password often.

Digital Research provides the option for limiting access to any mass storage device. The SET command can assign temporary read-only (RO) status to any logical drive, so that its files can be read but not modified. The SET command can also assign RO status to specific files, or to groups of files through the * and ? wildcard symbols. This feature is useful to keep files from being accidentally erased, rather than to keep unauthorized users from reading or copying files.

MC-DOS also allows any logical drive to be marked with a protected **label**, a directory entry that can cause the system to check for passwords on individual files. On a drive whose label activates password protection, a file can be given a password that limits the access to the protected file, except by users who type the password. Progressively restrictive levels of file password protection are available, including protection against deletion or renaming, protection against overwriting or modifying, and protection against reading or copying. The higher

levels include all the restrictions of the lower levels. The "read protect" level is of importance in protecting the most sensitive files. How to create a protected drive label to support file passwords is described earlier in this section.

Use of file passwords has some drawbacks, however. Since many commercially available programs do not provide for file passwords, processing password-protected data or invoking protected programs may be cumbersome and may require more intervention by the operator. (File passwords can be useful in an MC-DOS or DR Net network environment where several users are working on the same programs or documents: Setting a file with a password means that someone is currently revising a copy of the file, and that others should wait for the unprotected revision before adding their own improvements. This avoids creating different "latest versions" of the same file.)

The approach to system security which Gifford recommends, however, requires that the system manager set a protected label on logical drive A:. The operating system automatically puts passwords on the systems files on drive A: that maintain system security.

To enforce user area restrictions, the system itself puts file passwords on the PASSWD and UTMP files, but you must put a label on drive A:, using SET with PROTECT=ON.

Repeat this step on any drive that has a copy of the PASSWD file. Do not keep DDT86 or hard disk formatting utilities on your hard disk. Put an encrypted password on every valid login name in the PASSWD file. Use SET to assign SYS and RO status to programs on user area 0 of drive A:.

You have the option of putting password-protected labels on other logical drives where sensitive data are stored or processed. These enable you to set passwords on important files, on high level languages, and on powerful commands such as DDT86 and SET. Sensitive files on archival diskettes should be read-protected by file passwords, with a read-protect password set on the label of the diskette.

Even if some files are individually protected, the system manager should **not** put file passwords on the PASSWD, NEWUSER, and UTMP files on drive A:. You can use file passwords to read protect the SET command and the DDT86 program; and to write protect LPRS, command files, and data files that users do not need to modify; and to delete protect other system files. Do not put write-protect passwords on .MBX files. Do not put read-protect passwords on the AUTOST, MOTD, TTYS, and .REM files.

Optional File Passwords

If a logical drive has a label that has the PROTECT attribute ON, then the SET command can be used to put a password on any file on that drive. A password can be up to eight characters long. The SET command does not distinguish between upper case and lower case letters in passwords. For example, to put the password "HIDDEN" on the file "FINANCE.DOC," type:

```
0A>SET [PROTECT=ON]
0A>SET FINANCE.DOC [PASSWORD = HIDDEN]
```

Now FINANCE.DOC cannot be accessed without presenting the password HIDDEN. If no label set with PROTECT=ON has been SET on your current logical drive, you cannot put a password on a file or on a terminal.

The PROTECT option of the SET command specifies the degree of protection that a password provides. Three levels of file protection (READ, WRITE, and DELETE) are available, plus a fourth, NONE, that erases the file password. These modes can be defined in terms of the file operations that can be performed without using the file password:

Value of PROTECT:	READ	WRITE	DELETE	NONE
Can you read the file?	NO	YES	YES	YES
Can you copy the file?	NO	YES	YES	YES
Can you write on the file?	NO	NO	YES	YES
Can you erase the file?	NO	NO	NO	YES
Can you rename the file?	NO	NO	NO	YES

If you do not specify any PROTECT mode when you set a file password, the operating system selects the READ mode, the highest level of protection. To put a write protect password on the file in the last example, you would type:

```
0A>SET FINANCE.DOC [PASSWORD = HIDDEN, PROTECT = WRITE]
```

If you decide to use file passwords, it may be efficient to assign the same password protection to groups of files that are used in the same way. A default password on a terminal can simplify data processing procedures when protected files have a common password. By defining groups of file names in the usual wildcard notation (*, ?) of the PIP, DIR, and SDIR commands, you can give a group of files the same protection level or a common password with a single command. For example, the following protects from erasure all of the dBASE data files whose names begin with the letter B.

```
0A>SET B*.DBF [PASSWORD = INDELIBL, PROTECT = DELETE]
```

This will require the password "INDELIBL" to erase any of the .DBF files beginning with the letter B, such as:

```
0A>ERA  BOX1917.DBF;INDELIBL
```

The use of passwords on data files, on powerful commands and programs like SET, DDT86, or languages must be guided by the perceived risk and potential consequences of accidental or deliberate erasure, modification, or copying. An alternative to password protection for a "dangerous" file like DDT86 or any compiler is to erase it, keeping a copy on a floppy disk that is out of reach. **Never** put a password on the PASSWD or UTMP files on drive A:, since the operating system automatically assigns those files a password. Drive A: must have a label with PROTECT set ON.

If the security procedures recommended in this manual are followed, particularly the user area access restrictions of PASSWD, it is ordinarily not necessary to put passwords on files. There is no penalty, however, for putting a protected label and file passwords on archival diskettes that hold sensitive files, unless you forget the password; the consequences of forgetting a password can be serious.

Restricting Access to User Areas

Although MC-DOS supports file passwords on individual program and data files, Gifford's recommended security procedures for MC-DOS systems are based on restricting individual accounts from access to the directory areas where the files are stored. Access to the sixteen directory areas of an MC-DOS system is controlled by the system manager through the login PASSWD file. The system manager specifies in the PASSWD file the user areas that will be accessible to each account name, grouping together in the same areas accounts who need to share files. The PASSWD file controls whether an account can access a directory area, but file passwords, and a password on the label of drive A:, prevent anyone but the system manager from modifying PASSWD. Unless you need to keep someone with whom you share a user area from accessing a file, you do not need file passwords, because unauthorized users cannot access your directory areas to rename, erase, modify, copy, or read your files.

A unique Gifford MC-DOS system security feature that enforces this restriction on directory areas is a modified **command interpreter** (the process that responds to your keystrokes when the MC-DOS system prompt is displayed). Any attempt to override the PASSWD file restriction of user areas (for example, by a PIP, USER, ERA, or ED command affecting user area 6, by someone without a 6 in the proper field of the PASSWD file) will fail, and the system will respond with the message:

You don't have permission to access user area 6.

Exceptions to this restriction are that any account can execute command files in user area 0 that have SYS and RO attributes (and no password protection), and that all accounts can send or receive electronic mail. For MAIL files to be secure from anyone but the addressee, only the system manager should have access to user area 15 of the system node, where the .MBX and AP files keep the mail. As long as the label of drive A: has PROTECT set to ON, only the system manager can change the user area assignments of the PASSWD file.

DR Net Security

The security options of a single MC-DOS computer are all extended to DR Net local area networks linking many MC-DOS systems, and supplemented with additional network security features. DR Net allows users to access the drives and printers of other computers in the network, and the security options allow the system manager to specify the access rights of users.

From the user's perspective, the network is transparent: the same MC-DOS commands are used to run programs and manipulate and print files, but some of the programs, data, and printers happen to be on other systems linked by the network. Each virtual terminal (typically four per physical terminal) can be mapped onto sixteen logical drives, and the mapping of each virtual terminal is independent of how other virtual terminals of the same workstation are mapped. A special command, NET, allows users to change the mapping, so that different drives or printers can be accessed, except as restricted by security procedures. These are detailed in the DR NET USER'S MANUAL and include:

1. The number of **server processes.** A server process allows a user elsewhere in the network to access a system. If the system manager sets this at zero in the CONFIG.NET file, no remote users can access files on the system via the network.

2. The number of **requester processes.** A requester process accesses other systems. If the CONFIG.NET file specifies zero requesters, no virtual terminal on the system can use the network.

3. **Private drives.** Files on logical drives that CONFIG.NET designates as private cannot be accessed via the network.

4. **Node passwords.** If CONFIG.NET includes a password for a system, users who cannot present it cannot access the system.

5. **Account passwords.** Users elsewhere in the network can be required to present a login password matching the PASSWD file.

In addition to these specific DR Net security features, the ordinary MC-DOS user area restrictions and protection of files by file passwords can be implemented in Gifford's DR Net networks.

This discussion of the security features has so far concerned a single MC-DOS computer. In a DR Net network, many MC-DOS computers, both multiuser and single-user, are linked by ARCNET hardware and DR Net software, so that any terminal on any node of the network can access any file on any logical drive of the network, unless restricted by security procedures. Local area networks add to the task of maintaining security the potential problem of files being accessed via the network.

Fortunately, all of the security procedures described above are in effect in Gifford's DR Net networks of MC-DOS computers, including login names, encrypted login passwords, user area restrictions, and passwords on files. For example, even if a user at another node of the network is authorized to access a drive and user area of your system, a file that is SET with a read-protect file password and is stored on a drive whose label supports password protection cannot be accessed via the network, unless the file password is supplied.

In addition to the standard MC-DOS security procedures of **user area restrictions** and **file passwords**, Gifford's MC-DOS implementation of DR Net provides the system manager with several methods to restrict network access by remote users to a system or to part of a system. For more detailed information about DR Net security, refer to the DR NET USER'S MANUAL .

Powering Down

A seemingly trivial security procedure, but one which distinguishes a DR Net network from a minicomputer or mainframe that uses other similar security features, is that individual nodes of a network can be powered OFF while others remain active. A system that held very sensitive files (for example, financial files) could be powered down overnight, while other nodes of the network remained productive— a procedure that is only practical in distributed processing systems like DR Net. No one can access a hard disk that is powered OFF. The UPTIME command monitors when each node of a DR Net network was reset by pressing the red RESET button, or powered ON.

ADDITIONAL NOTES ON SECURITY

Additional Security Procedures

Two other security procedures are available that may be helpful as diagnostic aids if unauthorized access to files is suspected. If the system manager puts labels on the drives in question, and uses timestamp options on the labels, the directory will show when each file was last accessed or updated; see **Section 3.8** of this manual and Digital Research's CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE on extended file control blocks (XFCBs). Timestamp data can be used to see if unauthorized file access is occurring, and if so, when it occurred and by whom it was perpetrated. The HISTORY utility monitors the fine structure of system usage, recording every command line and the account, terminal, user area, date, and time of the command. Using HISTORY reduces the speed of the system and is normal recommended as a normal security procedure.

Contrasting Gifford's MC-DOS and Concurrent DOS Security

Controlling access to user areas represents a significant modification by Gifford of Concurrent DOS, which in its unmodified form allows unrestricted access to user areas. For example, Digital Research Inc. provides a Concurrent DOS system command, USER, for the purpose of changing areas. The DIR[S] command, the PRINT command, and the PIP command, each with the **[Gu]** option, can respectively display the names, print the contents, or copy to disk or printer all of the files in another user area, for **u** the number of any user area. **Section 2.5** describes in detail the DIR, PIP, USER, and PRINT commands. Some applications programs and languages are also capable of changing user areas. Thus, in ordinary Concurrent DOS and in most operating systems for business computers, a determined user who somehow gets on your system can use system utilities to read your directory and get copies of your files. This may be of no consequence in a "friendly" environment, but features like this pose a serious challenge to maintaining system security.

The system manager has the option of configuring a Gifford MC-DOS system as in ordinary Concurrent DOS without user area restrictions, but the optional features of MC-DOS offer an unprecedented degree of system security. MC-DOS features that enhance the standard security options of Concurrent DOS include:

1. The login PASSWD file, which restricts access to the entire system and to direrctory areas, to terminals, and can restrict an account to a single applications program.

2. Encryption of login passwords within the PASSWD file.

3. The NEWUSER utility to maintain PASSWD.

4. Automatic assignment by the system of file passwords to the PASSWD and UTMP files.
5. Modification of the command interpreter to enforce directory area restrictions.

6. WHO, TIMELOG, and HISTORY features to monitor system usage.

Limitations on System Security

Although Gifford designed MC-DOS's security options to provide the highest degree of system security available to date in any CP/M or MP/M system, the effectiveness of these security features is in the hands of the system manager. Passwords should be difficult to guess, system diskettes should not be accessible to strangers, physical access to the system should be limited, and written information describing the specific passwords that protect a system should be inaccessible to unauthorized persons. Passwords should be changed periodically. Archival files should be kept locked up in a secure location. System managers should take care not to lock themselves (and everyone else) out of the system by forgetting passwords, and should review the TIMELOG and the optional HISTORY records periodically, to make sure that neither careless nor malicious use of the system is occurring.

A critical ingredient in maintaining system security is common sense, and in its absence anything can happen. Many more problems are caused by failure to back up files (see **Section 3.8**) and by other errors by users than by malicious acts, but some systems may attract the attention of persons with unethical motivates. Gifford Computer Systems has made a determined effort to provide MC-DOS system managers with a strategy to maintain a very high degree of system security, but the effectiveness of these measures in individual cases cannot be guaranteed. Their success will depend on the care with which you implement them, on the resources available to potential intruders, and the extent to which unfriendly or unwittingly destructive users penetrate your organization or arise within it, or gain access to information stored outside your MC-DOS system. Because these are imponderables, Gifford disclaims responsibility for breaches of system security, or consequential damages arising therefrom. By following the security procedures recommended in this section however, the system manager can reduce to a very low level the risk that system security will be violated.

3.5 Timelog Utilities

Gifford Computer Systems provides dBASE II utilities to create and maintain login histories of accounts and of physical and virtual terminals on your MC-DOS system. These utilities read from the WTMP file, and write to dBASE II database files.

Use the TIMELOG.SUB file to create the TIMELOG database files, or to append it to the database if the files TIMELOGX.DBF, TIMELOGY.DBF, and TIMELOGZ.DBF already exist. From user area 0 of drive A: enter the command:

```
0A>SUBMIT TIMELOG
```

This program copies all the entries from the WTMP file, writes them to the data base, erases the WTMP file, and restarts the WTMP file with entries for the accounts that are still logged in to the system. It is best to perform this step when there are no other users on the system, so that your database is complete.

You can now use dBASE II to view the TIMELOGX.DBF file, or you can use the utilities in the dBASE II command file called TIMELOG.CMD. From user area 0 of drive A: enter the command:

```
0A>DBASE TIMELOG
```

After a sign-on message, the screen will display a menu:

```
Would you like to

1 = Add current log to previous log
    or
2 = Purge the previous log and keep only the current log

Please choose 1 or 2  (Default is 1) : :
```

Selecting 1 appends the latest timelog information to the database, while selecting 2 discards the previous data. Normally the first option is selected, unless the previous log has been saved to diskettes, or summarized in previous reports, or is no longer of interest. After either option is selected, the screen advises you to stand by. After the timelog information is processed, the following message is displayed:

You now have the option of printing a report of the raw TIMELOG datato a file. This file may then be printed or edited at your convenience. This step is not mandatory. You may skip this step by inputting "N" or you may continue by inputting "Y"

Input Y or N here : :

If you type (Y) for a raw timelog data file, the screen prompts:

Input the name of the file to which the report will be written: :

Type up to eight characters for the file name. The TIMELOG program supplies ".TXT" as the file extension. If, for example, you specify "AUG1" as the file name, the program acknowledges:

Raw TIMELOG data is now being written to AUG1.TXT

The screen next displays a menu of timelog report options:

Here are your options for timelog reports
1 = Report access by a specific account and specific device
2 = Report access by a specific account, all devices
3 = Report access by a specific device, all accounts
4 = Report access by all accounts & devices, subtotal by device
5 = Report access by all devices & accounts, subtotal by account
6 = Return to the operating system
Input your choice : :

Here **account** means the login names that appear as the first entry in each line of the PASSWD file, and **device** means the modem or physical terminal at which a user logged in to the system. To specify a terminal, give a terminal name that appears in the fifth field of the TTYS file (typically ranging from 0 through 6).

The program next asks you to choose between detailed and aggregated timelog information, and to name the report:

1 = list all access times
2 = list totals only

Your choice : :

Input the name of the file to which the report will be written: :

Each time that you request a report, TIMELOG will ask you for a file name for the report. Once the report is finished, you can use TYPE to display the report on your screen, or send it to a printer with the PIP, PRINT, or TYPE ^P command, or edit it with ED, with WordStar, or with some other text editor. Select option six (6) to exit from the DBASE TIMELOG program.

An example of a report produced by the timelog utility of MC-DOS is displayed below:

Account	**Device/Name**				
laverne	**10/tty4**	**0**	**14 Accesses**	**Total elapsed time =**	**27:40:13**

The "totals only" report above was produced in response to option (2) in the report menu, showing all of the occasions that an account called "laverne" logged in from virtual terminal 10. The column headed **Device/Name** lists "tty4 0" as the name of virtual terminal 10, indicating that this is the first screen (0 = first, 1 = 2nd, 2 = 3rd, 3 = 4th) of a physical terminal called "tty4" in the TTYS file.

Below is an example of a report prepared in the detailed "list all access times" format, this time in response to option (1) of the timelog report menu, requesting all the devices and occasions on which a user logged in to the system by presenting the "laverne" account name. The period of time included in the report reflects the date when the current database was created, up to the time when data was last appended to the database in response to the first menu (page 3.5-1) of DBASE TIMELOG.

Account	Device/Name		Log In	Log Out	Elapsed Time
laverne	10/tty4	0	09:17:13 Fri 08/03/84	16:35:02 Fri 08/03/84	7:17:49
laverne	10/tty4	0	11:12:38 Mon 08/13/84	11:56:37 Mon 08/13/84	0:43:59
laverne	10/tty4	0	13:07:18 Mon 08/13/84	16:07:03 Mon 08/13/84	2:59:45
laverne	10/tty4	0	08:49:32 Tue 08/14/84	08:56:37 Tue 08/14/84	0:07:05
laverne	10/tty4	0	08:55:25 Tue 08/14/84	09:25:34 Tue 08/14/84	0:30:09
laverne	10/tty4	0	09:12:48 Tue 08/14/84	09:46:05 Tue 08/14/84	0:33:17
laverne	10/tty4	0	09:36:48 Tue 08/14/84	09:46:14 Tue 08/14/84	0:09:26
laverne	10/tty4	0	09:45:26 Tue 08/14/84	09:46:23 Tue 08/14/84	0:00:57
laverne	10/tty4	0	09:47:16 Tue 08/14/84	11:36:06 Tue 08/14/84	1:48:50
laverne	10/tty4	0	12:57:50 Tue 08/14/84	16:43:56 Tue 08/14/84	3:46:06
laverne	10/tty4	0	11:00:42 Wed 08/15/84	16:38:36 Wed 08/15/84	5:37:54
laverne	10/tty4	0	09:16:38 Thu 08/16/84	11:20:04 Thu 08/16/84	2:03:26
laverne	10/tty4	0	11:22:26 Thu 08/16/84	13:23:51 Thu 08/16/84	2:01:25
laverne	10/tty4	0	20:38:57 Thu 08/16/84	20:39:02 Thu 08/16/84	0:00:05
laverne	10/tty4	0	14 Accesses	Total elapsed time =	27:40:13

Account	Device/Name		Log In	Log Out	Elapsed Time
laverne	11/tty4	1	11:26:39 Fri 08/03/84	16:35:02 Fri 08/03/84	5:08:23
laverne	11/tty4	1	15:59:08 Mon 08/13/84	16:53:21 Mon 08/13/84	0:54:13
laverne	11/tty4	1	11:19:57 Thu 08/16/84	11:20:04 Thu 08/16/84	0:00:07
laverne	11/tty4	1	11:22:56 Thu 08/16/84	13:24:05 Thu 08/16/84	2:01:09
laverne	11/tty4	1	13:31:57 Thu 08/16/84	14:07:16 Thu 08/16/84	0:35:19
laverne	11/tty4	1	5 Accesses	Total elapsed time =	8:39:11
laverne	All Devices		19 Accesses	Total elapsed time =	36:19:24

3.6 The HISTORY File

The HISTORY file maintains a historical record of all the commands given to the system. HISTORY is an optional file. If the file does not exist, then no historical records are maintained. Use of the HISTORY option also requires that the PASSWD, TTYS, UTMP, and WTMP files must also exist, and that the calendar/clock be functioning correctly.

The HISTORY file can be used to monitor system usage in more detail than the TIMELOG utility provides. By timestamping commands, HISTORY can document when events occurred on the system, and can help the system manager identify individuals who could benefit from guidance in the use of the system.

Because the operating system writes every command line to the HISTORY file (if it exists), HISTORY somewhat reduces the speed at which the system operates. The effect on system performance of a HISTORY file varies with the number of users, and with how they are using the system. HISTORY only records command lines, such as ERAQ or DIR, or commands that invoke applications programs. It does not record user activity within an applications program, so that WordStar commands and keystrokes, for example, are not recorded. Thus, the activities of a user who spent all day working on a spreadsheet might only be recorded in HISTORY as the line that invoked SuperCalc, while a SUBMIT file with 75 commands would make HISTORY write to the disk 75 times. The system manager can experimentally observe the extent to which HISTORY slows a particular system by comparing the speed of the system with and without this file.

To implement the HISTORY feature, give the command:

```
0A>MAKE HISTORY
```

This will create an empty file on user area 0 of drive A: to which the system can add information as it is generated. Only commands given by users who log in to the system after you give the command above will be recorded. If you erase the HISTORY file, command lines are no longer recorded by MC-DOS.

