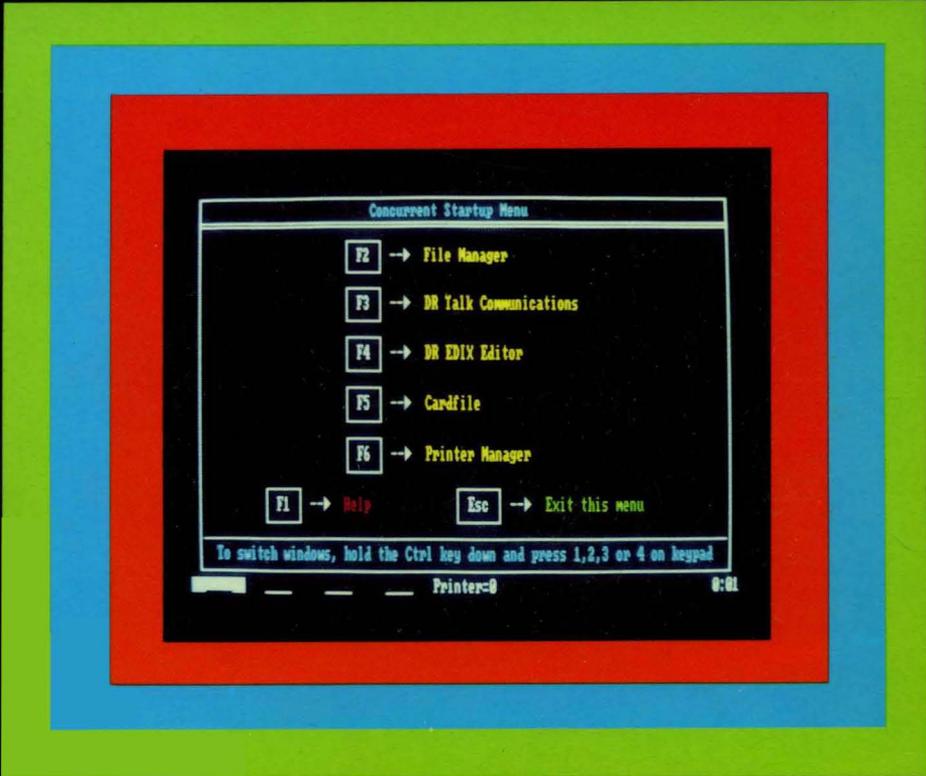


MARK DAHMKE **Using**  
**Concurrent**  
**PC DOS™**



---

# Using Concurrent PC DOS

**OTHER BOOKS BY THE AUTHOR**

---

**Microcomputer Operating Systems (1982)**

**The Byte Guide to CP/M-86 (1984)**

---

# Using Concurrent PC DOS

Mark Dahmke

**McGraw-Hill Book Company**

New York St. Louis San Francisco Auckland Bogotá  
Hamburg Johannesburg London Madrid  
Mexico Montreal New Delhi Panama  
Paris São Paulo Singapore  
Sydney Tokyo Toronto

**Library of Congress Cataloging-in-Publication Data**

Dahmke, Mark.

Using Concurrent PC DOS.

Bibliography: p.

Includes index.

1. Concurrent PC DOS (Computer operation system)

I. Title.

QA76.76.063D34 1986 005.4' 469 85-15473

ISBN 0-07-015073-7

Copyright © 1986 by McGraw-Hill, Inc. All rights reserved.  
Printed in the United States of America. Except as permitted  
under the United States Copyright Act of 1976, no part of this  
publication may be reproduced or distributed in any form or by  
any means, or stored in a data base or retrieval system, without  
the prior written permission of the publisher.

1234567890 DOC/DOC 893210876

**ISBN 0-07-015073-7**

The editors for this book were Steven Guty and Vivian Koenig,  
the designer was Naomi Auerbach, and the production  
supervisor was Teresa F. Leaden. It was set in Century Schoolbook  
by Byrd Data Imaging.

Printed and bound by R. R. Donnelley & Sons Company.