The format of HISTORY closely resembles the format of the WTMP file. An example of a line in a HISTORY file follows:

tty6 0mary 14 23952132392040 7221984 364 pip a:=e:data.fyl[vrw]

The first two fields (in this example "tty6 0") are the **name** (from file TTYS) of the physical terminal or modem, and the **relative terminal number** of the virtual terminal that issued the command. In MC-DOS a physical terminal can support up to four "screens," corresponding to virtual terminals, numbered 0, 1, 2, and 3. The next field is the **account name** that is logged in to the virtual terminal. The last field lists the actual **command line.** In the same example of a HISTORY line, but with blanks inserted for greater clarity, the other numerical fields are:

```
14   2395   21   32   39   204   0   7 22 1984      364
  \      \    \    \    \     \   \             \       \
   1      2    3    4    5     6   7             8       9
```

1. **Virtual Terminal number**
2. **Days** since January 1st, 1978
3. **Hour** of the day (24 hour clock)
4. **Minute** of the hour
5. **Second** of the minute
6. **Julian day** of the year (1-366)
7. **Day** of the week (Sunday = 0, Monday = 1, Tuesday = 2, etc.)
8. **Date** in month-day-year format
9. **Program file link number,** a serial number showing how many commands had been recorded in the HISTORY file when this command was recorded. The link number of the last command is also written in the first line of the HISTORY file.

The meaning of the line in the complete example is that from virtual terminal 14 (the first screen of physical terminal 6) someone who logged in as "mary" gave a PIP command to copy and verify a file called DATA.FYL to drive A: from drive E: at thirty-nine seconds after 9:32 pm on Sunday, July 22, 1984.

3.7 Creating and Modifying HELP Files

You can modify existing HELP files or create new HELP files. HELP files must have the file type .HLP. The file name should be the subject. For example, the MC-DOS help file has the name MCDOS.HLP. A WordStar HELP file could have the name WS.HLP.

HELP files can be created or modified with a text editor such as ED or WordStar in non-document mode. The HELP program expects subtopics to be identified with a question (?) mark. Precede each subtopic (a single word) with a question mark. The following example HELP file (PASSWD.HLP) has the PASSWD file as its subject and has three subtopics: "GENERAL," "NEWUSER," and "PASSWORD":

```
?general
     The PASSWD file defines the encrypted login password,
initial default user number, drive, printer, optional application
program, terminal access, and user area access for every valid
account name on a Gifford MC-DOS system.
?newuser
     The NEWUSER utility must be used by the system manager to
create and maintain the PASSWD file.  Only the "system" account
can use the NEWUSER program.
?password
     The PASSWORD program permits users to change their own
encrypted login passwords in the PASSWD file.
```

HELP lists all the subtopics if the option ALL follows a subject in a HELP command. For example:

```
0A>HELP PASSWD ALL

general         newuser         password
```

Each HELP file should have a subject of GENERAL in it, so that users can get general information about a given subject by typing "HELP subject." For example,

```
0A>HELP PASSWD
```

This will display information from the file PASSWD.HLP under the subtopic "general." The user will see the display:

The PASSWD file defines the encrypted login password, initial default user number, drive, printer, optional application program, terminal access, and user area access for every valid account name on a Gifford MC-DOS system.

The user can also request help on a specific topic:

```
2C>HELP PASSWD NEWUSER
```

In this case the program will display information under the subtopic "NEWUSER," and the display will look like:

The NEWUSER utility must be used by the system manager to create and maintain the PASSWD file. Only the "system" account can use the NEWUSER program.

Each HELP file has an index file associated with it, whose file type is .NDX. To create or maintain the index file, give the HELPNDX command with the file name whenever a HELP file is created or modified. Every time you create or modify a HELP file, you must run HELPNDX to update the index file. For example, if the above file is PASSWD.HLP, give the command:

```
0A>HELPNDX PASSWD
```

By making HELP files for the specific applications programs used on a MC-DOS system, and by teaching users to use the HELP facility, the system manager can give users easy access to advice on how to use the system. Note that Gifford's HELP facility in MC-DOS is somewhat different from the system of HELP files described in the CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE.

3.8 File Maintenance Procedures

Despite the versatility of computers and the wide range of programs and accessories available today, the main functions of most business computers are to create, modify, store, retrieve, print, and transfer files. The system manager must appreciate the importance and the requirements of file-related operations, so that work can be organized in a way that uses the system efficiently, protects the integrity and security of data and program files, and minimizes the risk of losing valuable files. By establishing good file maintenance procedures, and by motivating all users to cooperate in the rational use of data and programs, system performance can be maintained at a high level. Conversely, inefficient or careless file management can reduce the speed and productivity of the system, and can waste time and resources in needless searches for lost or mishandled files.

Deleting Unnecessary Files

Inexperienced users often needlessly copy files from other user areas to their own. This takes up valuable disk space, and causes all file accesses to take longer, since more directory entries must be read. The system manager should periodically check the contents of all user areas for files that exist in other places, such as copies of applications programs. To do this easily, use the command:

```
0A>SDIR [USERS=ALL,DRIVES=ALL]
```

Some applications programs (such as dBASE II and SuperCalc 86) require that the overlay and message files be on the current logical drive. Thus, you may need copies of these files on user area 0 of each drive. If they are in another user area, move them to user area 0, so that each drive has only one copy of these files. These overlay files should be set to system and read only status. This is done with the command:

```
SET *.OVR[SYS,RO]
```

The system manager should also remind users to delete backup files (files with .BAK extensions) when they are not needed. Some users of word processing software habitually leave the backup files on the disk. Erasing these files (once the original file has been backed up) yields more disk space and faster file access. Files that are no longer used should be transferred to floppy diskettes and erased from the hard disk. Use of the archive attribute in file backup may reveal directory areas where files no longer in use are sitting on the hard disk.

The system manager should provide users with an adequate supply of floppy diskettes, so that files of marginal or archival interest can be copied onto floppies with the PIP command and erased from the hard disk. The cost of diskettes is small compared to the potential degradation of system speed when megabytes of inactive files are left to clutter the hard disk.

Identifying Current Files

As mentioned in the context of security in **Section 3.4**, MC-DOS can put **file passwords** on files with the SET command. Even in environments where secrecy is not an issue, passwords can be helpful in protecting information from accidental erasure.

Another use of file passwords not directly related to security is in situations where a group of users are developing a large document or program that has been divided into component files. Unless precautions are taken, it is not uncommon for a user (or a whole group) to become confused over which is the most current version of a file, particularly if changes in one file require corresponding revisions in another. How can the situation be avoided in which two users are modifying different copies of the same file simultaneously?

One procedure is to designate one directory area as the repository of "latest" versions. Since two users should **never** work in the same directory area simultaneously, anyone modifying a "latest" version should PIP a copy to a work area, and set a password on the original in the repository. Even if the password is publicly known, it signals to others not to revise the file until the current user returns an unprotected update.

Time and datestamping is controlled by the SET and INITDIR commands. Marking files to show their time of creation or last update can assist users in recognizing current files, and noticing when another user has accessed a file. The overhead imposed by timestamping and passwords in MC-DOS is not large, so that some system managers may elect to implement these MC-DOS features as file maintenance tools.

A third alternative to file passwords or to timestamps as a means of marking the latest versions of files is the use of **comments** in files. In large projects where many programmers or writers are working concurrently, comments help in clarifying who was the last user to modify a file. Most applications programs allow the user to insert comments into a file. In WordStar, for example, a line beginning with two periods (..) appears onscreen, but is not printed. In a multiuser project, taking the time to provide comments can assist in the final assembly of a large group of files, and can reduce the likelihood of inadvertently substituting a non-current file for the latest revision.

Packing the Hard Disk

An important procedure related to overall system performance is **packing the hard disk.** A file that is updated or extended will have its parts spread out on the disk. Accessing files that are spread out like this takes longer than accessing more compact files, since the disk head must move further and more frequently to access the data. Copying all of the files to other media, erasing the original files, and then copying back to the disk again "repacks" the files into contiguous segments on the disk.

Repacking your hard disks every few months will increase the efficiency of file operations. To do this, perform a full backup of each logical drive, format the disks, and restore the backup files. Be absolutely sure that the backup worked correctly before formatting the disk. Remember that formatting a disk destroys all of the information on it. The fastest way to repack a hard disk is to have enough space on one logical drive to PIP onto it all the files of another logical drive, whose files you then erase. You can restore the files copied by PIP, and repeat the process until all logical drives have been repacked.

Reformatting a hard disk erases drive labels, so you must use the SET command to put password-protected labels on drive A:, and on any other drive that must support file passwords or time and date stamps, if you reformat a labelled logical drive.

Since PIP does not copy file passwords, after a drive has been repacked you must also use the SET command to restore the passwords on any files that have been assigned passwords, and the password on the drive label itself. (Exceptions to this are the PASSWD and UTMP files, whose passwords are automatically restored by MC-DOS. Do not set a file password on either PASSWD or UTMP.)

Backing Up Files

Apart from the speed advantages of repacking, data stored on diskettes or hard disks should be backed up (duplicated for later retrieval) at regular intervals. Although both floppy and hard disk drives are very reliable, equipment failures or accidental deletions can occur that may result in a loss of data. Potential causes of data loss are:

- Power loss
- Hardware failure
- Operator error
- Disk media failure

The more frequently you back up your data, the less work you will lose if the diskette or disk fails. Gifford Computer Systems recommends that you back up all data at least once a week. Many companies back up their data daily.

The time it takes to back up data should be thought of as insurance against loss of that data. The more critical the data, the more time may be justifiably spent in backing up files.

The procedures to select for backing up files will depend on what medium you intend to use as a backup. Most commonly used are floppy diskettes, but other options such as 4-track or 9-track magnetic tape are also available.

Backing Up Floppy Diskettes

Diskettes can be copied with a copy utility such as the MC-DOS CPY command, the CP/M COPY command (which only works under the single-user CP/M system), or PIP. Both CPY and COPY only work if both the original diskette and the intended backup diskette have the same physical format. Select a standard format, such as 1024 bytes per sector, for all diskettes.

Remember to put an adhesive label and date on copies, so that you can identify your files later if the need arises. It is good practice to give a DIR[S] or SDIR command with ^P, so that the printer will create a hard copy of the diskette directory that can be stored with the backup diskette. Backup diskettes should be stored in a safe place, according to the instructions of the manufacturer.

Backing Up Hard Disks

The most common method of backing up hard disks is to copy the files onto floppy diskettes with the PIP command. Keep in mind, though, that a 1024-byte sectored double sided floppy will only hold 1190 kilobytes of information. Files larger than this will have to be broken down into smaller files or stored on another medium such as tape. Backing up to tape is much easier than backing up to diskettes. The backup can usually be run with just a few commands. Follow the directions in the TIP OPERATOR'S GUIDE to copy files if you have a tape drive.

If you are upgrading your system from MP/M 8-16 to MC-DOS, it is IMPERATIVE that you do a total backup of your hard disk before you use the new MC-DOS version of the PIP program to copy the new system files onto the hard disk.

Regardless of the physical means used for file backup, two conceptually different backup procedures are available. The first procedure is called an **archival backup.** The archival backup uses the archive attribute to copy only those files which have been modified since the last archive backup. This saves time and storage space, since only recently modified files are copied. The archive backup should be executed once a day.

The second backup is called a **full backup.** In a full backup, all files on the system are copied onto floppy diskettes. This is a much slower process than the archive. The full backup should be executed once a week.

Keep in mind that while these two procedures are only recommendations, **backing up data is an essential process of system management.** Backing up files is the only insurance you have against loss of valuable data, should something happen to your system. Regardless of what kind of computer you have, if you do not back up your files you are asking for trouble.

The Daily Archive Backup

You should carry out all backup procedures when no other users are on the system and the system is in single-user mode. This prevents other users from modifying files while you are executing backup procedures.

Keep five sets of diskettes for your daily backup. If it takes 10 floppy diskettes to hold all the files on your system, then you will need fifty floppies. Label each set of diskettes Monday, Tuesday, Wednesday, Thursday. On Friday, you will do a full backup which will require an additional set of diskettes. The label should contain the date of the last backup, and the user area and drive designation from which the backup originated. By keeping several sets of backups, you have a "historical" collection of files. If for some reason your system were to go down, you would never lose more than one day's work.

Copy the files to your archive diskette with the PIP command using the archive option. You will have to give a PIP command for every backup of your drive and user area. (In a backup to tape, the DSKRESET commands are not needed, and TIP or NTIP is substituted for the keyword PIP.) The form of the command is:

```
A>PIP x:=y:*.*[VROWA]
       \  \
        \  source drive of files to be copied (hard disk)
         \
          destination where copy will be made (floppy drive)
```

You must replace the logical drive designations x: and y: above (x: for destination and y: for source) with drive designations that match your system.

Note that for an archival backup the PIP command line includes the A option, which specifies copying only those files which have been created or modified since the archive attribute was last set ON. The PIP command with the archive option turns ON the archive attribute for each file copied. Any file modified during the week will get its archive attribute set OFF. If this is the **first time** that you are doing a backup, set the archive attribute of all files OFF with the SET command, as in the example below:

```
QA>SET *.* [ARCHIVE=OFF]
```

The PIP command will copy as many files as the diskette can hold. If there are more files on the hard disk than the diskette can hold, the program will stop, with an error message such as:

```
FILE12.BIL
FILE39.BIL

ERROR - DISK WRITE - FILE39.$$$
```

All of the files on the source disk that were copied before the disk filled up have their archive attribute set to ON. Put in the next archive diskette, and give the DSKRESET command:

```
A>DSKRESET
```

Repeat the same PIP command that you gave before. There will be a long pause while the PIP command examines, but does not copy, all of the files whose archive attribute has been set ON. Copying finally resumes with the the first file that was not transferred by the previous PIP command. You must repeat the

previous steps for each user area number containing files on each of the logical drives. (The sequence of PIP commands on page 3.8-10 are typical of an archival backup, but the SET commands used in a full backup are not part of archival backup procedures.)

The Weekly Full Backup

Full backups are copies of every user number on every logical drive on your hard disk. Unlike in an archival backup, which only copies new or modified files, in a full backup the archive attribute is set OFF on each file. The following steps must be followed to back up every user area and logical drive:

1. Type a SHOW command of the form:

```
0A>SHOW A:USERS, B:USERS, C:USERS, ...
```

   listing all of your logical drives. The screen will display a list of the user areas on each drive where files exist.

2. For each occupied directory area identified above, set the archive attribute OFF with a command of the form:

```
7B>SET *.* [ARCHIVE = OFF]
```

3. Copy the files to a diskette with a command like:

```
0A>PIP x:=y:*.*[VROWA]
       \  \
        \  source drive of files to be copied (hard disk)
         \
          destination where copy will be made (floppy drive)
```

You must replace the logical drive designations, x: and y: above (x: for destination and y: for source) with the appropriate designations for your system. A PIP command can copy up to as

many files as the diskette can hold. If there are more files than the diskette can hold, the program will stop with an error message, such as:

```
FILE12.BIL
FILE39.BIL

ERROR - DISK WRITE - FILE39.$$$
```

All files on the source drive that were copied before the diskette filled up now have their archive attribute set ON.

4. Put in the next backup diskette, and type DSKRESET:

```
0A>DSKRESET
```

Repeat the same PIP command as before. The PIP command will ignore files with the archive attributes set ON, and begin with the first file that was not transferred. You must repeat steps 2 through 4 for each active directory area identified in step 1. Do the last step (DSKRESET, repeating PIP) only if the diskette runs out of available space during a PIP command.

Example of a Weekly Full Backup

Be sure that no other user is on the system, then log in so that user area 0 of drive A: is your current directory area. The backup diskettes go into the left-hand floppy drive, which is designated "D:" in the example that follows. Remember to give a DSKRESET command and repeat any PIP command that is interrupted because the diskette filled up, and be sure to update the adhesive label on each diskette with a soft felt-tip marker.

For this example, assume that the system has a 21 megabyte hard disk (logical drives A:, B:, and C:), and that by giving a SHOW A:USERS, B:USERS, C:USERS command, it has been determined that only the following directory areas have files:

```
A: 0, 1, 2, 3, 6
B: 0, 1, 2
C: 0, 1, 3
```

The listing below illustrates a series of commands for a full backup of the hard disk in this example:

```
0A>SET *.* [ARCHIVE=OFF]
0A>PIP D:=A:*.*[VROWA]
0A>DSKRESET
0A>PIP D:=A:*.*[VROWA]
0A>SET B:*.* [ARCHIVE=OFF]
0A>PIP D:=B:*.* [VROWA]
0A>SET C:*.* [ARCHIVE=OFF]
0A>PIP D:=C:*.*[VROW]
0A>USER 1
1A>SET *.* [ARCHIVE=OFF]
1A>PIP D:=A:*.*[VROWA]
1A>SET B:*.* [ARCHIVE=OFF]
1A>PIP D:=B:*.*[VROWA]
1A>DSKRESET
1A>PIP D:=B:*.*[VROWA]
1A>SET C:*.* [ARCHIVE=OFF]
1A>PIP D:=C:*,*[VROWA]
1A>DSKRESET
1A>PIP D:=C:*,*[VROWA]
1A>USER 2
2A>SET *.* [ARCHIVE=OFF]
2A>PIP D:=A:*.*[VROWA]
2A>SET B:*.*[ARCHIVE=OFF]
2A>PIP D:=B:*.*[VROWA]
2A>DSKRESET
2A>PIP D:=B:*,*[VROWA]
2A>USER 3
3A>SET *.* [ARCHIVE=OFF]
3A>PIP D:=A:*.*[VROWA]
3A>SET C:*.* [ARCHIVE=OFF]
3A>PIP D:=C:*.*[VROWA]
3A>USER 6
6A>SET *.* [ARCHIVE=OFF]
6A>PIP D:=A:*.*[VROWA]
6A>DSKRESET
6A>USER 0
0A>
```

In this example four (4) floppies were filled up during PIP commands, once on the A: drive, twice on the B:, and once on the C: drive, so that a DSKRESET was needed, and the previous PIP command had to be repeated. The last DSKRESET (after copying user area 6 of drive A:) was in preparation of removing the fifth backup diskette. The frequency of DSKRESET commands in a full backup depends on the size of the files being backed up. Five floppies in all were sufficient to backup the hard disk in this example, suggesting that it was no more than 30% full.

Backing Up Read-Protected Files

You can back up files that have read-protect passwords by first modifying their drive label so PROTECT is set OFF. You must know the password of the drive label to modify the label. Suppose that "SECRET" were the password of the label of drive B:. To back up protected files on B:, type:

```
0B>SET [PROTECT=OFF]

Password?
```

Now type SECRET, the password on the label of drive B:. This causes drive B: to ignore file passwords. You can now perform a backup, or you can erase the files from drive B:, if they are no longer needed on the hard disk. After you PIP files to an archival diskette, the same command as above, but substituting "ON" for "OFF," will restore password protection to files on drive B:.

To password protect files on an archival diskette (for example, on drive E:) you must put a label on the diskette. Below is a command to assign a label called "FLOPP-21" to the floppy disk:

```
0E>SET [NAME = FLOPP-21]
```

The screen will display the label, in the format shown on **page 3.4-5.** Next you must make the label recognize file passwords by setting the PROTECT option of the label ON:

```
0E>SET [PROTECT = ON]
```

To prevent unauthorized persons from modifying the label to disable the password protection, you must also put a password on the label. (You must also be sure that you do not forget the passwords!) In this example, "PADLOCK" is the label password:

```
0E>SET [PASSWORD = PADLOCK]
```

Now the diskette can recognize file passwords that you may assign. The PIP command does not copy passwords. This means that a file that has a password on the source will no longer have any password on the destination area. If you use PIP to copy files that require password protection, be sure to use the SET command to put file passwords on the archival diskette files immediately after the PIP operation is done. Wildcard symbols can be used to put the same password on each file. For example:

```
0E>SET *.*[PASSWORD = LOCKPAD]
```

This puts the password "LOCKPAD" on every file in user area 0. If files were stored in other user areas of the diskette (for example, user areas 2 and 9) additional commands would be required to protect the other user areas that had files:

```
0E>USER 2
2E>SET *.*[PASSWORD = LOCKPAD]

2E>USER 9
9E>SET *.*[PASSWORD = LOCKPAD]
```

Note that the command to put a label on the diskette did not have to be repeated, because a single label can support file passwords or timestamps on all of the user areas of a logical drive. The SET command to assign file passwords to the files, however, has to be repeated for each user area that has files.

Restoring Backup Data

If you need to restore data from a backup tape or diskette, you can either restore all of the data at once, or restore specific files selectively. For example, if you accidentally erase a file on your hard disk, you can copy the most recent version of the file from a backup diskette or tape. To restore files or disks from floppy diskettes, use the PIP command. To restore files from tape, use the TIP program.

3.9 Automatic RAM Memory Test

MC-DOS includes an automatic RAM test which runs at the lowest priority and does not decrease the efficiency of the system. The RAM test performs non-destructive tests on each byte of memory, and reports bad memory, if it is found, on the system console.

It is very unlikely that you will ever receive any messages from the memory test program. If you want to test all your RAM memory, leave your system ON with no active users. The memory test will be completed in an hour or less if there is no activity. The memory test starts over as soon as it finishes.

If bad memory is found, the following message appears on the system console screen:

Memory failure at hhhh:hhhh data was hh data should have been hh

The hhhh:hhhh and hhs are hexadecimal numbers, described on **page 2.5-DDT86-1.** Write down **all** of the information, and call your Gifford Computer Systems representative.