*To my sister Patricia*



---

# Contents

<b>Chapter 1. Introduction</b>	<b>1</b>
<b>What Is Concurrent PC DOS?</b>	<b>1</b>
<b>What Is an Operating System?</b>	<b>1</b>
<b>The DOS Family Tree</b>	<b>3</b>
<b>The Scope of This Book</b>	<b>5</b>
<b>Chapter 2. Concurrent PC DOS Compatibility</b>	<b>6</b>
<b>Concurrent PC DOS Compatibility</b>	<b>6</b>
<b>PC-DOS, TopView, and the IBM PC AT</b>	<b>7</b>
<b>Concurrent CP/M-86</b>	<b>9</b>
<b>Chapter 3. The Concept of Concurrency</b>	<b>10</b>
<b>Multiple Programs per Machine</b>	<b>11</b>
<b>Multiple Programs per User</b>	<b>14</b>
<b>Multiple Users per Machine</b>	<b>15</b>
<b>Background and Unattended Tasks</b>	<b>16</b>
<b>Chapter 4. Windows in Concurrent PC DOS</b>	<b>18</b>
<b>How Windows Work</b>	<b>18</b>
<b>Using Windows to Run Two or More Programs</b>	<b>22</b>
<b>Window Control Keys in Concurrent PC DOS</b>	<b>22</b>
<b>The WINDOW Command</b>	<b>24</b>
<b>The Window Manager</b>	<b>26</b>
<b>A Cut-and-Paste Example</b>	<b>28</b>
<b>Chapter 5. Diskettes and Fixed Disks</b>	<b>34</b>
<b>DOS Media Versus CP/M Media</b>	<b>34</b>
<b>Subdirectories and User Numbers</b>	<b>35</b>
<b>Logical and Physical Disk Drives</b>	<b>35</b>
<b>Floating Drives</b>	<b>38</b>
<b>The System Disk</b>	<b>38</b>
<b>Memory Disk</b>	<b>38</b>

<b>Chapter 6. The File Manager</b>	<b>40</b>
<b>The Special Function Keys</b>	<b>41</b>
<b>Selecting Groups of Files</b>	<b>41</b>
<b>The File Directory</b>	<b>44</b>
<b>The Subset of Files Command</b>	<b>46</b>
<b>The Drive Selection Menu</b>	<b>47</b>
<b>Manipulating Files</b>	<b>48</b>
<b>Context-Sensitive Help</b>	<b>48</b>
<b>The TYPE and PRINT Commands</b>	<b>49</b>
<b>The COPY Command</b>	<b>51</b>
<b>The FREE MEMORY Command</b>	<b>55</b>
<b>The SYSTEM SETUP Command</b>	<b>56</b>
<b>Assign Function Key Submenu</b>	<b>57</b>
<b>System Defaults</b>	<b>60</b>
<b>The SIZE/DATE Command</b>	<b>61</b>
<b>Chapter 7. Creating and Modifying Menus</b>	<b>67</b>
<b>What are Menus?</b>	<b>67</b>
<b>Menu File Organization</b>	<b>68</b>
<b>COPYMENU</b>	<b>75</b>
<b>Chapter 8. The Printer Manager</b>	<b>76</b>
<b>Starting the Printer Manager</b>	<b>76</b>
<b>Stopping the Printer Manager</b>	<b>82</b>
<b>Suspending the Printer Manager</b>	<b>82</b>
<b>Chapter 9. The Backup/Restore Facility</b>	<b>83</b>
<b>BackRest Files</b>	<b>84</b>
<b>Disk Media Considerations</b>	<b>85</b>
<b>Control File Options</b>	<b>85</b>
<b>Exceptions</b>	<b>86</b>
<b>Invoking BackRest</b>	<b>87</b>
<b>Chapter 10. Customizing Your Personal Computer</b>	<b>89</b>
<b>Autostart Batch Files</b>	<b>89</b>
<b>Changing Colors</b>	<b>91</b>
<b>The Printer Manager</b>	<b>92</b>
<b>Function Keys</b>	<b>92</b>
<b>Cardfile</b>	<b>92</b>
<b>The Help Facility</b>	<b>93</b>
<b>Passwords</b>	<b>93</b>
<b>The Memory Disk</b>	<b>93</b>
<b>Using Menus to Create a Turnkey System</b>	<b>93</b>
<b>Your PC as a Multiuser System</b>	<b>94</b>
<b>Other Considerations</b>	<b>97</b>

<b>Appendix I</b>	<b>Cardfile</b>	<b>98</b>
	<b>Starting Cardfile</b>	<b>98</b>
	<b>Selecting Commands</b>	<b>99</b>
	<b>Editing, Updating, and Deleting Cards</b>	<b>100</b>
	<b>Searching for a Card</b>	<b>101</b>
	<b>Printing Labels and Cards</b>	<b>101</b>
<b>Appendix II</b>	<b>DR Talk</b>	<b>103</b>
	<b>Starting DR Talk</b>	<b>103</b>
	<b>Using DR Talk as a Dumb Terminal</b>	<b>104</b>
	<b>Receiving and Transmitting Files</b>	<b>107</b>
	<b>The Dialing Directory</b>	<b>107</b>
	<b>Defaults and Configuration Parameters</b>	<b>108</b>
<b>Appendix III</b>	<b>DR EDIX</b>	<b>110</b>
	<b>Creating a Document</b>	<b>110</b>
	<b>Editing a Document</b>	<b>111</b>
	<b>Block Commands</b>	<b>111</b>
	<b>Translating and Searching</b>	<b>113</b>
<b>Appendix IV</b>	<b>Application Programs under Concurrent PC DOS</b>	<b>115</b>
	<b>PC-DOS/MS-DOS 3.0 Compatibility</b>	<b>115</b>
	<b>Using Windows with Application Programs</b>	<b>115</b>
	<b>Programs Tested on Concurrent</b>	<b>116</b>
	<b>Other Considerations</b>	<b>117</b>
<b>Appendix V</b>	<b>Summary of Features and Extensions</b>	<b>119</b>
	<b>PC-DOS Compatibility</b>	<b>119</b>
	<b>Features</b>	<b>119</b>
<b>Appendix VI</b>	<b>Concurrent PC DOS Command Reference</b>	<b>121</b>
<b>Appendix VII</b>	<b>Bibliography</b>	<b>129</b>
<b>Appendix VIII</b>	<b>Glossary</b>	<b>130</b>
	<b>Index</b>	<b>141</b>



---

# Preface

Concurrent PC DOS was designed by Digital Research as the next logical step in the evolution of the MS-DOS type of operating system. It combines the features of Concurrent CP/M-86 and PC-DOS, and it supports windows and multiuser operation. Soon after Concurrent PC DOS was announced, IBM announced its TopView, an add-on product that gives PC-DOS concurrency and windows but not multiuser features. The primary advantage of concurrency and windows on a single-user system is that instead of having just one console and the ability to run just one program at a time, the user may execute several applications at once. In this way, personal productivity can be increased since it is no longer necessary to wait to complete one program before gaining access to other program and data files. This book introduces the PC-DOS/MS-DOS or CP/M user to the concepts of windows and concurrency and shows various strategies users can take to maximize their productivity when using a personal computer.

Digital Research recently announced a new version of Concurrent PC DOS that takes advantage of the extended memory addressing feature offered by the Intel-Lotus standard. This feature allows memory addressing beyond the 640K limit of PC DOS. Up to 4M bytes of main memory can now be used to store programs and data.



---

# Acknowledgments

I would like to thank Kathy Strutynski of Digital Research for her support on this project. Also, I would like to thank Ken Livingston for letting me tie up several megabytes of disk space on his system for a couple of months. My editor at McGraw-Hill, Steve Guty, Ken Livingston, and Martin Q. Peterson were instrumental in determining the organization and content of this book. Special thanks go to Vivian Koenig, editing supervisor at McGraw-Hill, for her excellent work on this book. I would also like to thank my parents for making this author possible.



---

**Using Concurrent  
PC DOS**



---

## Chapter

# 1

## Introduction

### What Is Concurrent PC DOS?

Concurrent PC DOS is the first in a series of new operating systems for the IBM Personal Computer and IBM-compatible computers. Developed by Digital Research, the creators of CP/M, Concurrent PC DOS combines many of the features of previous operating systems into a more or less unified whole.