3.10 Upgrading and Updating MC-DOS Systems

Keeping Your System Current

As your computing needs expand, you may wish to expand your system and your software to meet those needs. Gifford Computer Systems supplies hardware, software, and technical assistance to meet the growing needs of MC-DOS customers.

Gifford Computer Systems is continually updating the MC-DOS operating system with new features. You will be notified of new developments of the MC-DOS system in our newsletter, the **GIFFORD OBSERVER**, and you will have the opportunity to purchase updates of our operating software for a reasonable fee.

From time to time Digital Research Inc., publisher of the Concurrent DOS operating system upon which Gifford's MC-DOS is based, releases new versions of Concurrent DOS that may offer new capabilities important to your computer applications. Gifford's policy has been to make new releases of Digital Research Inc. operating systems software products available to customers in a timely manner, when Gifford engineers are satisfied that the new software is adequately integrated with the physical equipment.

Gifford Computer Systems periodically updates the MC-DOS manuals, and you may want to purchase new documentation when it becomes available. Please contact your Gifford representative if you are interested in expanding the hardware or software capabilities of your MC-DOS system.

Hardware Upgrades

If you have purchased additional IEEE 696/S-100 hardware components, or if you have had Gifford upgrade old components to current versions, be sure that the system is powered OFF when you install the new boards. Sections of **Chapter 1** describe how to shut down the system and how to install hardware. **Appendix B** and **Appendix C** provide information on the board arrangement and on switch and jumper settings. Never run the computer without the metal cover properly installed on the chassis.

CHAPTER 3: MANAGING THE SYSTEM

Preserving Old Operating Systems

Before you install an updated version of MC-DOS, be sure that you have a copy of **ALL** of your files on floppy diskettes (not just the system files), so that you can go back to the old software in the event that anything is wrong with your copy of the new version of MC-DOS. Label the diskette that holds the old version and keep it in a secure place offsite. It is good practice to keep in your archives a labelled copy of any operating system that ever worked on your system. Follow the procedures in **Section 1.6** to back up the new system diskette. Keep all copies of system diskettes inaccessible to intruders.

Upgrading MP/M 8-16 Systems to MC-DOS

If you are upgrading a computer to MC-DOS that formerly supported an MP/M 8-16 operating system, read the letter that accompanied your upgrade regarding any differences in switch settings or jumpers on your S-100 boards. Detailed information is provided in **Appendix C** of this manual. **Make sure that you back up ALL of your files onto floppy diskettes (not just your system files), before you copy your new system files onto the hard disk. Also, it is imperative that you use the new MC-DOS version of the PIP program to copy ALL your new system files onto your hard disk. Otherwise, the new system files will not be copied correctly.** If your computer is a Gifford MP/M 8-16 operating software release earlier than 2.1F, or CompuPro's version of the MP/M 8-16 operating system, you will need to run the special FIXPASS upgrade utility to modify your unencrypted PASSWD file. Read the description of FIXPASS in **Chapter 2** of this manual. You do **not** need FIXPASS to upgrade systems that already have Gifford's MC-DOS or releases 2.1F or 2.1G of Gifford's MP/M 8-16.

If you have a version of the MP/M 8-16 operating system from 1983 or earlier, read the section near the end of **Chapter 2** on system security to decide what level of system security you need to establish. With encrypted login passwords and restrictions on who can access various user areas, you must decide which user areas will be open to which accounts. Before you upgrade, you must put an account called **system** in your PASSWD file. The procedures for installing an upgrade are described in the next section. Read about NEWUSER in **Section 3.3** to learn how to add account names to the PASSWD file.

CHAPTER 3: MANAGING THE SYSTEM

Installing a MC-DOS Upgrade

Below are procedures to install an upgrade. Keep a copy of your older operating system handy, until you are sure that the upgraded version is installed properly. Follow these nine steps:

1. Boot the system with the older operating system.

2. Copy PASSWD, LPRS, TTYS, and VTTYS files to a diskette and store nearby. If the .HLP files have been customized, copy them too. If you have a hard disk, the customized files from your old MC-DOS that do not appear on the new system diskette will be on your hard disk. Review this chapter, particularly **Sections 3.3** through **3.7**, to be sure that the system management files are as they should be. If you have files on your old system that for some reason are superior to the corresponding new files (for example, HELP files that you have augmented with messages tailored to your users and applications, or .MBX and .REM files), these can be copied to drive A: from your archival diskette, using the PIP command.

3. Type SYNC to flush the cache buffer to the hard disk.

4. Boot the system with the newest version of MC-DOS.

5. Log in as "system" on the system console.

6. Copy the new operating system floppy diskette to user area 0 of drive A: with the new MC-DOS version of the PIP command (to determine which PIP program you're using type PIP and then RETURN) like the following, but substituting your floppy drive source designation for "y:"

   ```
   PIP A:=y:*.*[VROW]
   ```

   If you have SET a label on drive A: with protection ON, remember to set protection OFF so that you can overwrite old system files that are protected by file passwords. Remember to put a label with a password, with protection set ON, on drive A: so that your new system will be secure. Read **Section 3.4** of this chapter for information on the use of drive labels and on establishing system security.

7. Restore old files (like LPRS, PASSWD, TTYS and VTTYS, etc. that you set aside in Step 2 above) to user area 0 of drive A: of the hard disk. Do not copy the old PASSWD file if you were required to use FIXPASS. If you have a new PASSWD file that has been modified by FIXPASS, this must be used on the new system. An old MP/M 8-16 PASSWD file may lock you out of the system, if it contains unencrypted passwords.

8. Run SYNC to copy to drive A: all the new files stored in the buffer.

9. Reboot with the new MC-DOS system diskette in your left-hand floppy drive.

Remember to keep labelled copies of your old operating system diskette in a safe place.

Upgrading Systems Without a Hard Disk

The reduced speed and reduced storage capacity of a system without a hard disk generally make it impractical to install all of the features of Gifford's MC-DOS on a floppy-only system. You must choose selectively among the options. If this is an unsatisfactory situation, you should consider upgrading to a hard disk.

If you are adding a hard disk to a system that formerly lacked one, or if you are upgrading a CP/M system to MC-DOS, read **Chapter 1** of this manual and proceed as if you were installing a new system.

CHAPTER 4: TROUBLESHOOTING THE SYSTEM

These charts will assist you in getting out of trouble should your system fail to operate properly. If you find that you need to call Gifford Computer Systems, read **What You Will Need to Know to Call Gifford Computer Systems** at the end of this chapter. For systems with only one floppy drive, assume that the left-hand floppy drive is the equivalent of your single floppy drive.

Warning: Never work on the inside of the enclosures when the power is ON. Turn OFF the power and unplug the power cords before opening an enclosure.

4.1 SYSTEM WILL NOT START UP

Computer Enclosure

Normal operation is indicated by the sound of the fans and the RESET button should be lit.

Problem	Probable Cause	Remedy
RESET button not lit and fan not ON.	No electrical power to the computer.	Plug power cord into wall socket. Plug power cord into computer enclosure. Check wall outlet for power. Check circuit breaker to outlet.

Problem	Probable Cause	Remedy
RESET button not lit and fan not ON.	Circuit breaker has tripped the main power switch.	Check system for shorts; turn breaker OFF, then ON.
Fan ON, RESET button not lit.	Loose connection or bad light.	Call Gifford.
RESET button lit, fan not ON.	Fan disconnected or broken.	Call Gifford.

Floppy Disk Drives

Normal operation is indicated by a blinking light on the left floppy drive when the drive door is open.

Problem	Probable Cause	Remedy
Drive light fails to light.	Floppy drives do not have power.	Turn back panel power switch ON. Plug power cord into power socket on main enclosure or into a wall socket. Check circuit breaker on rear panel.
Drive light glows dimly.	Weak connection between drives and controlling cables.	Check that the cables are securely connected between drives and Disk 1 or Disk 1A board. Check that Disk 1 or Disk 1A board is seated in the motherboard.

Problem	Probable Cause	Remedy
Both drive lights glow brightly.	Cable is installed upside down.	Disconnect, turn over the connector, and reconnect.
Drive light continues to blink when the drive door is closed.	The drive belt has detached.	Call Gifford.
Drive light blinks, drive heads click, but system will not load.	Damaged system diskette or the wrong diskette.	Try another system diskette.

Other Problems Loading MC-DOS

Problem	Probable Cause	Remedy
System begins to load, but stops in mid process.	Bad RAM board.	Call Gifford.
System sounds as if it booted, but no message on console.	No power to terminal.	Plug in terminal and turn ON power
	RS-232 cable not connected properly.	Check cable connections.
	Terminal incorrectly set.	Check baud rate and word size settings.

Hard Disk Drives

Problem	Probable Cause	Remedy
Drive does not initialize.	Drive does not have power.	Check power switch. Check power cord.
	Drive heads not unlocked.	Unlock heads.
	Data or control cables not secure.	Check that cables are connected properly and are secure.

4.2 SYSTEM STARTS UP, BUT DOES NOT OPERATE PROPERLY

Runs slowly or loses data sporadically.

Problem	Probable Cause	Remedy
System runs slowly.	Some terminals or printers are not ON or not connected.	Reconnect any disconnected terminals and printers and turn ON power to all terminals and printers.
	RS-232 cables are using pin-4.	Disconnect any wires from pin-4.
	The hard disk needs reformatting.	Call Gifford.
The computer occasionally loses data or files.	Bad RAM board.	Call Gifford.

Specific programs do not work correctly.

Problem	Probable Cause	Remedy
CPY does not work.	Source and destination diskette formats are not identical.	Use the appropriate diskette or use PIP to copy files.
Applications programs such as WordStar or SuperCalc do not work correctly.	The program is not installed properly for your system.	Read the manuals. Try running the program under a single-user operating system. If you purchased the software with your system, call Gifford.
Programs work on all but a few terminals.	TTYS file has been damaged.	Check TTYS file. For GCS-80 terminals, check that handshaking is set and scroll is ON.

4.3 WHAT YOU WILL NEED TO KNOW TO CALL GIFFORD COMPUTER SYSTEMS OR YOUR DEALER

Gifford Computer Systems fully supports all of the hardware and software we sell. Call the Customer Service Department between 9:00 A.M. and 5:00 P.M. (Pacific time) for help. Please have the following items handy when you call:

Your system number. This number is listed on the invoice you received with your system.

The version of your operating system. This is listed on the label of your master diskette.

An exact description of what is not working. Also list whether the product was working before, and what has changed between the time that it did and did not work. Have this manual with you when you phone us, in case we need to refer you to specific pages in the manual.

We will attempt to rectify the problem with you on the phone. If we are unable to solve your problem immediately, we will take your name and number and get back to you as soon as we can.

If it is necessary to repack your system, make sure that it is packed as it was originally shipped. Take special care in repacking the hard and floppy disk drives. It is important that the disk drives be secured before moving or shipping. Install the cardboard inserts in the floppy drives, and secure the heads on hard disk drives.

4.4 MC-DOS BUG REPORT

This form should be used to report problems with MC-DOS or with the manual itself to Gifford Computer Systems. Every report will be reviewed by our Customer Support Department.

Name:

Company:

Address:

City, State, Zip Code:

Telephone:

MC-DOS System Number:

MC-DOS Version:

State the exact problem and symptoms that you encountered with MC-DOS below:

APPENDICES

CONTENTS

APPENDICES

APPENDIX A

A. Hardware Requirements for MC-DOS

The following CompuPro products are necessary for a basic MC-DOS system:

Mainframe:	An IEEE 696/S-100 motherboard and power supply.
Processor:	Macrotech MI-286 (dual CPU with 8-bit Z80H and 16-bit 80286); or alternatively, CompuPro CPU 8085/88 (dual CPU with 8-bit 8085 and 16-bit 8088); or CompuPro CPU 8086/87 (16-bit).
Support:	System Support 1 (one serial port for system console, two interrupt controllers, a real-time calendar and clock, three interval timers, RAM and EPROM).
RAM Memory:	RAM 16, 21, 22, or 23 (high speed static RAM). 192 kilobytes of RAM is required for the operating system. 64 kilobytes is needed for each user running 8-bit programs. Users running 16-bit programs require somewhere between 16 kilobytes and 1 megabyte depending on the nature of the program. The MC-DOS operating system allocates up to 128 kilobytes per 16-bit user as a default value. This value can be changed with the SETMEM command or by generating a new system file with the GENCCPM menu-driven program. Systems with hard disks can utilize an additional 0K, 32K, or 64K of RAM for a hard disk cache buffer.
I/O Control:	Interfacer 3-8 board (terminal, printer, and modem support).
Mass Storage:	Disk 1 or Disk 1A floppy disk controller (with one or two 8-inch double sided, double density floppy disk drives). From one to four hard disk drives, each holding at least 20 megabytes, is recommended for systems which support three or more users.
Terminals:	One for each logged-in user. Extra display RAM will support multitasking virtual terminals.

APPENDIX A

Each user on the system requires a serial interface port to connect a terminal or modem. Additional serial ports can be added to the system using CompuPro's interfacer boards.

Systems with tape backup units (4-track or 9-track) require a tape controller on the bus, such as the Alloy controllers.

DR Net local area networks require a network controller on each computer within the network, to support Datapoint's ARCNET network hardware protocols. Networks with more than two nodes will require active or passive repeaters to link the coaxial cables that connect the network controllers.

Computers like the IBM PC or IBM PC/XT must have 256K or more of RAM, and must include a network controller that supports ARCNET protocols to be in DR Net network.

APPENDIX B

B. Description of the Board Arrangement on IEEE 696/S-100 Motherboard

The tables in **Appendix C** document the switch and jumper settings for the MC-DOS operating system. Unless you install a different CPU, it is unlikely that you will ever have to change any of the switches or jumpers. This information is provided in the event that switch settings are accidentally changed.

The switches referred to are located on the circuit boards inside the main enclosure. You will need to turn the system OFF and unplug the computer from its power socket before you remove the cover of the enclosure. Once the cover is removed, you will find the circuit boards plugged into a series of sockets on the motherboard. Each socket has 100 contacts (which is the origin of the IEEE 696/S-100 bus designation).

All of the boards must be inserted so that the component side of the board faces the front of the computer. The boards can be placed in any order. Gifford Computer Systems installs them in the following order, counting from the front:

Slot #	Board description
1	
2	CPU
3	
4	RAM
5	RAM
6	RAM
7	RAM
8	RAM
9	RAM
10	MDRIVE/H
11	MDRIVE/H
12	DR NET NETWORK BOARD
13	
14	HARD DISK CONTROLLER

APPENDIX B

Slot #	Board description
15	
16	DISK 1 or DISK 1A
17	
18	SYSTEM SUPPORT 1
19	
20	INTERFACER 3-8

(CompuPro introduced motherboards during 1984 that have 21, rather than 20 bus sockets.)

Each board has its description on it. The switches are likely to be the blue components with tiny white switch levers on them. The switch can be identified by looking on the board near the switch for the component description. The component descriptions are printed in yellow on the CompuPro boards. If you have trouble locating a board or the switches, you can refer to the user manuals for each board. These are thin blue books with photographs of the boards on the covers (for CompuPro boards), similar books for Macrotech or Alloy boards, and a small-format booklet for the Konan controller. The user manuals include component layouts that indicate the position of the switches and jumpers.

C. Documentation of Switch Settings and Jumpers

Unless otherwise noted, the settings described below are valid for the current versions of MC-DOS (V1.00 and later).

CompuPro CPU 8085/88

Switch 1		Switch 2		Switch 3	
OFF	ON	OFF	ON	OFF	ON
1	==	1 ==		1 ==	
2	==	2 ==		2	==
3	==	3 ==		3 ==	
4 ==		4 ==		4 ==	
5 ==		5 ==		5 ==	
6 ==		6 ==		6 ==	
7	==	7 ==		7 ==	
8 ==		8 ==		8 ==	

CompuPro CPU 8086/87

Switch 1 and switch 2: all paddles (subswitches) are OFF.

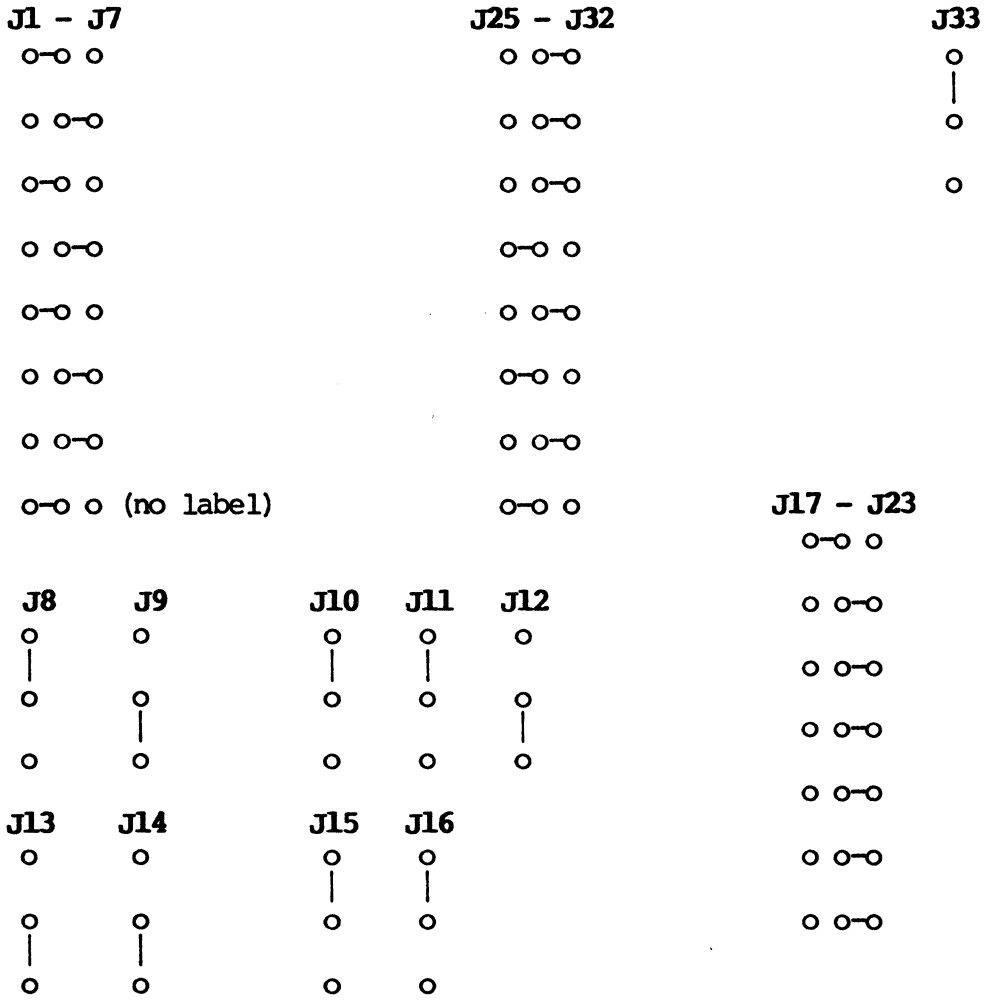
Switch 3		Switch 4		Switch 5	
OFF	ON	OFF	ON	OFF	ON
1 ==		1 ==		1 ==	
2 ==		2 ==		2 ==	
3	==	3	==	3 ==	
4	==	4	==	4 ==	
5 ==		5	==	5 ==	
6 ==		6	==	6 ==	
7	==	7	==	7 ==	
8 ==		8	==	8 ==	
		9	==	9 ==	
		10	==	10 ==	

J8-jumpered.

APPENDIX C

Macrotech MI-286 (Z80/80286 Dual Processor)

Note that there is an eighth, unlabeled jumper in the upper left set of jumpers, below J7, and that the location labelled J24 has no jumper. The following jumper settings should be used:



APPENDIX C

For CompuPro 8086 CPU or Macrotech MI-286 CPU, remove pin-4 of IC U28 on the System Support 1 from its socket. For an MI-286, 8085/88, or 8086, move slide jumper J17 on the Disk 1 from position A-B to B-C:

CompuPro Disk 1, J17

CompuPro System Support 1

	Switch 1			Switch 2			Switch 3	
	OFF	ON		OFF	ON		OFF	ON
1	==		1		==	1	==	
*2		== <See note	2		==	2	==	
*3		== <below *	3		==	3	==	
4		==	4		==	4	==	
#5	==		5	==		5		==
6	==		6	==		6	==	
7		==	7	==		7		==
8	==		8	==		8	==	

* If the CPU is a Macrotech MI-286, subswitches #2 and #3 of Switch 1 on the System Support 1 board must be OFF, rather than ON. (The position illustrated above is required for the CompuPro CPU 8085/88 and CPU 8086/87).

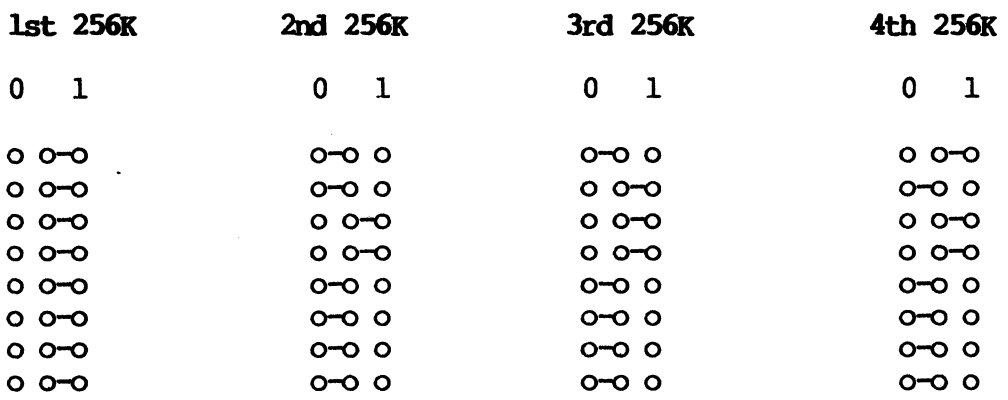
\# For systems with one megabyte of RAM, subswitch #5 of Switch 1 should be set ON.