### What Is an Operating System?

An operating system can best be visualized as a traffic cop whose job it is to make certain that all the various hardware components of your personal computer perform their functions in the correct sequence at the correct time.

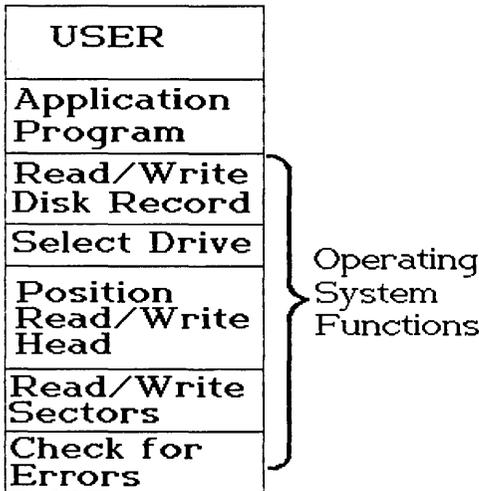
Without an operating system, you would have to enter programs the way people did in the early days of microcomputers and in the even more primitive days of the 1950s on mainframe computers—by throwing switches on the front panel of the machine.

The main function of an operating system is to isolate the hardware from the user. For example, the interface to a floppy disk drive consists of several levels of hardware and software (see Figure 1.1). The top-most level is the one the user or programmer is most familiar with: the Read/Write record. Whether you are programming in Basic or Fortran or using a word processor, you will be operating at this level.

The moment you invoke the disk Read/Write function, the operating system takes over and calls on ever more sophisticated and detailed programs to make sure the disk drive is ready, to select the appropriate drive, to position the Read/Write head of the drive over the correct track, to wait for the correct sector, to read or write the sector, and to check for errors.

Each one of these activities might take only a few milliseconds, but in that short time, hundreds of machine-level instructions are being executed. The operating system also takes care of the way the data is mapped onto the disk. A disk directory must be maintained or your data or programs will be lost. All of these tasks must be performed every time you run a program or copy a file from one disk to another.

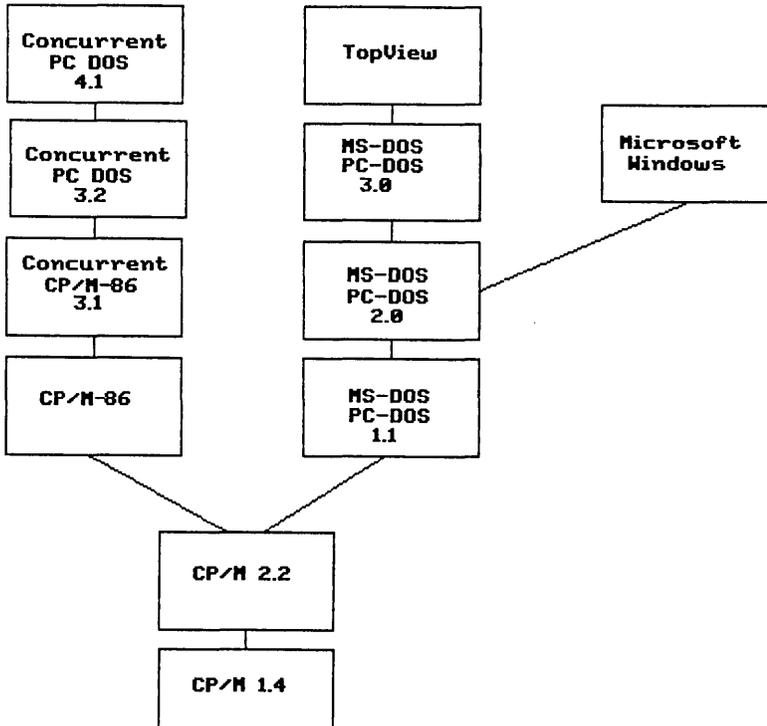
Without the operating system, the programmer would be faced with a much greater challenge in writing applications programs, and without the standardization enforced by the operating system, each programmer would use different techniques, which in most cases would lead to incompatibility between programs.



**Figure 1.1** Operating system levels. When an application program needs to perform an operating system function, such as reading a record from a disk file, it calls on the operating system. The operating system then calls another program that selects the disk drive to use, calls the "position disk head" routine, and then calls a routine that actually reads the record.

## The DOS Family Tree

Even with standard operating system methods and disk formats, there have been many incompatibilities in the personal computer world. In the mid-1970s each hardware manufacturer designed its own operating system simply because there were no standards to follow. Whole operating system empires rose and fell in those days. For several years



**Figure 1.2** The DOS family tree. CP/M 1.4 was the first major release of CP/M on the 8080 and Z80 microprocessors. In 1979, CP/M 2.2 was announced, which quickly became the standard for 8-bit 8080 and Z80 microcomputers. CP/M-86 was developed for the 8086 and 8088 microprocessors, which are used in the IBM PC and compatibles. MS-DOS and PC-DOS are similar to CP/M and can even run some old CP/M-86 programs, but Versions 2.1 and 3.0 of PC-DOS have significant differences. Concurrent CP/M-86 allowed multiple windows, but wouldn't run all MS-DOS and PC-DOS software. Concurrent PC DOS is functionally at the same level of sophistication as TopView, IBM's concurrent add-on to DOS, and Microsoft Windows.

operating systems such as Altair's DOS and Processor Technology's Helios DOS were widely used. Radio Shack's TRSDOS is a survivor from that period as is Apple's DOS. CP/M (Control Program for Microcomputers) was written by Gary Kildall to run on the 8080 microprocessor, the predecessor of the Intel 8088 in the IBM PC XT and the 80286 used in the IBM PC AT. He later formed a company called Digital Research. CP/M had many of the elements of previous operating systems and has been widely copied since it achieved popularity and displaced other, older operating systems as a de facto standard.

Microsoft's DOS (MS-DOS), which is the parent of IBM's PC-DOS, was developed to replace Digital Research's CP/M-86. Since Digital Research tried to make CP/M-86 compatible with the original CP/M, many people viewed it as a limited operating system that had not grown with hardware technology. Microsoft's DOS was virtually identical to CP/M-86, but the disk format and a few other features had been changed to take advantage of the new hardware available (see Figure 1.2).

Because IBM chose Microsoft DOS as its standard operating system, CP/M-86 did not achieve much popularity. PC-DOS 2.1 brought many revisions to the original DOS and added new functions that greatly enhanced its usefulness, but made it even more incompatible with CP/M-86.

In an effort to reclaim some of its lost market share, Digital Research announced Concurrent CP/M-86 and later added the PC Mode, which allowed users to run many PC-DOS programs in the Concurrent environment. Concurrency allows a user to run several programs at once on a single personal computer. Concurrent CP/M-86 was still not fully compatible with what IBM PC users had come to expect in their computer, so it did not meet with much success.

Table 1.1 Features present in each operating system.

<b>Feature</b>	<b>CCP/M-86</b>	<b>PC-DOS</b>	<b>TopView</b>	<b>Concurrent</b>
<b>PC-DOS Disk Media</b>		●	●	●
<b>CP/M-86 Disk Media</b>	●			●
<b>PC-DOS 2.1 Function Calls</b>		●	●	●
<b>Windows</b>	●		●	●
<b>Concurrency</b>	●		●	●
<b>Multuser</b>	●			●

Concurrent PC DOS can best be thought of as a superset of CP/M-86 and PC-DOS. It runs programs written for either operating system, supports concurrency and windows, and comes with a number of useful utilities, such as a word processor and a communications program. In addition, Concurrent PC DOS can support up to three users. Table 1.1 shows the features present in these operating systems.

### **The Scope of This Book**

The purpose of this book is to explain the concepts of concurrency and windows and to describe some of the features and utilities provided by Concurrent PC DOS.

It is assumed that you have some knowledge of either CP/M-86 or PC-DOS. If you don't, refer to Appendix VII for a list of books on the subject.