Jumpered Settings:
J1 Cable connector on upper card edge: Serial port connection.
J2 Dip socket near upper edge: Insert a dip shunt.
J3 Battery connection on upper edge: Plug auxiliary battery cable into this connector, noting the "+" designation on the board.
J8 Dip socket in lower row of ICs: Insert a dip shunt.
J13 Three-pin jumper: Insert a slide jumper onto prongs 8 and C.
All remaining jumpers are left unconnected.

APPENDIX C

Macrotech 256K-512K Static RAM Memory Board

The jumper settings for the jumper marked J3 on the 256K-512K Macrotech memory board should be as follows:



Refer to the Macrotech technical Addressing Addendum 1.1 for documentation on alternate memory value jumper settings.

APPENDIX C

RAM 16 Memory Boards

Switch Settings for S1:

1st 64K		2nd 64K		3rd 64K		4th 64K	
OFF	**ON**	**OFF**	**ON**	**OFF**	**ON**	**OFF**	**ON**
1	==	1	==	1	==	1	==
2	==	2	==	2	==	2	==
3	==	3	==	3	==	3	==
4	==	4	==	4	==	4	==
5	==	5	==	5	==	5	==
6	==	6	==	6	==	6	==
7	==	7	==	7 ==		7 ==	
8	==	8 ==		8	==	8 ==	
0-FFFF		10000-1FFFF		20000-2FFFF		30000-3FFFF	

5th 64K		6th 64K		7th 64K		8th 64K	
OFF	**ON**	**OFF**	**ON**	**OFF**	**ON**	**OFF**	**ON**
1	==	1	==	1	==	1	==
2	==	2	==	2	==	2	==
3	==	3	==	3	==	3	==
4	==	4	==	4	==	4	==
5	==	5	==	5	==	5	==
6 ==		6 ==		6 ==		6 ==	
7	==	7	==	7 ==		7 ==	
8	==	8 ==		8	==	8 ==	
40000-4FFFF		50000-5FFFF		60000-6FFFF		70000-7FFFF	

APPENDIX C

RAM 21 Memory Boards

Switch Settings for S1:

1st 128K		2nd 128K		3rd 128K		4th 128K	
OFF	**ON**	**OFF**	**ON**	**OFF**	**ON**	**OFF**	**ON**
1	==	1	==	1	==	1	==
2	==	2	==	2	==	2	==
3	==	3	==	3	==	3	==
4	==	4	==	4	==	4	==
5	==	5	==	5	==	5	==
6	==	6	==	6 ==		6 ==	
7	==	7 ==		7	==	7 ==	
8 ==		8 ==		8 ==		8 ==	
0-1FFFF		20000-3FFFF		40000-5FFFF		60000-7FFFF	

RAM 22 Memory Boards

Switch Settings for S1:

1st 256K		2nd 256K		3rd 256K		4th 256K	
OFF	**ON**	**OFF**	**ON**	**OFF**	**ON**	**OFF**	**ON**
1	==	1	==	1	==	1	==
2	==	2	==	2	==	2	==
3	==	3	==	3	==	3	==
4	==	4	==	4	==	4	==
5	==	5	==	5	==	5	==
6	==	6	==	6 ==		6 ==	
7	==	7 ==		7	==	7 ==	
8 ==		8 ==		8 ==		8 ==	
0-3FFFF		40000-7FFFF		80000-BFFFF		C0000-FFFFF	

APPENDIX C

HDC/DMA Hard Disk Controller

Versions V1.00 or later of MC-DOS:

Jumper Settings:

J5 to 0, J3 shorted.

HDCA Hard Disk Controller

Switch 8

	OFF	**ON**
1		==
2	==	
3	==	
4	==	
5	==	
6	==	
7	==	
8	==	

Jumpered Settings:

Versions 1.00 or later:

Interrupt jumper: A to B to 0.

APPENDIX C

Konan DGC-100 Hard Disk Controller

Jumper Settings:

Near R41: leftmost jumper connected, others unconnected.

Above RP3: second and fifth from left connected, others unconnected.

Jumper "DMA": middle and bottom connected.

Jumper "CBA": middle and bottom connected.

Disk 2 Hard Disk Controller

Version V1.00 or later of MC-DOS:

	Switch 1			Switch 2	
	OFF	ON		OFF	ON
1	==		1	==	
2	==		2	==	
3		==	3		==
4	==		4		==
5		==	5	==	
6	==		6		==
7		==	7		==
8	==		8		==
9	==				
10	==				

The interrupt jumper should be moved to position 1 of jumper area J7-J15.

APPENDIX C

Selector Channel

	Switch 1	
	OFF	**ON**
1	==	
2		==
3	==	
4	==	
5	==	
6	==	
7		==
8		==
9		==
10		==

Disk 3 Hard Disk Controller

	Switch 1		Switch 2	
	OFF	**ON**	**OFF**	**ON**
1	==		1 ==	
2		==	2	==
3		==	3	==
4	==		4 ==	
5		==	5 ==	
6		==	6 ==	
7		==	7 ==	
8		==	8 ==	

RP1 Dip socket: Insert a dip shunt.

J1-J4 Top edge of card: Cable connection to drives.

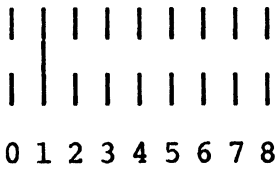
J5 Top edge of card: Cable connection to drives.

J7 Three jumper pins: Insert a jumper slide between C and B.

J8 Three jumper pins: Insert a jumper slide between A and C.

APPENDIX C

J9



J10

Jumper should be set in top position.

Disk 1 Floppy Disk Controller

	Switch 1			Switch 2	
	OFF	**ON**		**OFF**	**ON**
1		==	1	==	
2	==		2		==
3	==		3	==	
4	==		4	==	
5	==		5		==
6	==		6		==
7	==		7		==
8	==		8		==

On switch 2, subswitch 1 controls the baud rate of the terminal attached to the System Support 1 board. Subswitch 2 of switch 2 controls the number of stop bits used by the console attached to the System Support 1 board. This is not the way that standard CompuPro software uses these switches.

Switch 2

	ON	**OFF**
Subswitch 1	9600 baud	19200 baud
Subswitch 2	2 stop bits	1 stop bits

Interrupt jumper J4: installed.

With either an 8085/88, an 8086 CPU, or a MI-286, slide jumper J17 should be moved to position "B," connecting the lower two posts of J17.

Disk 1A Floppy Disk Controller

The switch settings documented below are for the Disk 1A floppy disk controller. The Disk 1A will control both 8-inch and 5 1/4-inch floppy disk drives.

	Switch 1			Switch 2			Switch 3	
	OFF	ON		OFF	ON		OFF	ON
1		==	1	==		1	==	
2		==	2	==		2		==
3		==	3	==		3		==
4		==	4	==		4		==
5	==		5	==		5		==
6		==	6	==		6	==	
7	==		7	==		7	==	
8	==		8		==	8		==

Subswitch 4 should be set ON. Subswitch 5 on Switch 1 controls the baud rate of the terminal attached to the System Support 1 board (called Console 0). Subswitch 6 on Switch 1 controls the number of stop bits used by the terminal attached to the System Support 1 board.

Switch 1	ON	OFF
Subswitch 5	9600 baud	19200 baud
Subswitch 6	2 stop bits	1 stop bit

APPENDIX C

JUMPER SETTINGS

Jumper Position	Shunt Position
J1	5
J2	5
J3	8
J4	8
J5	OPEN
J6	AC
J7	BC
J8	AC
J9	AC
J10	4
J11	Shunt Installed
J12	OPEN
J13	OPEN

Interfacer 3

Switch 1

	OFF	**ON**
1	==	
2		==
3		==
4		==
5	==	
6		==
7		==
8		==

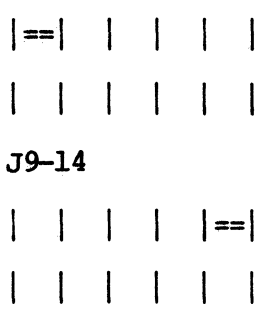
For a second Interfacer 3 board, subswitch 7 of switch 1 should be set OFF.

Jumpered Settings:

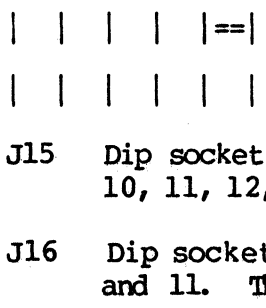
J1 Upper left dip socket: Insert a dip shunt. Consult the Interfacer 3 manual for hardware handshaking.
J2 Dip socket to right of J2: Insert a dip shunt.

APPENDIX C

J3-J8



J9-14



J15 Dip socket one row up from bottom edge: Connect pins 5, 9, 10, 11, 12, 13, 14, 15, and 16. This is an interrupt jumper.

J16 Dip socket below J15: Connect pins 1, 2, 3, 4, 5, 6, 7, 8 and 11. This is an interrupt jumper.

J17 Three-pin jumper between U30 and U31: Jumper the top two pins.

The remaining jumpers are unconnected.

Interfacer 4

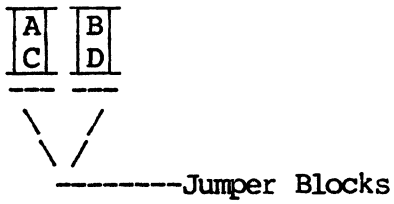
Versions V1.00 or later of MC-DOS:

Switch 1: all paddles (subswitches) are OFF.

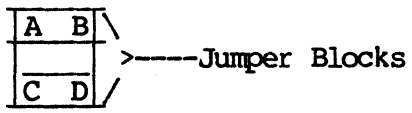
	Switch 2			**Switch 3**	
	OFF	**ON**		**OFF**	**ON**
1		==	1	==	
2		==	2	==	
3		==	3	==	
4	==		4	==	
5		==	5		==
6		==	6		==
7		==	7		==
8	==		8		==
9		==			
10		==			

APPENDIX C

J26 For PRINTER=0



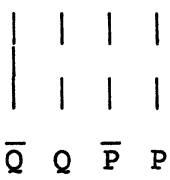
J26 For PRINTER=1



If the Interfacer 4 is used with an Interfacer 3, paddle 1 of switch 2 should be set OFF.

Jumpers:

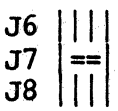
J1



J2-J5

APPENDIX C

J6-J8



J26

All remaining jumpers are unconnected.

JS1-JS3 Dip sockets near upper edge of card: Insert dip shunts in these three sockets.

JS4 Not connected.

JS5 Pins 5,9,10,11,12,13,14,15 and 16 connected together.

JS6 Pins 6,9,10,11,12,13,14,15 and 16 connected together.

APPENDIX C

CompuPro MDRIVE/H

	Board 1		Board 2	
	OFF	**ON**	**OFF**	**ON**
1	==		==	
2	==		==	
3		==		==
4		==		==
5		==		==
6	==		==	
7	==		==	
8		==		==
9		==		==
10		==	==	

You can have one or two MDRIVE/H boards, for a total size of either 512K or one megabyte, as a hard disk cache buffer.

ARCS100 Network Controller (DR Net networks only)

ARCS100 Switch Settings

	Switch 1		Switch 2		Switch 3		Switch 4	
	Off	**On**	**Off**	**On**	**Off**	**On**	**Off**	**On**
1		==	==			==	==	
2		==	==			==		==
3		==	==			==	==	
4		==		==		==	==	
5	==		==			==		==
6	==			==		==	==	
7	==		==			==	==	
8	==		==			==	==	

APPENDIX C

The ARCS100 switch configuration for Switch 5 will vary from system to system.

System 0 Switch 5		System 1 Switch 5		System 2 Switch 5		System 3 Switch 5	
Off	On	Off	On	Off	On	Off	On
1 ==		1	==	1 ==		1	==
2 ==		2 ==		2	==	2	==
3 ==		3 ==		3 ==		3 ==	
4 ==		4 ==		4 ==		4 ==	
5 ==		5 ==		5 ==		5 ==	
6 ==		6 ==		6 ==		6 ==	
7 ==		7 ==		7 ==		7 ==	
8 ==		8 ==		8 ==		8 ==	

These are binary numbers that identify each network node. Refer to **page C-18** to determine the proper switch settings for each node on the DR Net network.

ARCS100 Jumper Configurations

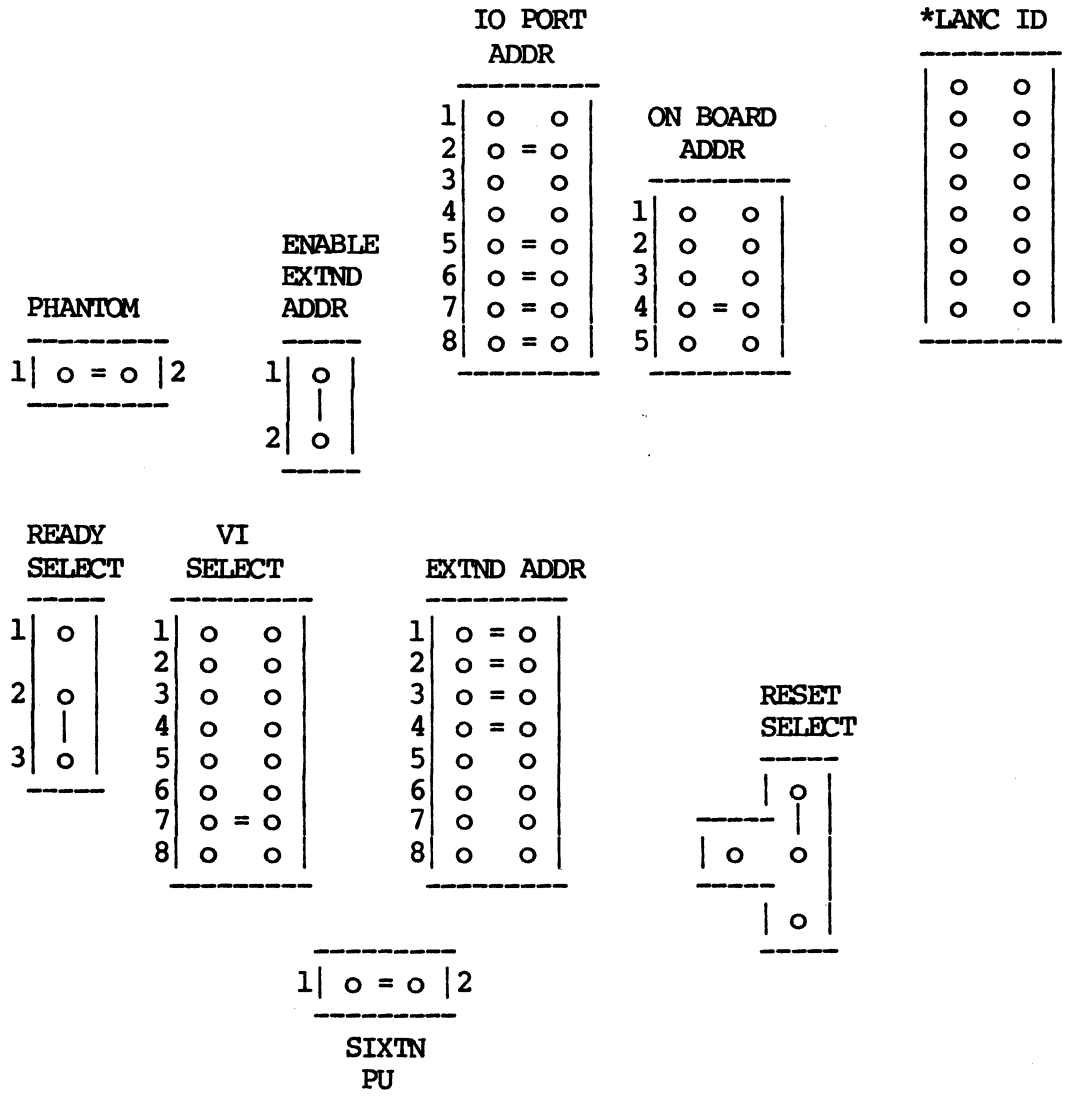
For the ARCS100 jumper configuration, the following slide jumpers are the only ones that should be installed:

JUMPER	FUNCTION
JP14	COM 9026 INTR = S100 Vectored Int line 6
JP22	S100 POC = ARCS100 POC
JP23	ARCS100 MREQ = S100 PHANTOM
JP26	ARCS100 WAIT = S100 XRDY

APPENDIX C

Intercontinental Microsystems LANS100 (DR Net networks only)

LANS100 Jumper Configurations



* The node ID on the LANC ID jumper will change for each board. Refer to the following page to determine the proper jumper configuration for each node.

APPENDIX C

The table below displays the switch settings for the ARCS100 network controller board and the jumper configurations for the LANS100 network controller board. This information has been provided for up to 64 nodes attached to the DR Net network.

ARCS100 Board

0 = switch should be on the left (OPEN)
1 = switch should be on the right (CLOSED)

Rightmost character corresponds to position 1 of switch 5.
Leftmost character corresponds to position 8 of switch 5.

LANS100 Board

0 = jumper should be not be installed
1 = jumper should be be installed

Rightmost character corresponds to bottom prong on LANC ID jumper.
Leftmost character corresponds to top prong on LANC ID jumper.

Node:	0	00000000	Node:	19	00010011
Node:	1	00000001	Node:	20	00010100
Node:	2	00000010	Node:	21	00010101
Node:	3	00000011	Node:	22	00010110
Node:	4	00000100	Node:	23	00010111
Node:	5	00000101	Node:	24	00011000
Node:	6	00000110	Node:	25	00011001
Node:	7	00000111	Node:	26	00011010
Node:	8	00001000	Node:	27	00011011
Node:	9	00001001	Node:	28	00011100
Node:	10	00001010	Node:	29	00011101
Node:	11	00001011	Node:	30	00011110
Node:	12	00001100	Node:	31	00011111
Node:	13	00001101	Node:	32	00100000
Node:	14	00001110	Node:	33	00100001
Node:	15	00001111	Node:	34	00100010
Node:	16	00010000	Node:	35	00100011
Node:	17	00010001	Node:	36	00100100
Node:	18	00010010	Node:	37	00100101

Node:	38	00100110	Node:	51	00110011
Node:	39	00100111	Node:	52	00110100
Node:	40	00101000	Node:	53	00110101
Node:	41	00101001	Node:	54	00110110
Node:	42	00101010	Node:	55	00110111
Node:	43	00101011	Node:	56	00111000
Node:	44	00101100	Node:	57	00111001
Node:	45	00101101	Node:	58	00111010
Node:	46	00101110	Node:	59	00111011
Node:	47	00101111	Node:	60	00111100
Node:	48	00110000	Node:	61	00111101
Node:	49	00110001	Node:	62	00111110
Node:	50	00110010	Node:	63	00111111

EXAMPLE:

Suppose that you have six nodes attached to the DR Net network. For node number 4, refer to the table on the preceding page, and you'll notice that the number to the right of node number 4 is 00000100. This means that if you have an ARCS100 or LANS100 network controller board, your switch settings or jumper settings should look like one of these configurations:

ARCS100
Switch 5

```
   Off   On

1  ==
2  ==
3        ==
4  ==
5  ==
6  ==
7  ==
8  ==

ON = 1
OFF= 0
High bit = subswitch 8
```

LANS100
LANC ID

```
o    o
o    o
o    o
o    o
o    o
o == o
o    o
o    o

Jumpered = 1
Unjumpered = 0
Highbit = top
```

APPENDIX D

D. Formatting Hard Disks

Formatting a disk erases all of the information on it, and it is not recommended that you format a disk during normal use. It is also not recommended that you run the destructive disk test, since this also destroys the information on your disk.

Formatting a disk is useful, however, when you repack the disk. See **Repacking the Disk** in **Section 3.8** for a description of packing the disk.

If you format or destructively test your hard disk, you must reinstall the operating system on the hard disk. See **Installing the Operating System on Your Hard Disk** on **page D-12.**

Do not put any of the hard disk formatting programs on your hard disk, since a user may accidentally run them and destroy the information on the hard disk.

Begin the formatting process by backing up all files that you want to save onto floppy diskettes. This should be done in single-user mode. Remember to use the WHO command before you use DOWN to put the system into MC-DOS single-user mode. Refer to **Chapter 3** for information on how to back up disks.

Once all the files on the hard disk have been copied onto floppy diskettes, press the RESET button and load the CP/M system diskette with the label "CP/M-86 Diagnostic Diskette." If, however, your system incorporates the Konan Controller, press the RESET button and load the MC-DOS diskette with the label "MC-DOS Diagnostic Diskette." (While either diskette can be used, Gifford Computer Systems recommends using the MC-DOS diskette on the Konan.) Do not run any hard disk format programs in multiuser mode.

According to your hard disk subsystem, go to the appropriate section in the following pages. No matter what type of hard disk you are formatting, the last two steps that the FMTHARD program performs are to run the INITDIR program to initialize the directory and the SET program to enable date and timestamping.

APPENDIX D

Formatting 8-Inch with the Disk 2 Controller

Format Morrow M10, M20, or M26 drives running on the CompuPro Disk 2 disk controller with the DISK2 program. Give the command:

A>DISK2 [drivetype] [arguments]

The "drivetype" is either M10 or M20, depending upon the capacity of your hard disk. The arguments are:

DRIVE nn	Specify a physical drive (0-3) to be formatted. The default is drive 0.
SKEW nn	Specify the skewing factor to be used. The suggested value is 2.
FORMAT	Format only the headers.
DATA	Set all the data bytes to E5 hex.
TEST	Perform a destructive test on the data fields.
SEEK	Perform a non-destructive seek test.
ALL	Perform the format, data test, and seek test.

To format the disk, use the ALL argument. The formatting takes two hours to complete. The program displays:

Formatting Track 0: D Hard 0 Soft 0

The track number is given before the colon, and the number of hard and soft sector errors (if any are found) is displayed to the right of each entry. If many errors are reported, consult your DISK 2 TECHNICAL MANUAL and contact your dealer.

After the disk is formatted, the system displays:

Verifying Track

Data Test Track

Seek Test

There are twelve passes in the last test. When it is finished, bad sectors are listed. Bad sectors that are found are mapped out and blocked from use.

5.25-Inch with the Konan Controller

Format Gifford Computer Systems 5.25-inch hard disk drives running on the Konan controller with the DGCFMT program.

Give the command:

```
0A>DGCFMT TYPE OPTIONS
```

The "type" is either F5-5, F5-11, F5-16, F5-21, F5-32 (Quantum), F5-37, F5-40, F5-41, or F5-56, where the second number matches the size of your hard disk in megabytes. That is, if you have a 21 megabyte hard disk, give the "type" as F5-21.

The options are:

BAD Maps out the bad sectors on the hard disk. Before you type this command, you must type:

```
0A>DGCFMT MAP
```

to find out which sectors are bad. Write these numbers down. Suppose that you have a 37 Megabyte hard disk drive, and the DGCFMT MAP command has indicated that cylinder #200 and head #3 are bad. To map the bad cylinder and head to a bad map list, type the following command:

```
0A>DGCFMT F5-37 BAD 200 3
```

The DGCFMT program will display a message indicating that it has mapped cylinder #200 and head #3 to a bad map list.

DRIVE n — Specify a physical drive, either 0 or 1. This is used only if you have two hard disks attached to your hard disk controller. The first drive, (0) is connected on the right of the board, and the second drive (1) is connected on the left.

d: — Give the letter of the logical drive, such as A:, B:, etc. If you have two hard disks connected to one controller, the letters for both drives are as if the disk was the first disk (that is, A: and B: instead of D: and E:).

MAP — Creates a list showing bad sectors on the hard disk drive.

SKEW n — Change the skewing factor from the default of 2. This option is rarely used.

TEST — Formats the drive. Formatting the drive will erase all information on the drive. The TEST option then tests the drive and clears out all bad sectors by remapping them. Although this takes longer than simply formatting the disk, it will give you better performance from your hard disk. Keep in mind that aborting the TEST procedure is dangerous as it destroys the bad map.

RETEST Test the disk for bad sectors without erasing them.

CLEAR Clear the disk without affecting the bad sectors.

FORMAT Format the disk and clear out bad sectors without testing it first. All information on the disk is lost.

If you specify a drive number but no drive letter, the entire hard disk is formatted. If you give a drive letter but no drive number, drive 0 is assumed. The drive number and drive letter must be specified before TEST or FORMAT.

For example, to test and format only drive B: of a 21 megabyte drive, give the command:

```
0A>DGCFMT F5-21 B: TEST
```

To test and format the second drive if you have two 11 megabyte drives, give the command:

```
0A>DGCFMT F5-11 DRIVE 1 TEST
```

APPENDIX D

5.25-Inch with the Disk 3 Controller

Format Gifford Computer Systems 5.25-inch hard disk drives running on the Disk 3 controller with the DISK3 program.

Give the command:

```
0A>DISK3 TYPE OPTIONS
```

The "type" is either F5-5, F5-11, F5-16, F5-21, F5-37, F5-40, or F5-56, where the second number matches the size of your hard disk in megabytes. That is, if you have a 21 megabyte hard disk, give the "type" as F5-21.

The options are:

DRIVE n — Specify a physical drive, either 0 or 1. This is used only if you have two hard disks attached to your hard disk controller. The first drive, (0) is connected on the right of the board, and the second drive (1) is connected on the left.

d: — Give the letter of the logical drive, such as A:, B:, etc. If you have two hard disks connected to one controller, the letters for both drives are as if the disk was the first disk (that is, A: and B: instead of D: and E:).

SKEW n — Change the skewing factor from the default of 2. This option is rarely used.

TEST — Test the disk for bad sectors, erase them and then format the disk. All information on the disk is lost. Although this takes longer than simply formatting the disk, it will give you better performance from your hard disk.

FORMAT Format the disk and clear out bad sectors without testing it first. All information on the disk is lost.

CLEAR Clear the disk without affecting the bad sectors.

RETEST Test the disk for bad sectors without erasing them.

If you specify a drive number but no drive letter, the entire hard disk is formatted. If you give a drive letter but no drive number, drive 0 is assumed. The drive number and drive letter must be specified before TEST or FORMAT.

For example, to test and format just drive B: of a 21 megabyte drive, give the command:

```
0A>DISK3 F5-21 B: TEST
```

To test and format the second drive if you have two 11 megabyte drives, give the command:

```
0A>DISK3 F5-11 DRIVE 1 TEST
```

APPENDIX D

8-Inch with the HDCA Controller

Boot the system with the CP/M-86 diagnostic diskette, and give the command:

```
A>FMTHD
```

The program responds:

```
Discus M10, M20 and M26 hard disk format program, Rev. 2.1.
Choose the desired function:

L = Format a Logical drive.
F = Format an entire physical drive.
C = Continue an interrupted test.
D = Run a Diagnostic test.
```

Enter (F) to format the drive:

```
Function (RETURN to exit):F
```

The program prompts:

```
Enter physical number to be tested or formatted (1-4, RETURN to
exit):
```

Enter the drive number and press RETURN. The drive number is 1 for the first drive, 2 for the second, etc.

The program prompts:

Select the drive type:

A = Discus M26, 26 megabyte drive.
B = Discus M10, 10 megabyte drive.
C = Discus M20, 20 megabyte drive.

Drive type (RETURN to exit):

Enter the letter corresponding to your disk drive type and press RETURN. If you have an M10, the system asks:

Select drive type:

F = Fujitsu
M = Memorex

Type (RETURN to exit):

Enter the 1 tter (F) and press RETURN. For all drive types, the program prompts:

Enter amount of formatting desired:

H = format Headers only (data remains intact)
D = erase Data fields also

Format the headers and data:

Function:D

The program prompts:

Formatting the entire physical drive will take about four minutes.....

All finished, returning to CP/M
Press RETURN to return to CP/M:[RETURN]

APPENDIX D

Testing the Drive

You only need to test the hard disk if you receive disk errors after the disk has been formatted. **Do not run this test without first contacting your dealer.**

This test destroys any information on your disk, and you must reformat the disk after the test is complete. The test takes six to fourteen hours to complete, depending on the size of your disk.

Test Morrow M10, M20, or M26 drives running on the HDCA disk controller with the FMTHD program. Follow the instructions in the **Shutting Down Your System (section 1.8-1)** of this manual. Boot the system with the CP/M-86 diagnostic diskette, and give the command:

A>FMTHD

If you have just turned ON the hard disk, allow time for it to stabilize (about two minutes) before beginning the test. The program prompts:

Discus M10, M20 and M26 hard disk format program, Rev. 2.1.
Choose the desired function:

L = Format a Logical drive.
F = Format an entire physical drive.
C = Continue an interrupted test.
D = Run a Diagnostic test.

Function (RETURN to exit):

Enter (D) to begin the diagnostic test:

Function (RETURN to exit):D

APPENDIX D

The program prompts:

How much of a diagnostic do you want to run:

1 = Sector header field test only.
2 = Sector data field test only.
4 = Seek mechanism test only.

Choose the diagnostic by adding together the desired options.

Options (RETURN to exit):

If the suggested option sum is seven (7), all diagnostics are run.

Options (RETURN to exit):7

The program prompts:

Enter physical drive number to be tested or formatted (1-4, RETURN to exit):

Enter the number and press RETURN. The number corresponds to the order in which the drive is connected to MC-DOS. The program prompts:

Select the drive type:

A = Discus M26, 26 megabyte drive.
B = Discus M10, 10 megabyte drive.
C = Discus M20, 20 megabyte drive.

Drive type (RETURN to exit):

Enter the letter corresponding to your hard disk type and press RETURN.

The first areas tested are the sector headers:

Testing sector headers.....

For the next six to fourteen hours, information about the tests is displayed on the terminal. As each test completes, new descriptions are displayed such as "Seek test" or "Testing sector data." Error information is also displayed if any are found. For example:

SECTOR ERR,TRACK 233, HEAD 0, SECTOR 2, COUNT 189

If this test reveals bad sectors or other errors, contact your dealer. Remember that you must format the disk after you run the destructive test.

APPENDIX D

Installing the Operating System on Your Hard Disk

If you have reformatted your hard disk, follow the instructions below to replace the MC-DOS operating system on your hard disk. You only need to do this if you have reformatted your hard disk, since Gifford Computer Systems includes all necessary files on your hard disk before shipping it to you.

Use the PIP utility to copy files from the floppy diskette back onto drive A:. The format of the command is as follows:

```
0A>PIP A:=d:*.*[VROW]
```

where d: is the drive designation for the floppy drive which will be the source drive.

Note that the small letter d: should be replaced with your appropriate drive designation. You can refer to the following chart for the drive designations appropriate for your system. If you have a single floppy drive system, assume that your floppy drive is the equivalent of the left-hand floppy drive in the table below.

Type of System	Logical Drive Designations		
	Hard disk Logical Drives	Left-hand Floppy Drive	Right-hand Floppy Drive
Floppy-only	none	A:	B:
5MB hard disk	A:	B:	C:
11Mb hard disk	A:B:	C:	D:
16Mb hard disk	A:B:	C:	D:
21Mb hard disk	A:B:C:	D:	E:
37Mb hard disk	A:B:C:D:E:	F:	G:
40Mb hard disk	A:B:C:D:E:	F:	G:
56Mb hard disk	A:B:C:D:E:	F:	G:
84Mb hard disk	A:B:C:D:E:F:	G:	H:

Systems with two or more hard disks add the same number of logical drives as shown in the table above for each hard disk, with one or two floppy drives moved up in the alphabet (towards "P").

APPENDIX E

E. Programming Serial Ports in MC-DOS

Advanced programmers may want to interface non-standard RS-232 devices with MC-DOS. You can control the console port from either the 8085 or the 8088 using BDOS call number fifty (50). A few precautions must be taken before connecting the device to the system.

In the following calls, be sure to use a valid terminal number, since invalid terminal numbers may stop the system. Terminal numbers are the same as those marked on the back panel (0, 1, etc.).

For every physical terminal on your system, there is an MC-DOS "Physical Input Process" (PIN) which is responsible for switching virtual terminal screens and for control characters ^S, ^P, ^C, ^Q, and ^O. Since you cannot have two processes inputting from the same port, you must stop PIN from inputting before you can begin to program a serial port.

Each physical terminal has its own queue called MXconsn, where **n** is the physical terminal number. MC-DOS mutual exclusion queues enable you to open a queue with Q_OPEN, and read a queue with Q_READ so that your process can have sole control of the queue which is assigned to the physical terminal number.

This prevents other processes from occupying that queue. You must set a flag to alert the PIN process to call the same queue.

This alerts the PIN process to await the queue which you are currently holding. This prevents the PIN process from inputting to the serial port which you are trying to program until you release the queue with Q_WRITE. You are now able to program the serial port.

Set up a 5-byte area of memory for the BIOS descriptor.

8-bit programs load the C register with 32H, and DE with the address of the BIOS descriptor.

APPENDIX E

16-bit programs load the CL register with 32H, DX with the offset of the BIOS descriptor, and DS with the segment of the BIOS descriptor.

The structure of the BIOS descriptor is:

```
FUNC  arg1  TYPE  arg2  arg3

  6
```

The first byte, FUNC, is always 6. The third byte, TYPE, is either 1, 2, 3, 4, 7, or 8: these represent setup, output, input, input status, extended status, and extended setup, respectively. The three arguments depend on the value of TYPE.

```
Setup TYPE = 1
arg1 = physical terminal number (console number)
arg2 = bits 0 - 3: select baud rate
          0 = 9600
          1 = 110
          2 = 300
          3 = 600
          4 = 1200
          5 = 1800
          6 = 2400
          7 = 4800
          8 = 9600
          9 = 19200
     bits 4 - 5:  select parity
          0 = no parity
          1 = odd parity
          2 = even parity
      bit 6:  select number of stop bits
          0 = 1 stop bit
          1 = 2 stop bits
      bit 7:  select number of data bits
          0 = 7 data bits
          1 = 8 data bits
 arg = handshaking
          0 = no handshaking
          1 = DTR handshaking
          2 = XON/XOFF
```

APPENDIX E

Output TYPE = 2
arg1 = character to output
arg2 = console number
No value returned.

Input TYPE = 3
arg1 = physical terminal number (console number)
Returns byte in A register (8085) or AL register (8088).

Status TYPE = 4
arg1 = physical terminal number (console number)
Returns input status, (0 = not ready, FF = ready) in A register, (8085) or AL register (8088).

Extended status TYPE = 7
arg1 = physical terminal number (console number)
Returns extended hardware information in A register (8085) or AL register (8088). The hardware status is indicated by the following bits:

Bit #	0	1
0	DSR high	DSR low
1	DCD high	DCD low
4	ok	parity error
5	ok	overrun error
6	ok	framing error

Normally, DSR is line twenty (20) and DCD is line eight (8) of the RS-232 connector.

Extended setup TYPE = 8
arg1 = physical terminal number (console number)
arg2 = set parameters

Bit #	0	1
0	set DTR high	set DTR low
1	set RTS high	set RTS low
2	no break	send break
7	no reset	reset all errors

Normally, DTR is line six (6) and RTS is line five (5) of the RS-232 connector.

APPENDIX E

The following example is a C program that uses the calls described on the previous page:

```
/*
 * Logical port we will be talking through
 */

#define         PORT            5

/*
 * MC-DOS operating system function calls used in this program
 */

#define         S_BIOS           50
#define         DEV_FLAGSET     133
#define         Q_OPEN          135
#define         Q_READ          137
#define         P_DELAY         141

/*
 * Function 6 extended XIOS call commands
 */

#define         STTY            1
#define         OUTPUT          2
#define         INPUT           3
#define         INPUT_STATUS    4

/*
 * MC-DOS Queue Parameter Block defintion
 */

struct qpb
       {
       int      qpb_fill0;
       int      qpb_queueid;
       int      qpb_filll;
       char *   qpb_buffer;
       char     qpb_name[8];
       } qpb;
```

APPENDIX E

```
/*
 * MC-DOS BIOS Descriptor
 */

struct xios
       {
       char    xios_func;
       char    xios_arg1;
       char    xios_type;
       char    xios_arg2;
       char    xios_arg3;
       } xios;

main ()
       {
       char    port_name[8];
       char    c;
       int     count;

       qpb.qpb_buffer  = 0;
       qpb.qpb_queueid = 0;
       sprintf (port_name, "%d ", PORT);
       strmove (&qpb.qpb_name[0], "MXcons  ", 8);
       strmove (&qpb.qpb_name[6], port_name, 2);

       bdos (Q_OPEN, &qpb);
       bdos (Q_READ, &qpb);
       bdos (DEV_FLAGSET, 0x80 + PORT);
       bdos (P_DELAY, 1);
       pinit ();

       for (count = 0; count < 10; count++)
              {
              c = pin ();
              pout (c);
              }
       }

/*
 * Return character from selected port
 */
```

```
pin ()
        {
        xios.xios_func = 6;
        xios.xios_arg1 = PORT;
        xios.xios_type = INPUT;
        return bdos (S_BIOS, &xios);
        }

/*
 * Output a character to selected port
 */

pout (c)
        char    c;
        {
        xios.xios_func = 6;
        xios.xios_arg2 = PORT;
        xios.xios_type = OUTPUT;
        xios.xios_arg1 = c;
        bdos (S_BIOS, &xios);
        }

/*
 * Set up the selected port
 */

pinit ()
        {
        xios.xios_func = 6;
        xios.xios_type = STTY;
        xios.xios_arg1 = PORT;
        xios.xios_arg2 = 0xc0;
/* 8 databits, 2 stopbits, no parity, 9600 baud */
        xios.xios_arg3 = 0;     /* No handshaking */
        bdos (S_BIOS, &xios);
        }
```

APPENDIX F

F. Using the Printer Port on TeleVideo Terminals

The printer port on the TeleVideo 925 or the TeleVideo 950 has limited use. All information which goes to the printer port must pass through the terminal and must be displayed on the video screen. Since the terminal is not a printer, it may not recognize printer control characters or special characters imbedded in ASCII files. The printer port can be used with the CTRL P (^P) printer toggle, and with commands such as TYPE, which are intended to send information to a video screen.

If you are going to attach a printer to your TeleVideo terminal, be sure that the cable between the terminal and computer, or between the terminal and modem, has pin-20 of each end connected, so that the computer and terminal can use DTR handshaking. The terminal, computer, and printer must operate at the same baud rate, and the terminal must be attached to an Interfacer board, not the System Support board.

The cable between the terminal and printer must also be able to handle DTR handshaking. Be sure that pin-20 on the terminal end of the cable is connected to the DTR pin on the printer end. For example, on Diablo printers, this is pin-20; on Okidata 82A and 93A printers, this is pin-11.

This terminal-to-printer cable can be tested by using it to connect the printer directly to the computer's printer port. Send a few pages of text to the printer, and check the output for lost characters.

If the terminal is a TeleVideo 925, set the internal switch S3-10 down. This connects the DTR line (pin-20) to the computer.

The communications protocol between computers and terminals is different from that between computers and modems. When communicating with terminals, XON/XOFF must be disabled; when communicating with modems, XON/XOFF must be enabled.

APPENDIX F

You must enable or disable XON/XOFF in the TTYS file on the computer. Send a character from the computer to the terminal (or type the character in local mode on the terminal) to enable or disable it in the terminal. Enable XON/XOFF in the terminal by sending a CTRL O (^O) (ASCII value 15 decimal, 0F hex); disable XON/XOFF with a CTRL N (^N) (ASCII value 14 decimal, 0E hex).

While the terminal is running, turn the printer ON by sending the terminal an ESCAPE@ (ASCII values 27 and 64 decimal, 1B and 40 hex). All data received by the terminal are transmitted to the printer.

Turn OFF the printer by sending the terminal an ESCAPE A (ASCII values 27 and 65 decimal, 1B and 41 hex).

APPENDIX G

G. Port Numbers Used in MC-DOS

If you add IEEE 696/S-100 boards to your MC-DOS system, you must be sure that they do not conflict with the I/O ports already assigned. The following chart lists all of the ports used by the system. You may use devices that do not conflict with any of the ports listed. You can also use devices that conflict with the ports listed with an asterisk (*), as long as you do not have the device listed for those ports. For example, you can add a device that uses port 64. You can also add a device that uses port E5, as long as you do not have a PMMI modem attached to the system.

Port # in Hex	Device
00-01	Interfacer 1 or 2 # 1
02-03	Interfacer 1 or 2 # 2
04-05	Interfacer 1 or 2 # 3
06-07	Interfacer 1 or 2 # 4
08-09	Interfacer 1 or 2 # 5
0A-0B	Interfacer 1 or 2 # 6
0C-0D	Interfacer 1 or 2 # 7
0E-0F	Interfacer 1 or 2 # 8
10-17	Interfacer 3 or 4 # 1
18-1F	Interfacer 3 or 4 # 2
20-2F	Reserved by Gifford Computer Systems
*30-3F *	Alloy Engineering tape controller
40-41	Futurex data encryption board
42-4F	Reserved by Gifford Computer Systems
50-5F	System Support 1
*80-83 *	Morrow HDCA disk controller
*90 *	Disk 3 hard disk controller
*B0-B1 *	DR Net network controller
C0-C3	Disk 1 or Disk 1A floppy disk controller
*C6-C7 *	MDRIVE/H
*C8-C9 *	Disk 2 hard disk controller
*D0-D3 *	SemiDisk
*D7-D7 *	Konan hard disk controller
*E0-E3 *	DC Hayes modem
*E4-E7 *	PMMI modem
*F0-F0 *	Selector channel (part of Disk 2)
*FD-FE *	CPU 8085/88 or CPU 8086/87
*FD-FF *	Macrotech MI-286 (80287 uses F8, FA, FC)
*FE-FF *	CPU-Z

APPENDIX H

H. Notes on Installing WordStar and dBASE II

The following will help you run the WordStar text processing program more efficiently on your MC-DOS system. If you purchased WordStar from Gifford Computer Systems and your printers and terminals were known, it is already installed on your computer.

WordStar Installation

If you reinstall WordStar, you do not need to reset any of the flags listed on pages 8-5 or 8-6 of the WORDSTAR INSTALLATION MANUAL. Since MC-DOS treats 8-bit programs differently from MP/M II, most of the timing considerations that are discussed on these pages are not valid for your system.

Delays

If you reinstall your copy of WordStar, the following labels should be set with the INSTALL program:

Label	New setting
DELCUS:	1
DELMIS:	1
DEL1:	0
DEL2:	0
DEL3:	0
DEL4:	0
DEL5:	0

These labels refer to values used in timing loops for certain delays in WordStar. They take valuable CPU time and slow down WordStar and your system.

These values work for TeleVideo terminals. If you have other terminals, and the above settings prevent you from running WordStar, set the values higher.

APPENDIX H

Printer Release

There are other locations in WordStar that should be patched with the DDT86 program. There is a bug in WordStar that does not release the printer when you finish printing a file. The following are terminal sessions with DDT86 to free the printer in 8- and 16-bit WordStar, Version 3.3 and 8-bit WordStar, Version 3.0. The boldfaced characters are values that you should enter.

16-bit WordStar, Version 3.3:

```
OA>DDT86
DDT86 1.1
-RWS.CMD
  START       END
4400:0000  4400:54FF
-s324
4400:0324 90 e9
4400:0325 90 39
4400:0326 C3 00
4400:0327 90 .
-s846
4400:0846 90 e9
4400:0847 90 2b
4400:0848 C3 fb
4400:0849 00 .
-s33d
4400:033D 00 e9
4400:033E 90 39
4400:033F C3 00
4400:0340 00 .
-s33a
4400:033A 00 e9
4400:033B 90 41
4400:033C C3 00
4400:033D E9 .
-s360
4400:0360 00 b1
4400:0361 00 0c
4400:0362 00 cd
4400:0363 00 e0
```

```
4400:0364 00 80
4400:0365 00 ff
4400:0366 00 11
4400:0367 00 75
4400:0368 00 0a
4400:0369 00 c6
4400:036A 00 06
4400:036B 00 ba
4400:036C 00 02
4400:036D 00 e9
4400:036E 00 c6
4400:036F 00 06
4400:0370 00 bd
4400:0371 00 02
4400:0372 00 e9
4400:0373 00 c3
4400:0374 00 b1
4400:0375 00 9f
4400:0376 00 cd
4400:0377 00 e0
4400:0378 00 c3
4400:0379 00 b1
4400:037A 00 03
4400:037B 00 cd
4400:037C 00 e0
4400:037D 00 c3
4400:037E 00 b1
4400:037F 00 06
4400:0380 00 b2
4400:0381 00 fe
4400:0382 00 cd
4400:0383 00 e0
4400:0384 00 c3
4400:0385 00 .
-WWSTEST.CMD
-^C
0A>
```

APPENDIX H

This Control C (^C) will return you to the system prompt, **0A>**, or whatever your current drive really is. Test the patched version, to make sure that you made no keystroke errors, before replacing your old copies with this enhanced version of WordStar. Then type the commands:

```
REN WS.CMD=WSTEST.CMD
SET WS.CMD[sys,ro]
```

8-bit WordStar, Version 3.3

```
0A>DDT86
DDT86 1.1
-RWS.COM
START           END
2800:0000       2800:3DFF
-S614
2800:0614 00 C3
2800:0615 00 CB
2800:0616 C9 02
2800:0617 00
-S1CB
2800:01CB 00 0E
2800:01CC 00 9F
2800:01CD 00 C3
2800:01CE 00 05
2800:01CF 00 00
2800:01D0 00
-WWSTEST.COM
-^C
0A>
```

This Control C (^C) will return you to the system prompt **0A>**, or whatever your current drive really is. Test the patched version, to make sure that you made no keystroke errors, before replacing your old copies with this enhanced version of WordStar. Then type the commands:

```
REN WS.COM=WSTEST.COM
SET WS.COM[sys,ro]
```

APPENDIX H

8-bit WordStar, Version 3.0

```
0A> DDT86
-RWS.COM
  START    END
2800:0000  2800:3DFF
-S610
2800:0610 00 C3
2800:0611 00 E0
2800:0612 C9 02
2800:0613 00 .
-S1E0
2800:01E0 00 0E
2800:01E1 00 9F
2800:01E2 00 C3
2800:01E3 00 05
2800:01E4 00 00
2800:01E5 00 .
-WWS.COM
-^C
0A>
```

Renaming WordStar

When you reinstall WordStar, the INSTALL program asks what you want to name the new cop . The INSTALL program will write that name in the program, and you cannot rename it later and still have the RUN subcommand of WordStar work.

Troubleshooting a Reinstallation

Before you reinstall WordStar, write down the printer configuration, communications protocol, and output device, which are displayed when you start running WordStar. Be sure to keep these settings when you go through the reinstallation, so that your printer works correctly. Almost every printer will work with the settings "Teletype-like printer," "No Communications Protocol," and "CP/M LST: device."

APPENDIX H

Installation of dBASE II, Version 2.4

During the installation process, when the screen displays:

Change macro, date, etc (Y/N)?

Answer yes by typing (Y), and the screen will display:

Enter a character to be used for indicating macros or a RETURN for a default character of ampersand (&)

Specify an ampersand (&) by pressing RETURN. The screen will display:

Type a RETURN if the error correction dialogue is to be used, or any other key if no dialogue is wanted:

Press RETURN and the screen will display:

A - CP/M 2.2
B - MP/M II system

Specify that you are using an MP/M II by typing (B).

Only this part of the procedure is specific to MC-DOS.

APPENDIX I

I. Directory of System Files Supplied by Gifford Computer Systems

The only file which is essential in loading the operating system is MC.DOS. All other files are optional. This does not mean that all the remaining files can be deleted. You should keep as many of the files in user area 0, drive A: as you can. You must have enough system programs, (such as PIP.CMD and ERA.CMD) to operate the system effectively. If you run any 8-bit programs, SW.CMD must be on user area 0, drive A:.

There is no reason to delete any of the files unless you are running a floppy-only system and space on drive A: is critical.

Your MC-DOS system is distributed on up to four physical diskettes: the MULTIUSER CONCURRENT DOS SYSTEM DISK, the MULTIUSER CONCURRENT DOS XIOS DISK, the DR NET NETWORKING DISK, and the MULTIUSER CONCURRENT DOS UTILITIES diskette. Systems that are not linked to DR Net networks do not include the DR Net diskette. A listing for all of the files contained on these diskettes is provided on the following pages.

Files with the extension .CON and .RSP are required by GENCCPM. Files with the extension .CMD are command files, which may include system commands and 8-bit dBASE II commands related to system functions; those with .SAM are sample files; and those with .DBF are dBASE II database files. Application programs which are widely used on your system should be located on user area 0, drive A:. This makes them accessible to all user areas. Use the SET command to set commonly used applications programs to system and read-only status.

Data files (files with no extension, and files that have the extensions .HLP, .KEY, .NDX, and .DBF) can have the DIR attributes. HELP files (.HLP) must be set with the RO and DIR attributes to be accessible from all directory areas.

APPENDIX I

MULTIUSER CONCURRENT DOS SYSTEM and UTILITIES DISKETTES

ABORT.CMD
AP.CMD
ASM86.CMD
ATTACH.CMD
BUFFER.CMD
BUF0.CMD
BUF1.CMD
BUF2.CMD
BUF3.CMD
CCPMINIT.CMD
CHRON.CMD
CHSET.CMD
CLOCK.CMD
CONSOLE.CMD
CPY.CMD
DATE.CMD
DAYFILE.CMD
DDT86.CMD
DIR.CMD
DOWN.CMD
DSKRESET.CMD
DSP.CMD
ED.CMD
ERA.CMD
ERAQ.CMD
FILES.CMD
FORMAT.CMD
GENCMD.CMD
HELP.CMD
HELPNDX.CMD
INITDIR.CMD
KEYS.CMD
LOGIN.SAM
LOGOUT.SAM
LPRS
MAIL.CMD
MAIL.HLP
MAIL.NDX
MAKE.CMD
MCDOS.HLP
MCDOS.NDX
NEWUSER.CMD
PASSWD
PASSWD.HLP
PASSWD.NDX
PASSWORD.CMD
PIP.CMD
PRINT.CMD
PRINTER.CMD
REN.CMD
SDIR.CMD
SET.CMD
SETMEM.CMD
SHOW.CMD
SPACE.CMD
SPL.CMD
SPOOL.CMD
SUBMIT.CMD
SW.CMD
SW86.CMD
SYNC.CMD
SYSDISK.CMD
SYSGEN.CMD
SYSINIT.SAM
SYSTAT.CMD
TIME.CMD
TIMELOG.CMD
TIMELOG.DBF
TIMELOG.SUB
TIMELOG1.CMD
TIMELOG2.CMD
TIMELOG3.CMD
TIMELOG4.CMD
TIMELOG5.CMD
TIMELOGX.CMD
TIMELOGX.DBF
TIMELOGY.CMD
TIMELOGZ.CMD
TTYS
TYPE.CMD
UPTIME.CMD
USER.CMD
USERINIT.SAM
VCMODE.CMD
VT950.KEY
VTGCS80.KEY
VTTYS.SAM
WHO.CMD
WHYBOOT.CMD
WRITE.CMD
WIMP.DBF
ZWIMP.CMD

APPENDIX I

DR NET NETWORKING DISK

ADDNET.CMD	NAMSVR.DAT
CONFIG.NET	NET.CMD
DRNET.CMD	NETMOTD
NAMENODE.CMD	SETNET.CMD

This diskette is supplied only for systems that are part of DR Net local area networks.

MULTIUSER CONCURRENT DOS XIOS DISK

BDOS.CON	FIXSTEP.CMD	PIN.RSP
BOOT.LDR	FLOPPY	PIP.CMD
BUFFER	FMTHD.CMD	REN.CMD
CCPMINIT.CMD	FORMAT.CMD	RTM.CON
CIO.CON	GCS.RSP	SHELL.RSP
CPY.CMD	GENCCPM.CMD	SMCFMT.CMD
DGCFMT.CMD	HRD32K-1	STATUS.RSP
DIR.CMD	HRD64K-1	SUP.CON
DISK1.LDR	KRON.RSP	SYSDAT.CON
DISK1A.LDR	MC.DOS	SYSGEN.CMD
DISK2.CMD	MEM.CON	TYPE.CMD
DISK3.CMD	MEMORY.RSP	VOUT.RSP
ERA.CMD	PASSWD	

X1030702	X1071502	X1101503
X1031502	X1090704	X1110706
X1050704	X1090708	X1111506
X1051504	X1091504	X1130704
X1060703	X1091508	X1131504
X1061503	X1100703	XFLOPPY
X1070702		

The file extensions of the XIOS files will indicate the hard disk configuration. The XIOS diskette is used to customize or reconfigure MC-DOS.

APPENDIX J

J. dBASE II Enhancements

Gifford Computer Systems has developed several dBASE II command files which will be useful to dBASE II programmers. These command files include:

CALL.CMD Executes MC-DOS function calls from a dBASE II program.

SYSCMD.CMD Executes an MC-DOS command from a dBASE II program.

GETDATE.CMD Stores the MC-DOS system date to the dBASE II date function. Also prints date and day of the week.

GETTIME.CMD Displays the time of day.

FREEPRNT.CMD Detaches the list device, enabling other users to print to that device.

QUITTO.CMD Allows a dBASE II user to use the QUIT TO function.

WHOM.CMD Displays the dBASE II user's terminal number.

The most powerful of these commands are the CALL.CMD and the SYSCMD.CMD.

In addition to the dBASE II command files developed for 8-bit processors, listed above, Gifford Computer Systems has developed several dBASE II command files for 16-bit processors.

CALL.PRG Executes MC-DOS function calls from dBASE II programs.

FREEPRNT.PRG Detaches the list device, enabling other users to print to that device.

GETDATE.PRG Stores the MC-DOS system data to the dBASE II date function. Also prints date and day of the week, as well as the time of the day.

SYSCMD.PRG Executes an MC-DOS command from a dBASE program.

APPENDIX J

WHOM and **QUITTO** were not converted for the 16-bit processor.

The documentation and listings for the commands listed above follows.

Purchasers of Gifford systems are hereby given permission to reproduce or modify the listings provided here.

APPENDIX J

CALL.CMD

Concurrent DOS function calls allow a dBASE II routine to change printers, read the system clock/calendar, access a file that is located in another user area, or carry out a number of functions which are valuable in a multiuser environment.

The CALL.CMD file enables the dBASE II user to make system function calls. This is done by storing the number (in decimal) of the call to be made to a numeric variable called FUNCTION. A numeric value must also be stored to a variable called ARG. This is the argument of the function call and may be zero for those functions which require no argument. The DO CALL command will execute the routine, and the result can be found beginning in location 48 decimal. The value of the first byte of the result is stored in a numeric variable called RESULT.

Here is the dBASE II code for any function call:

```
STORE ' ' TO RUN
POKE 29,FUNCTION
STORE INT(ARG/256) TO ARG2
STORE ARG-(256*ARG2) TO ARG1
POKE 30,ARG1
POKE 31,ARG2
POKE 32,58,29,0,79,42,30,0,235,205,5,0,34,48,0,201
SET CALL TO 32
CALL RUN
STORE PEEK(48) TO RESULT
RELEASE RUN,ARG1,ARG2
RETURN
```

APPENDIX J

CALL.CMD is equivalent to the following assembly code:

```
29 DB   FUNCT
30 DW   ARG
32 LDA  29
35 MOV  C,A
36 LHLD 30
39 XCHG
40 CALL 5
43 STA  48
46 RET
47
48 DB   RESULT
```

The following function calls can be executed from dBASE II. Programmers should refer to Digital Research Inc.'s CONCURRENT CP/M OPERATING SYSTEM PROGRAMMER'S REFERENCE GUIDE for additional information.

0	System reset
11	Get console status
14	Select disk
25	Return current disk
24	Return login vector
28	Write protect disk
29	Get R/O vector
32	Set/get user code
37	Reset drive
141	Delay
143	Terminate process
145	Set priority
146	Attach console
147	Detach console
148	Set console
153	Get console number
155	Get a date and time
158	Attach list
159	Detach list
160	Set list
161	Conditional attach list
162	Conditional attach console
164	Get list number

APPENDIX J

The set/get user code cannot be used to access restricted user areas that are not specified in your login PASSWD file.

SYSCMD.CMD

The SYSCMD.CMD file is a dBASE II command file which allows execution of an MC-DOS command from within a dBASE II program. Store the command that you want to execute into a character variable called CMD. Then give the command DO SYSCMD. If the command that you intend to execute is an 8-bit program, include the SW command in the variable string.

For example, suppose that you want to execute DSKRESET from within dBASE II:

```
STORE 'DSKRESET' TO CMD
DO SYSCMD
```

Or if you want to PIP a file to another user area and drive:

```
STORE 'PIP B:filename.typ[G3]=A:filename.typ[G1]' TO CMD
DO SYSCMD
```

If you want to run an 8-bit program, then include the SW command as follows:

```
STORE 'SW PROG80' TO CMD
DO SYSCMD
```

Note: With dBASE II releases earlier than 2.4, avoid commands which return messages to the screen such as DIR. These messages tend to hang up the computer. Version 2.4 or later can

execute commands which send messages to the terminal and can even execute programs such as WordStar.

The dBASE II listing for SYSCMD.CMD is as follows:

```
POKE 49408,229,14,145,17,196,0,205,5,0,225,126,35,50,50
POKE 49422,193,79,17,51,193,126,35,18,19,13,194,19,193
POKE 49435,175,18,14,150,17,50,193,205,5,0
POKE 49445,14,146,205,5,0,14,145,17,200,0,195,5,0,0,0
SET CALL TO 49408
CALL CMD
```

The PIP command cannot be used to copy files to or from restricted user areas.

GETDATE.CMD

This command file will do a system function call to find the date. It will store the date to the dBASE II system date and print the date. It does a calculation to find the day of the week and displays the day of the week and the date.

NOTE: In order to store the system date to dBASE II, the dBASE II macro must be defined as an ampersand (&).

Here is the dBASE II listing for GETDATE.CMD:

```
STORE STR(PEEK(16)-6*INT(PEEK(16)/16),2) TO mm
STORE STR(PEEK(17)-6*INT(PEEK(17)/16),2) TO dd
STORE STR(PEEK(18)-6*INT(PEEK(18)/16),2) TO yy
SET DATE TO &mm &dd &yy
STORE VAL(mm) TO month
STORE VAL(dd) TO day
STORE VAL(yy) TO year
STORE 708926.000 TO BaseDay
STORE year + 1900 TO year
STORE INT(30.57*month)+INT(365.25*year-395.25)+day TO X
IF month>2
  IF INT(year/4)=year/4
   STORE X-1 TO X
```

```
  ELSE
   STORE X-2 TO X
  ENDIF
ENDIF
STORE year - 1900 TO year
STORE (X - BaseDay)/7.000 TO interval
STORE interval - INT(interval) TO fraction
STORE INT((fraction* 7.000) + 0.5) TO DayNum
DO CASE
CASE DayNum = 0
     STORE 'Sunday' TO weekday
CASE DayNum = 1
     STORE 'Monday' TO weekday
CASE DayNum = 2
     STORE 'Tuesday' TO weekday
CASE DayNum = 3
     STORE 'Wednesday' TO weekday
CASE DayNum = 4
     STORE 'Thursday' TO weekday
CASE DayNum = 5
     STORE 'Friday' TO weekday
CASE DayNum = 6
     STORE 'Saturday' TO weekday
ENDCASE
ERASE
? 'Today is ' + weekday + ', ' + DATE()
RELEASE interval, fraction, BaseDay, weekday, DayNum, X
RETURN
```

APPENDIX J

GETTIME.CMD

This short command file pokes a routine to execute the function call which returns the time of day. That time is then peeked and displayed for the user.

Here is the dBASE II listing for GETTIME.CMD:

```
STORE ' ' TO RUN
POKE 32,14,155,17,48,0,195,5,0
SET CALL TO 32
CALL RUN
STORE PEEK(50)-6*INT(PEEK(50)/16) TO HOUR
STORE PEEK(51)-6*INT(PEEK(51)/16) TO MIN
STORE PEEK(52)-6*INT(PEEK(52)/16) TO SEC
? STR(HOUR,2)+':'+STR(MIN,2)+':'+STR(SEC,2)
RELEASE HOUR,MIN,SEC,RUN
RETURN
```

FREEPRNT.CMD

This short command file will poke a routine which will detach the list device, enabling other users to print to that device.

Here is the dBASE II listing for FREEPRNT.CMD:

```
STORE ' ' TO RUN
POKE 32,14,159,205,5,0,201
SET CALL TO 32
CALL RUN
RELEASE RUN
RETURN
```

APPENDIX J

QUITTO.CMD

This command file effectively allows a user to utilize the 'QUIT TO fn' command in dBASE II. The calling convention is as follows:

```
STORE 'udSW command' TO cmd
DO quitto
```

In the preceding example, u is the user area number where you would like to execute the command, and d is the drive name on which you would like the command to execute. u and d are upper case, single alphabetic characters with A=user 0, B=user 1, etc.

For example:

```
STORE 'JC SUBMIT GIFFORD' TO CMD
DO QUITTO
```

Or:

```
STORE 'JCSUBMIT GIFFORD' TO CMD for 16-bit commands
```

This will quit dBASE II and change the user area to user 11 (J), drive C: and then submit the file GIFFORD.SUB. You can return to dBASE II by invoking dBASE as the last line of the submit.

```
.STORE 'JC SUBMIT GIFFORD' TO CMD
 1B> USER 11
 11B> C:
 11C> SUBMIT GIFFORD
    .
    .
    .

 11C> USER 1
  1C> B:
  1B>
```

APPENDIX J

Here is the dBASE II listing for QUITTO.CMD:

```
POKE 49408,70,4,17,90,193,126,18,35,19,5,194,5,193,62
POKE 49422,195,33,25,193,50,0,0,34,1,0,201,17,128,0,14
POKE 49437,26,205,5,0,58,91,193,14,32,205,81,193,58,92,193
POKE 49452,14,14,205,81,193,58,90,193,33,93,193,214,2,71
POKE 49466,17,128,0,126,18,35,19,5,194,61,193,175,18,14
POKE 49480,47,205,5,0,14,0,205,5,0,254,32,200,214,65,95
POKE 49495,195,5,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
POKE 49517,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0

SET CALL TO 49408
CALL CMD
QUIT
```

The USER command cannot be used to access restricted user areas.

WHOM.CMD

This short command file will poke a routine which returns the user's terminal number. That number is then peeked and displayed for the user.

Here is the dBASE II listing for WHOM.CMD:

```
SET TALK OFF
STORE ' ' TO RUN
POKE 32,14,153,205,5,0,50,48,0,201
SET CALL TO 32
CALL RUN
STORE PEEK(48) TO CONSOLE
? 'CONSOLE = '+STR(CONSOLE,2)
SET TALK ON
RELEASE CONSOLE,RUN
RETURN
```

APPENDIX J

CALL.PRG

This routine allows 16-bit dBASE users to make MC-DOS function calls. It can be used to find the time or your terminal number, to find or change your printer device, to find or change your user area, to detach your list device so others can print after you finish, to set your priority, and much more.

The MC-DOS function calls are implemented by storing the number (in decimal) of the call to be made to a numeric variable called FUNCTION. A numeric value must also be stored to a variable called ARG. This is the argument of the function call and should be zero for those functions which require no argument. The DO CALL command will execute the routine, and the result can be found beginning in location 48 decimal. The value of the first byte of the result is stored in a numeric variable called RESULT.

The following is the source code for CALL.PRG:

```
STORE ' ' TO RUN
POKE 65297,FUNCTION
STORE INT(ARG/256) TO ARG2
STORE ARG-(256*ARG2) TO ARG1
POKE 65298,ARG1,ARG2
POKE 65280,46,138,14,17,255,46,139,22,18
POKE 65289,255,205,224,46,163,20,255,195
SET CALL TO 65280
CALL RUN
STORE PEEK(65300) TO RESULT
RELEASE RUN,ARG1,ARG2
RETURN
```

This file must be in your current directory area, or on user area 0 of your current logical drive (SET to SYS and RO status) for it to be accessible when you run dBASE.

Refer to CONCURRENT CP/M PROGRAMMER'S REFERENCE GUIDE for further information.

APPENDIX J

FREEPRNT.PRG

This command is the 16-bit analog to the 8-bit FREEPRNT.CMD routine which releases the printer while in the dBASE program. FREEPRNT.PRG will "poke" a routine to detach the list device (function 159), thus enabling other users to print to that device when your dBASE II program finishes with the printer.

The code is as follows:

```
SET TALK OFF
STORE ' ' TO RUN
POKE 65280,46,138,14,17,255,46
POKE 65286,139,22,18,255
POKE 65290,205,224,46,163,20,255
POKE 65296,195,159,0,0
SET CALL TO 65280
CALL RUN
SET TALK ON
RELEASE RUN
RETURN
```

dBASE programs that continue after printing should include the command:

```
CALL FREEPRNT
```

in order to free the printer.

GETDATE.PRG

This command is the 16-bit version of GETDATE.CMD with one major modification: GETDATE.PRG returns the system date and time with a single MC-DOS function call (function 155).

APPENDIX J

The GETDATE.PRG code is listed below:

```
SET TALK OFF
STORE ' ' TO RUN
POKE 65280,46,138,14,17,255
POKE 65285,46,139,22,18,255
POKE 65290,205,224,46,163,20
POKE 65295,255,195,155,22,255
SET CALL TO 65280
CALL RUN
```

The following section computes the time:

```
STORE STR(PEEK(65304)—6*INT(PEEK(65304)/16,2) TO HOUR
STORE STR(PEEK(65305)—6*INT(PEEK(65305)/16),2) TO MIN
STORE STR(PEEK(65306)—6*INT(PEEK(65306)/16),2) TO SEC
IF $(HOUR,1,1)=' '
 STORE '0'+$(HOUR,2,1) TO HOUR
ENDIF
IF $(MIN,1,1)=' '
 STORE '0'+$(MIN,2,1) TO MIN
ENDIF
IF $(SEC,1,1)=' '
 STORE '0'+$(SEC,2,1) TO SEC
ENDIF
```

To compute the date:

```
STORE PEEK(65302) + 256*PEEK(65303) TO DAYS
STORE (DAYS+13173.000)/7.000 TO INTERVAL
STORE INTERAL — INT(INTERVAL) TO FRACTION
STORE INT((FRACTION * 7.000) + 0.5) TO DAYNUM
STORE 'SunMonTueWedThuFriSat' TO DAYNAME
STORE TRIM($(DAYNAME, 3*DAYNUM +1,3)) TO WDAY
RELEASE INTERVAL,FRACTION
RELEASE DAYNUM,DAYNAME
STORE 78 TO YEAR
STORE 1 TO MONTH
DO WHILE (DAYS>366) .OR. (DAYS = 366 .AND. YEAR/4 <> INT(YEAR/4))
 IF YEAR/4=INT(YEAR/4)
  STORE DAYS—366 TO DAYS
 ELSE
```

```
  STORE DAYS-365 TO DAYS
 ENDIF
 STORE YEAR+1 TO YEAR
ENDDO
IF YEAR/4+INT(YEAR/4)
 STORE T TO LEAP
ELSE
 STORE F TO LEAP
ENDIF
STORE T TO DECREASE
DO WHILE DECREASE
 DO CASE
 CASE STR(MONTH,2)$ ' 1 3 5 7 8 10' .AND. DAYS>31
  STORE MONTH+1 TO MONTH
  STORE DAYS-31 TO DAYS
 CASE STR(MONTH,2)$'4 6 9 11' .AND. DAYS>30
  STORE MONTH+1 TO MONTH
  STORE DAYS-30 TO DAYS
 CASE MONTH=2 .AND. DAYS>29 .AND. LEAP
  STORE 3 TO MONTH
  STORE DAYS-29 TO DAYS
 CASE MONTH=2 .AND. DAYS>28 .AND. .NOT. LEAP
  STORE 3 TO MONTH
  STORE DAYS-28 TO DAYS
 OTHERWISE
 STORE F TO DECREASE
ENDDO
RELEASE LEAP,DECREASE
STORE STR(YEAR,2) TO YR
STORE STR(MONTH,2) TO MON
STORE STR(DAYS,2) TO DAY
IF $(MON,1,1)=' '
 STORE '0'+$(MON,2,1) TO DAY
ENDIF
IF $(DAY,1,1)=' '
 STORE '0'+$(DAY,2,1) TO DAY
SET DATE TO &MON &DAY &YR
? 'Today is "+WDAY+' '
?? MON+'/'+DAY+'/'+YR+' '
??HOUR+':'+MIN+':'+SEC
RELEASE HOUR,MIN,SEC,DAY
RELEASE MON,YR,WDAY,RUN
```

APPENDIX J

```
RELEASE DAYS,MONTH,YEAR
SET TALK ON
RETURN
```

This routine can be used to date documents or records. The code may be instructive to readers interested in perpetual calendars.

SYSCMD.PRG

SYSCMD.PRG is a routine which allows 16-bit dBASE users to execute MC-DOS commands from within dBASE. This lets you execute commands like PIP, WS (WordStar), DSKRESET, or SC (SuperCalc) from dBASE. When the command or program is done, you are returned to the next line in your dBASE command file. You can maintain program control over every aspect of your application, even if software other than dBASE is required.

The code to create the 16-bit dBASE II program SYSCMD.PRG is listed below:

```
POKE 65280,83,177,145,186,196,0,205
POKE 65287,224,91,138,7,67,46,162,53
POKE 65295,193,138,200,181,0,191,54
POKE 65302,193,138,7,67,136,5,71,226
POKE 65310,248,198,5,0,177,150,186
POKE 65317,53,193,205,224,177,146
POKE 65323,205,224,177,145,186,200,0
POKE 65330,205,224,195,0,0
SET CALL TO 65280
CALL CMD
RETURN
```

APPENDIX K

K. A Brief Description of MC-DOS

This appendix gives a brief description of Gifford's MC-DOS and its relationship to Digital Research Inc.'s Concurrent DOS. This will be of value to those who have worked with Concurrent DOS and need to know what features are unique to MC-DOS. A list of features unique to MC-DOS is provided. Documentation of these features is presented in the previous chapters of this manual. Note that under MC-DOS the DATE, HELP, PRINTER, TYPE, and VCMODE commands have been modified. DR Net, the networking extension of Concurrent DOS, has been rewritten, so that a menu-driven NET command and a CONFIG.NET file integrate most of the user commands of DR Net. The NET program can operate in a command mode with the names of the Digital Research DR Net commands as arguments.

MC-DOS and Concurrent DOS

Concurrent DOS 3.1 can be viewed as having three parts: the user interface which processes your commands, the internal Concurrent DOS system function calls, and the hardware interface.

Under MC-DOS, the basic structure of the operating system designed by Digital Research Inc. remains unchanged. In systems equipped with the CompuPro MDRIVE/H, MC-DOS can utilize up to 1024K of solid state memory as a cache buffer for the hard disk, in place of the Concurrent DOS buffer in RAM. All of the system enhancements by Gifford Computer Systems are in areas designed to be modified, such as the user interface. These enhancements do not interfere with the normal operation of Concurrent DOS.

Gifford Computer Systems has modified MC-DOS's Terminal Message Process, and named it the **system shell.** The shell's main functions are:

1. Default terminal, printer, user number, and drive assignment for each account. These are stored in a file that can be modified to add or delete accounts, or to change their default assignments.

2. Printer and terminal I/O parameters. You can change these by modifying files that are provided with your system, so that no assembly language I/O programming is needed. This makes your system much more flexible and easy to modify.

3. User login and password functions. You can prevent unauthorized users from logging into your system, put restrictions on what programs can be executed by users, and specify the terminals that individuals can use.

Features Unique to MC-DOS

The following list includes features which have been implemented by Gifford Computer Systems. Included in the list are only those features of which the user needs to be aware. Features which are essentially invisible have not been listed. (See also **Chapter 2, Section 2.4.** The list includes:

SW, SW86 (Allow simultaneous processing of 8- and 16-bit programs by 8085/88 and 8086 CPUs respectively).

VTTYS (ASCII file to configures Virtual Terminals for concurrent processing)

Automatic RAM memory test (see **Chapter 3**)

User customization files and utilities (see **Chapter 3**):

DOWN.SUB
DWN.CMD
KEYS.CMD,
LOGIN.CMD
LOGIN.SUB
LOGOUT.SUB
MAIL.HLP
MCDOS.HLP
name.NET (DR Net only)
NAMENODE.CMD (DR Net only)
name.REM
SYSINIT.BAN
SYSINIT.CMD
SYSINIT.SUB
USERINIT.CMD
USERINIT.SUB

System security files and utilities (see **Chapter 3**):

PASSWD
HISTORY
LOGOUT
NEWUSER.CMD
PASSWORD.CM
PWASK.NET (DR Net only)
TIMELOG.CMD
TIMELOG.SUB
UTMP

APPENDIX K

Communications utilities (see **Chapter** 3)

AP.CMD	NETMOTD (DR Net only)
MAIL.CMD	NETUSERS (DR Net only)
MOTD	WRITE.CMD

Other utilities (see **Chapter** 3)

ATTACH.CMD	RESET.CMD
BUFFER.CMD	SETMEM.CMD
CCPMINIT.CMD	SETNET.CMD (DR Net only)
CONFIG.NET (DR Net only)	SYNC.CMD
CPY.CMD	SYSGEN.CMD
DOWN.CMD	TIME.CMD
DSKRESET	TYPE.CMD
FORMAT.CMD	UPTIME.CMD
LOGOUT	VCMODE.CMD
MAKE.CMD	WHO.CMD
NET.CMD (DR Net only)	

SIO Control (Programmable serial ports for interrupt driven input and output. See Appendix E).

Additional Notes on MC-DOS

Users should note that under MC-DOS several commands no longer perform as specified by Digital Research. The following notes describe the more important changes, and some supplementary Gifford commands.

In MC-DOS the DATE command does not set the system clock. See the CLOCK command. DATE still returns the correct time.

A DSKRESET command is built into MC-DOS and also is supplied as a separate file DSKRESET.CMD. This allows you to have two diskettes on a floppy-based system with no DSKRESET command on either diskette.

The HELP command is somewhat different in its syntax and structure, and MAIL uses a separate help facility.

APPENDIX K

The NEWUSER utility can be used to restrict accounts to the user areas, terminals, and optional default program assigned in the PASSWD file.

The PRINTER command in MC-DOS can take printer names, as well as printer numbers, as arguments, and can attach the user to printers at other nodes of a DR Net network. With the "?" option it displays all of the list device numbers and names.

The SW program enables Gifford's MC-DOS with 80286/Z80 or 8088/85 dual processors to support programs written for MP/M II, despite different queue calls in Concurrent DOS and MP/M II. The SW86 program enables an 8086-based Gifford MC-DOS system to support applications programs written for an 8080 or Z80.

The TYPE command in Gifford's MC-DOS can accept wildcards, list multiple files, change user areas, concatenate files, pause in its scrolling after an arbitrary number of lines, and access other DR Net nodes.

The USER command prompts the user to select a user number, rather than displays the current user number. The system prompt always displays the current user number, even for user area 0.

The VCMODE command in DYNAMIC mode allows a background process to write to the screen, rather than write to a buffer. The FLUSH option in MC-DOS discards detached screen output.

In MC-DOS, the Concurrent DOS command interpreter has been modified to enforce any user area access restrictions of the login PASSWD file, so that commands like USER, PIP, PRINT, DIR, or applications programs cannot access unauthorized areas.

MC-DOS and MP/M 8-16

MC-DOS incorporates many external features of MP/M 8-16, Gifford's enhancement of Digital Research's MP/M-86 multitasking, multiuser operating system. This similarity in the user interface is superficial, in that most MC-DOS command files are not identical to their MP/M 8-16 counterparts. The following pages identify differences between MC-DOS and MP/M 8-16.

APPENDIX K

MC-DOS has several features not available in MP/M 8-16. The most important is **support for networking**, since the DR Net network I/O system (NIOS) and utilities allow MC-DOS systems to be linked by Datapoint's ARCNET hardware. Gifford Computer Systems has enhanced DR Net to allow networks to be configured without assembly code. The maximum number of workstations in a DR Net network of MC-DOS systems surpasses by more than an order of magnitude the practical limit on the number of terminals attached to a single MP/M 8-16 system.

The MC-DOS operating system allows users to share an 80287 coprocessor, with a CPU like the Macrotech MI-286 dual processor (80286/Z80). In applications requiring intensive numerical processing, support for this device in a multiuser environment offers important advantages over MP/M 8-16, which does not allow a coprocessor to be shared. The same CHSET utility of MC-DOS that enables use of a coprocessor also allows certain programs that separate logical code from data to be shared by multiple users. This feature can conserve RAM, but the operator must determine by experiment which programs can be shared in this way.

MC-DOS accesses files faster than MP/M 8-16, through a more efficient, "hashed" directory search. Unlike MP/M 8-16, which uses XFCBs for both file passwords and time stamping, MC-DOS uses XFCBs only for passwords. MC-DOS systems that implement file passwords and timestamping show much less degradation in speed than MP/M 8-16 systems that implement these features. Repacking the hard disk is less important in MC-DOS than in MP/M 8-16.

Apart from the DR Net commands, several other MC-DOS files have no counterparts in MP/M 8-16: CHSET, INITDIR, LOGOUT.SUB, PRINT, SW86, and UPTIME. The functions of eleven MP/M 8-16 files are performed by MC-DOS files that have different names:

MC-DOS	MP/M 8-16	MC-DOS	MP/M 8-16
BUFFER.CMD	SWAP.CMD	SYSINIT.BAN	MPMINIT.BAN
CCPMINIT.CMD	VT.CMD	SYSINIT.CMD	MPMINIT.CMD
DATE.CMD	TOD.CMD	SYSINIT.SUB	MPMINIT.SUB
GENCCPM.CMD	GENSYS.CMD	SYSTAT.CMD	MPMSTAT.SUB
MCDOS.HLP	MPM.MLP	VCMODE.CMD	VIMODE.CMD
MCDOS.NDX	MPM.NDX		

APPENDIX K

Various MC-DOS commands, including CLOCK, CONSOLE, MAIL, PRINTER, TYPE, and WHO, have supplementary features that are not in their MP/M 8-16 counterparts.

Versions of the MODEM 8-16 telecommunications language written for MP/M 8-16 will not run under MC-DOS. A version of MODEM 8-16 is available from Gifford that runs under MC-DOS, and that occupies less than half the space in RAM that the MP/M 8-16 version required.

Both MC-DOS and MP/M 8-16 support a 200-character type-ahead buffer, to accept rapid character input.

Gifford's initial release of MC-DOS (version 1.00) did not include several commands from MP/M 8-16:

SCHED.CMD
SETWARP.CMD
SETPASSW.CMD
SETUTMP.CMD
STAT.CMD

The SETPASSW and SETUTMP security utilities are not used in MC-DOS, because the MC-DOS shell automatically restores a special password to the PASSWD or UTMP files if they are copied to the system drive. Readers who have MP/M 8-16 will find that STAT works in MC-DOS, but STAT is an MP/M-86 utility that Digital Research does not include with a Concurrent DOS license. Releases of MC-DOS later than 1.00 provide the CHRON.CMD utility to schedule recurring processes linked to the clock/calendar.

SETWARP is not in the MC-DOS command set, because neither disk emulation hardware nor system RAM can be used as a logical drive in MC-DOS. Systems that incorporate up to a megabyte of semiconductor disk like CompuPro's MDRIVE/H can use it as a cache buffer in MC-DOS, rather than as a volatile logical drive. This restricts to 1024K the amount of MDRIVE/H that MC-DOS can utilize, rather than the limit of 4096K in MP/M 8-16.

In MC-DOS, the control character ^D cannot detach a foreground process, nor attach a background process, as it can in MP/M 8-16. In MC-DOS a ^C unconditionally aborts the current foreground process, while MP/M 8-16 asks the operator to confirm (Y/N) that a process interrupted by ^C should be terminated.

APPENDIX L

L. Cables and Connections

The recommended length for RS-232 cables is anything under fifty feet. This is the maximum length specified by the IEEE standard. If you must use cables over fifty feet, use twisted pair, unshielded cables. Belden #9745 cable, 22 gauge has been used in runs over 500 feet, although we cannot guarantee that your devices will operate at such a distance. Shielded cables cannot be used for long runs. If you need to make your own cables, the following connections are what the CompuPro computer needs at the back panel. Be sure that line 4 does not have any connections. If it does, you should clip the wire to this pin before installing the cable to your computer. Use of this line can cause your system to freeze.

Pin Number	Connection
1	Ground
2	Device to computer
3	Computer to device
4	Leave disconnected
6	Terminal Input Handshake
7	Ground
20	DTR. Data terminal ready.
	-12V for logic 1
	+12V for logic 0

The RS-232 ports perform bidirectional hardware handshaking for all terminals that interface with the computer. If a terminal sends characters too fast to the computer, MC-DOS will signal to that device by lowering the RS-232 pin-six (6). This tells the transmitting device to stop sending characters to the computer. If more characters are sent to the computer after pin six (6) is lowered, the computer will send a CTRL S (^S) to the terminal. When the computer is ready to receive more characters, pin-6 is raised and becomes clear to send more characters. The TTYS and LPRS files allow you to set the computer to monitor DTR pin-20 with both terminals and printers in order not to output characters faster than the terminal or printer can receive them. Hardware handshaking is not enabled for remote terminals which communicate with the computer via modem.

APPENDIX L

In a DR Net local area network, the network controller boards must be linked by RG62 coaxial cable (93 ohms). Networks of only two nodes can be directly connected at distances of up to 2,000 feet.

Networks of more than two nodes must have each node connected to a repeater. A **passive repeater** can connect up to four cables (attached to a network controller or to an active repeater) over distances of up to 200 feet. An **active repeater** can connect up to eight cables (attached to a network controller or to another active repeater) over distances of up to 2,000 feet. The total distance separating two nodes of a network linked by active repeaters can be no more than 20,000 feet. The total number of nodes (single user or multiuser MC-DOS computers) linked in a single DR Net network can be no greater than 255.

APPENDIX M

M. Map of Logical Devices, Cables, Rooms, Users, and Status

Copy this page and fill in the information pertinent to your system. Update the map whenever you make changes, and post copies of the map at the system console and other strategic locations.

Device	Cable Name	Room	Operator	Status
Console	Sys. Console	_____	_____	Connected
TTY1	Terminal 1	_____	_____	Connected
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____
_____	_____	_____	_____	_____

APPENDIX N

N. Table of Hexadecimal Numbers

This Appendix and the next respectively describe **hexadecimal** (base-sixteen) numbers, and the **ASCII** (American Standard Code for Information Interchange) representation of keyboard characters.

Ordinary numbers are based on powers of ten. When we write a number like "1985" for the name of a year, we mean the value of the sum:

1 times ten to the 3rd power (1,000 x 1 = 1000),
plus 9 times ten to the 2nd power (100 x 9 = 900),
plus 8 times ten to the first power (10 x 8 = 80),
plus 5 times ten to the zeroeth power (1 x 5 = 5).

We do not need to write this as a sum, because the position of each numeral indicates the power of ten by which it should be multiplied. The way that we write numbers in daily life (as opposed, for example, to Roman numerals or bar codes) is called a "base-ten" or "decimal" system of numbers, because it is based upon powers of ten.

A 16-bit MC-DOS machine sometimes represents numbers on a base of sixteen, rather than ten. In numbers written in base-sixteen notation, the smallest two-digit number ("10") represents the quantity sixteen; the smallest three-digit number ("100") represents the quantity sixteen squared; the smallest four-digit number ("1000") represents sixteen cubed, and so forth. To be able to count past nine with single-digit numbers, six new numerals must be used: A (= decimal ten), B (= decimal eleven), C, D, E, and F (= decimal fifteen). Numbers in base-sixteen notation are called **hexadecimal** numbers, or "hex" for short.

In base-sixteen the hexadecimal number "1985" would represent the decimal number 6,533, or 1 times 16 cubed (4,096), plus 9 times 16 squared (256), plus 8 times 16, plus 5 times 1. The quantity 1,985 in decimal notation would be written "7C1" in hex, because it is the sum of 7 times 16 squared (= 1,792 decimal) plus C (twelve) times 16 (= 192 decimal), plus one, or "7C1" hex. The next page displays a table of the decimal (base-ten) equivalents of some hexadecimal (base-sixteen) numbers.

APPENDIX N

Hex	Decimal	Hex	Decimal	Hex	Decimal	Hex	Decimal	Hex	Decimal
1	1	10	16	100	256	1000	4096	10000	65536
2	2	20	32	200	512	2000	8192	20000	131072
3	3	30	48	300	768	3000	12288	30000	196608
4	4	40	64	400	1024	4000	16384	40000	262144
5	5	50	80	500	1280	5000	20480	50000	327680
6	6	60	96	600	1536	6000	24576	60000	393216
7	7	70	112	700	1792	7000	28672	70000	458752
8	8	80	128	800	2048	8000	32768	80000	524288
9	9	90	144	900	2304	9000	36864	90000	589824
A	10	A0	160	A00	2560	A000	40960	A0000	655360
B	11	B0	176	B00	2816	B000	45056	B0000	720896
C	12	C0	192	C00	3072	C000	49152	C0000	786432
D	13	D0	208	D00	3328	D000	53248	D0000	851968
E	14	E0	224	E00	3584	E000	57344	E0000	917504
F	15	F0	240	F00	3840	F000	61440	F0000	983040

Use the table above to calculate the decimal equivalent of any hex number of 5 or fewer digits, by adding the decimal value of each hexadecimal place. For example, FFFF hex equals the sum:

	F000	(=	61,440 decimal)
	F00	(=	3,840 decimal)
	F0	(=	240 decimal)
+	F	(=	15 decimal)
	FFFF	=	65,535 decimal.

Some MC-DOS commands and utilities use **paragraphs**, a unit equal to 10 hex (= 16 decimal) bytes. To convert a hex number of paragraphs to bytes, append "0" on the right to multiply by 10 hex. Thus, 5C hex paragraphs equal 5C0 hex bytes (= 1472 decimal). To convert bytes to paragraphs, divide by 10 hex (by dropping the last digit). For example, 4040 hex bytes = 404 hex paragraphs.

Specific RAM locations in MC-DOS programming utilities like DDT86 or SYSTAT are written in the form **s:o**, for s and o hexadecimal numbers of no more than four digits. (Here s is the **segment** of RAM, showing the hex number of a paragraph, counting from zero, the beginning RAM address). The value of o is the **offset** of the RAM address, the hex number of **bytes** above the beginning of s. For example, the RAM address A000:7850 denotes a RAM location A000 paragraphs + 7850 bytes (= A7850 hex bytes, or A785 paragraphs) above the beginning address (0000:0000) of RAM.

APPENDIX O

O. ASCII Character Set

The following is a chart of the ASCII character set as modified for printing, including Meta characters.

Dec.	Hex	Graphic	Name	(Meaning)
0.	00	^@	NUL	(used for padding)
1.	01	^A	SOH	(start of header)
2.	02	^B	STX	(start of text)
3.	03	^C	ETX	(end of text)
4.	04	^D	EOT	(end of transmission)
5.	05	^E	ENQ	(enquiry)
6.	06	^F	ACK	(acknowledge)
7.	07	^G	BEL	(bell or alarm)
8.	08	^H	BS	(backspace)
9.	09	^I	HT	(horizontal tab)
10.	0A	^J	LF	(line feed)
11.	0B	^K	VT	(vertical tab)
12.	0C	^L	FF	(form feed, new page)
13.	0D	^M	CR	(carriage return)
14.	0E	^N	SO	(shift out)
15.	0F	^O	SI	(shift in)
16.	10	^P	DLE	(data link escape)
17.	11	^Q	DC1	(device control 1, XON)
18.	12	^R	DC2	(device control 2)
19.	13	^S	DC3	(device control 3, XOFF)
20.	14	^T	DC4	(device control 4)
21.	15	^U	NAK	(negative acknowledge)
22.	16	^V	SYN	(synchronous idle)
23.	17	^W	ETB	(end transmission block)
24.	18	^X	CAN	(cancel)
25.	19	^Y	EM	(end of medium)
26.	1A	^Z	SUB	(substitute)
27.	1B	^[	ESC	(escape, alter mode, .SEL)
28.	1C	^\	FS	(file separator)
29.	1D	^]	GS	(group separator)
30.	1E	^^	RS	(record separator)
31.	1F	^_	US	(unit separator)

APPENDIX O

Dec.	Hex	Graphic	Name (Meaning)
32.	20		space or blank
33.	21	!	exclamation mark
34.	22	"	double quote
35.	23	#	number sign (hash mark)
36.	24	$	dollar sign
37.	25	%	percent sign
38.	26	&	ampersand sign
39.	27	'	single quote (apostrophe)
40.	28	(	left parenthesis
41.	29	)	right parenthesis
42.	2A	*	asterisk (star)
43.	2B	+	plus sign
44.	2C	,	comma
45.	2D	-	minus sign (hyphen)
46.	2E	.	period (decimal point, dot)
47.	2F	/	(right) slash
48.	30	0	numeral zero
49.	31	1	numeral one
50.	32	2	numeral two
51.	33	3	numeral three
52.	34	4	numeral four
53.	35	5	numeral five
54.	36	6	numeral six
55.	37	7	numeral seven
56.	38	8	numeral eight
57.	39	9	numeral nine
58.	3A	:	colon
59.	3B	;	semi-colon
60.	3C	<	less-than sign
61.	3D	=	equal sign
62.	3E	>	greater-than sign
63.	3F	?	question mark

APPENDIX O

Dec.	Hex	Graphic	Name (Meaning)
64.	40	@	at-sign
65.	41	A	upper-case letter ALPHA
66.	42	B	upper-case letter BRAVO
67.	43	C	upper-case letter CHARLIE
68.	44	D	upper-case letter DELTA
69.	45	E	upper-case letter ECHO
70.	46	F	upper-case letter FOXTROT
71.	47	G	upper-case letter GOLF
72.	48	H	upper-case letter HOTEL
73.	49	I	upper-case letter INDIA
74.	4A	J	upper-case letter JERICHO
75.	4B	K	upper-case letter KAPPA
76.	4C	L	upper-case letter LIMA
77.	4D	M	upper-case letter MIKE
78.	4E	N	upper-case letter NOVEMBER
79.	4F	O	upper-case letter OSCAR
80.	50	P	upper-case letter PAPPA
81.	51	Q	upper-case letter QUEBEC
82.	52	R	upper-case letter ROMEO
83.	53	S	upper-case letter SIERRA
84.	54	T	upper-case letter TANGO
85.	55	U	upper-case letter UNICORN
86.	56	V	upper-case letter VICTOR
87.	57	W	upper-case letter WHISKEY
88.	58	X	upper-case letter XRAY
89.	59	Y	upper-case letter YANKEE
90.	5A	Z	upper-case letter ZEBRA
91.	5B	[	left square bracket
92.	5C	\	left slash (backslash)
93.	5D	]	right square bracket
94.	5E	^	up-arrow (caret)
95.	5F	_	underscore

APPENDIX O

Dec.	Hex	Graphic	Name (Meaning)
96.	60	`	(single) back quote (grave accent)
97.	61	a	lower-case letter alpha
98.	62	b	lower-case letter beta
99.	63	c	lower-case letter charlie
100.	64	d	lower-case letter delta
101.	65	e	lower-case letter echo
102.	66	f	lower-case letter foxtrot
103.	67	g	lower-case letter golf
104.	68	h	lower-case letter hotel
105.	69	i	lower-case letter india
106.	6A	j	lower-case letter jericho
107.	6B	k	lower-case letter kappa
108.	6C	l	lower-case letter lima
109.	6D	m	lower-case letter mike
110.	6E	n	lower-case letter november
111.	6F	o	lower-case letter oscar
112.	70	p	lower-case letter pappa
113.	71	q	lower-case letter quebec
114.	72	r	lower-case letter romeo
115.	73	s	lower-case letter sierra
116.	74	t	lower-case letter tango
117.	75	u	lower-case letter unicorn
118.	76	v	lower-case letter victor
119.	77	w	lower-case letter whiskey
120.	78	x	lower-case letter xray
121.	79	y	lower-case letter yankee
122.	7A	z	lower-case letter zebra
123.	7B	{	left brace
124.	7C	|	vertical bar
125.	7D	}	right brace
126.	7E	~	tilde
127.	7F	^?	DEL (delete, rub out)

APPENDIX P

P. Notes

APPENDIX P

APPENDIX P

GLOSSARY

account: A name that can be presented at the login prompt to access the system. MC-DOS can require that an encrypted password be presented with a valid account name before an operator can access a system. Account names are also used by the AP, MAIL, HISTORY, TIMELOG, WHO, and WRITE utilities, and by the reminder file utility.

application program: A program that performs a business or scientific function such as word processing or structural analysis. As opposed to a system utility, like PIP, or a language, like FORTRAN.

archive attribute: File attribute that indicates whether a file has been modified since last being backed up (archived). It is used with PIP to avoid backing up files needlessly.

ARCNET: A networking hardware standard (baseband, token-passing) developed by Datapoint Corporation.

argument: A symbol that is entered after the program name in a command line to give the program specific instructions.

ASCII: American Standard Code for Information Interchange. It is a widely accepted standard for the digital representation of numbers, letters, and symbols. "ASCII" terminals and printers can convert data between ASCII format and human readable text. In contrast, binary or hexadecimal data has meaning only in the context of the specific hardware or software which uses it.

attribute: A file flag that can be set ON or OFF to determine how that file will be treated by the operating system. For example, an attribute can be set ON to prevent a file from being modified.

background: Processing that occurs without affecting a terminal screen is said to be in the background. With Virtual Terminal software, screens other than the current, visible (foreground) screen are background screens.

backup: A copy of a file or files that are created to insure against irrecoverable loss of data.

baud rate: The speed at which serial data is transferred, as with a modem or an RS-232 device. Usually baud rate divided by 10 equals bytes per second, so that, for example, 19.2 kilobaud equals about 1.92 kilobytes per second.

binary data: In the most general sense, any digital representation of data. Since no formal standard exists to define binary data, the term is often used to mean non-ASCII data.

bit: The smallest unit of data that a computer can manipulate. Eight bits make a byte, 1,024 bytes make a kilobyte.

bus: A physical path for electrical signals.

boot: The process of loading an operating system into memory, usually by physically resetting the system. Most systems when powered up will automatically attempt to read the "boot tracks" of the system disk into memory and then execute that code, which in turn instructs the computer to read into memory the remainder of the operating system. By completing this process, the computer "pulls itself up by its bootstraps."

buffer: A temporary data storage area, typically used to compensate for the comparative slowness of another component.

cache: A type of buffer used to enhance disk performance.

command file: A program file which can be run by typing its file name (omitting its file type) followed by a carriage return. 16-bit command files have the file type .CMD, and 8-bit command files have the file type .COM.

concatenate: To combine two or more files into a single file. This can be done by the PIP command or the TYPE command.

configuration: The installation of hardware or software so that it is compatible with other hardware and software in a system. Configuration also refers to the specific hardware and software that make up a system.

console: A terminal. This manual will use the word console only to refer to the **system console**, that is, the terminal which Multiuser Concurrent DOS first talks to during the boot sequence.

CPU: Central Processing Unit. The microprocessor which runs the operating system and programs.

dBASE II: An application program by Ashton-Tate designed for database management.

default: The node, disk drive, user number, password, program, options, terminal, or list device that will be assumed by the operating system when the user gives a command without specifying this information.

diagnostics: Programs used to identify system problems.

directory: A listing of files. The directory can be evoked with the **DIR** or **SDIR** commands. Directory also refers to the area on a disk that tells the operating system where files are stored on that disk. If this latter type of directory is damaged, the files to which it refers may be irrecoverably lost.

DR Net: The networking module of the Multiuser Concurrent DOS operating system, providing software protocols to link computers.

encryption: A scheme used to encode information so that it is unintelligible without the encryption algorithm and "key." It is analogous to a combination lock for data.

executable code: Code that is machine readable and ready to be run by the computer without translation. Source code, by contrast, is human readable and must be compiled or assembled into executable object code before it can be run by the CPU.

Extended File Control Block: Part of a directory listing that allows the optional use of file password protection.

Extended Input/Output System (XIOS): The software interface between the Multiuser Concurrent DOS operating system and the specific physical hardware which makes up your computer. In a DR Net network, the Network Input/Output System (NIOS) is the software that talks to the network controller hardware.

file: A collection of data, characters, or instructions stored on disk.

File Control Block (FCB): A structure used by the operating system to describe attributes of files.

file name: The name given to a file. Technically, **file name** refers to the first part of a file's name and precedes the optional **file type.** The **file name** cannot exceed 8 characters. In the file WS.CMD, the **file name** is WS.

file type: The optional second part of a file's name, separated from the **file name** by a period. The **file type** cannot exceed 3 characters. In the file WS.CMD, the **file type** is CMD.

floppy diskette: A flexible disk made of mylar that is coated with magnetic oxide and encased in square cardboard. It is used for storing data. Floppy diskettes are inexpensive and highly portable.

function keys: A special set of preprogrammed or programmable keys on a keyboard. They are typically used to condense multiple character strings into a single keystroke.

handshaking: The means by which the flow of data between peripheral devices is controlled so as to prevent data overrun. The facility for handshaking must be built in to the devices being used and must be mutually compatible. The most popular hardware handshake protocol is DTR, and the most popular software handshake is XON/XOFF.

hardware: The physical elements of a computer system, in contrast to software.

hard disk: A rigid disk that is coated with magnetic oxide and used for storing data. Generally, hard disks store much more data and operate much faster than floppy diskettes.

hexadecimal notation: Notation for the base-16 number system that uses 16 different characters, 0 through 9 and A through F. This is a "natural" number system for microcomputers because there is a one-to-one relationship between the 256 possible combinations of two-digit hexadecimal characters and the 256 possible bytes.

input: Data sent to a program by an I/O (input/output) device such as a keyboard, disk, or tape drive.

interface: A device or program that allows devices and/or programs to communicate.

I/O: Abbreviation for input/output.

kilobyte: 1024 bytes, commonly written as 1K.

local device: A device that is physically attached to your local network node.

local node: The computer to which your terminal is physically attached, as opposed to "remote nodes" (other computers in a DR Net network).

logical drive: All or part of a physical drive, represented by a letter (A: through P:). Under DR Net, you can arbitrarily assign a logical drive letter to remote physical drives on the network.

MC-DOS: Multiuser Concurrent DOS, Gifford's enhanced version of Digital Research's Concurrent DOS operating system.

MC-DOS prompt: See **"system prompt."**

megabyte: 1024 kilobytes.

modem: A device that allows communication over phone lines between two computers or between a computer and a terminal.

multiuser system: A computer that allows two or more physical terminals to be used simultaneously for running programs.

multitasking: Synonymous with **concurrent.** The ability to run two or more programs simultaneously from a single physical terminal, as in Multiuser Concurrent DOS with Virtual Terminals software.

network: A collection of single-user and/or multiuser computers so integrated as to permit the sharing of I/O devices such as logical drives, tape drives, plotters, printers, and spooler systems.

node: A single-user or multiuser computer attached to a network. A node can function as a self-sufficient computer, when not attached to the network.

object code: Machine readable code that can be executed by the CPU. See **executable code.**

operating system: Software such as Multiuser Concurrent DOS that provides a common, productive environment for application programs, utility software, languages, and I/O devices.

option: A value supplied by a user usually from a list of special symbols, that specifies what a command should do.

output: Data produced by a program and sent to an I/O device such as a terminal, printer, plotter, disk, or tape drive.

password: A string of symbols used to maintain security, serving as a kind of software padlock. Passwords can be put on nodes, files, and accounts to restrict unauthorized users from accessing sensitive data.

peripheral device: Input/Output hardware such as a terminal, disk drive, tape drive, printer, plotter, etc., as opposed to the CPU or RAM memory which are part of the main computer system.

program: Software that contains instructions for the CPU to execute.

queue: A data structure used by Multiuser Concurrent DOS to allow processes to communicate in an orderly manner with the operating system and with one another.

RAM: Random Access Memory. A program must be loaded into RAM before it can be executed by the CPU. The expression "random access" refers to the fact that any randomly selected RAM address takes exactly the same amount of time for the CPU to access as any other RAM address, unlike disk or tape records which require variable amounts of mechanical motion to access. The implication is that RAM is extremely fast since it has no mechanical constraints. RAM also implies read-write memory, as opposed to ROM (read-only) memory.

read-only: Able to be read but not modified.

read-write: Able to be read and/or modified.

record: In terms of a disk file, a 128-byte segment of data.

Requester Configuration Table (RCT): These tables specify which virtual terminals are currently mapped over the network.

source code: Human readable code that must be compiled or assembled into executable object code before it can be run by the CPU.

remote device: A device that is not physically attached to your local network node; for example, a logical drive on another node.

requester: A network node that can receive files or other network services from another node.

Resident System Process (RSP): Programs that are incorporated in the operating system so as to be permanently resident in memory.

resource map: The current configuration of a terminal's network resources such as disk drives and printers. The resource map can be easily changed or "remapped." Each virtual terminal screen has an independent resource map.

sector: A section of a disk track. Sector size is determined by the software that is used to format the disk. Typical sector sizes range from 128 bytes to 2 kilobytes.

server: A network node that can send files to another node.

software: Programs or data. Anything that can be stored on a disk, on tape, or in RAM or ROM memory. This is as opposed to hardware, which includes the physical devices that actually store and manipulate software.

spooler: A system of programs that buffers output intended for a printer so that in case the printer is busy, the output data is stored automatically in a temporary disk file until the printer becomes free. This allows the terminal to go on to other tasks while waiting for a printer to become free.

string: A sequence of ASCII characters.

SuperCalc 86: An application program by SORCIM for creating and maintaining electronic spread sheets.

syntax: The formal order or grammar of a computer command.

system console: The terminal to which Multiuser Concurrent DOS signs on during the initial boot sequence.

system manager: The person responsible for seeing that a computer installation is used efficiently.

system program: A utility program such as DIR or PIP that is of general use regardless of the specific application for which the computer is being used. This is as opposed to an application program such as an accounting package.

system prompt: A three-character string displayed by Multiuser Concurrent DOS to indicate that the operating system is waiting for you to type a command. The system prompt characters consist of the current user number, the current logical drive, and the "greater than" (>) character.

Terminal Message Processor: The part of Concurrent Multiuser DOS that controls the user interface. Gifford's revised Terminal Message Processor, or TMP, is called the "shell."

text editor: Any application program designed to help in the creation or modification of ASCII, human readable text, or source files.

time stamp: A feature of a file's directory listing that can keep track of when that file was created, last accessed, or last modified. The SET and INITDIR commands control time and datestamping in MC-DOS.

track: On a floppy diskette or a hard disk, concentric magnetic rings that can contain data, programs, directory information, or the operating system.

upward compatible: A highly desirable feature that allows new software or hardware to accommodate products designed to run with older versions of that same software or hardware.

upgrade: Hardware or software additions or enhancements to increase or improve the capabilities of an existing system.

user area: Any of sixteen logical attributes of files on logical drives, whose value must be a whole number from 0 to 15. In MC-DOS, the basis of file security is that acccounts can be restricted in their access to user areas.

virtual terminal: A feature of MC-DOS that allows up to four programs to share the screen and keyboard of a properly equipped serial terminal.

wildcard characters: The ? and * characters that can be used to substitute for literal characters when specifying file names. The ? stands for any one or fewer characters, and the * stands for any string of characters.

WordStar: An application program by MicroPro International designed for word processing, an alternative to the ED text editor.

write-protect: A file attribute that prevents a file from being modified, but allows it to be read or copied without presenting a password.

XFCB: See **Extended File Control Block.**

XIOS: See **Extended Input/Output System.**

REFERENCES

Please Note: Any reference to other manuals in the text of this manual does not denote that the referenced manual has been included with your Gifford MC-DOS system. If you wish to purchase the referenced manual, please call our Sales Department.

ARCS100 USER'S GUIDE. Standard Microsystems Corporation, 1983.

BITS, BYTES AND BUZZWORDS. Mark Garetz. CompuPro, a Division of Godbout Electronics, 1983.

CONCURRENT CP/M OPERATING SYSTEM PROGRAMMER'S REFERENCE GUIDE. Digital Research Inc., 1984. (While this publication does not come with your system, it can be purchased from Gifford.

CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE. Digital Research Inc., 1984.

CONCURRENT CP/M OPERATING SYSTEM USER'S GUIDE SUPPLEMENT FOR THE PRINT SPOOLER SYSTEM. Digital Research Inc., 1984.

CONTROL DATA CARTRIDGE MODULE DRIVE HARDWARE AND MAINTENANCE MANUAL. Control Data Corporation, 1981.

CP/M-86 OPERATING SYSTEM USER'S GUIDE. Digital Research Inc., 1981.

CUSTOMIZING YOUR MULTIUSER CONCURRENT DOS SYSTEM, Revision 4.0. Gifford Computer Systems, a Division of Zitel Corporation, 1984.

DISK 2 TECHNICAL MANUAL. CompuPro, a Division of Godbout Electronics, 1982.

DR NET USER'S MANUAL, Revision 3.0. Gifford Computer Systems, a Division of Zitel Corporation, 1984.

GCS-80 USER'S MANUAL, Revision 2.0. Gifford Computer Systems, a Division of Zitel Corporation, 1984.

REFERENCES

LANS100 USER'S GUIDE, Revision 1. Intercontinental Micro Systems Corporation, 1984.

MULTIUSER CONCURRENT DOS USER'S MANUAL, Revision 1.50. Gifford Computer Systems, a Division of Zitel Corporation, 1984.

TIP OPERATOR'S GUIDE, Version 4.xx. Alloy Computer Products, a Division of Alloy Engineering Company, Inc., 1981.

VIRTUAL TERMINALS USER'S MANUAL, Revision 6.0. Gifford Computer Systems, a Division of Zitel Corporation, 1984.

WORDSTAR INSTALLATION MANUAL, Version 3.0. MicroPro International, 1981.

MULTIUSER CONCURRENT DOS USER'S MANUAL INDEX