Appendices I, II, and III describe the features and capabilities of Cardfile, DR Talk, and DR Edix respectively. These application programs are included with Concurrent PC DOS and provide 3 × 5 card index, communications, and file editing services.

Also, Appendix IV shows how to run application programs such as Lotus 1-2-3 and Wordstar on Concurrent PC DOS.

Appendix V summarizes the features and extensions of Concurrent, and describes how they relate to PC-DOS and CP/M-86.

Appendix VIII is a glossary of terms related to features of Concurrent PC DOS.

---

Chapter

# 2

## Concurrent PC DOS Compatibility

### Concurrent PC DOS Compatibility

Compatibility is becoming harder to define as each day passes, but we can start by breaking the issue down into levels of hardware and software compatibility. First, there are hardware differences between the IBM PC*jr*, the PC, the XT, and the AT. Just because the microprocessors in all these products are based on the 8086/8088 instruction set does not guarantee that programs will run on all of them identically.

Several hardware issues must be dealt with when determining compatibility: memory address space; hardware devices, such as the video display, the disk drives, and the serial ports; and internal devices such as direct memory access (DMA) channels and counter/timer circuits. If a program communicates directly with any of these devices, instead of going through the software interface provided by the operating system, programs written for one type of machine may not run on other models even if the microprocessor is the same.

The manufacturer of a computer generally provides software interfaces that the programmer can use to avoid communicating directly with the hardware. In the IBM PC series, the interface is known as the basic input/output system (BIOS). The BIOS is stored in permanent read-only memories (ROMs), which are on the main circuit board of all IBM PCs.

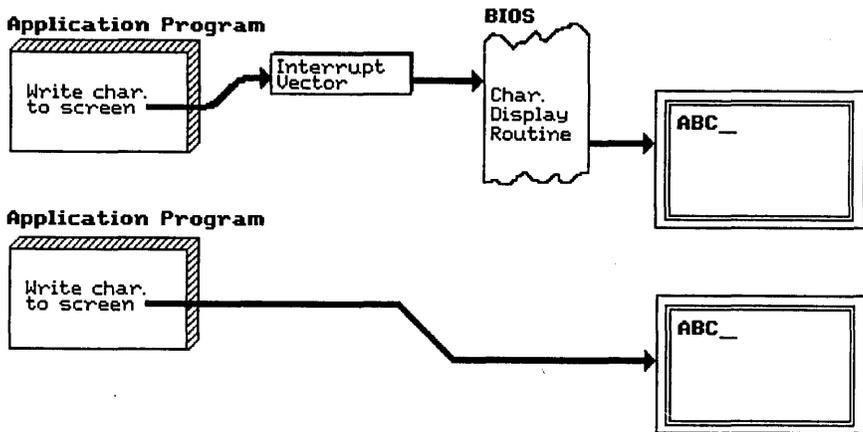
Some software vendors aren't satisfied with the performance of the

BIOS interface because it runs slower than reading or writing information directly to a device. For example, a word processor may appear to run up to five times faster if the programmer reads and writes characters directly to the video display adapter memory. The problem with this method is that the program won't work properly if IBM or another vendor introduces a new computer or a new video display adapter that is incompatible with the old one. Figure 2.1 shows how the interrupt vectors work.

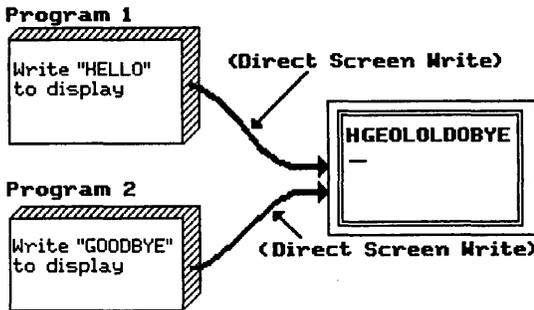
Another major problem with direct hardware access comes when a user tries to run more than one program on the personal computer at once. For example, two programs are loaded which write directly to the video display, bypassing the operating system interface. If both programs assume that they "own" the video display (which would be the case in the single-program-per-system case), both would attempt to display text at the same time and the display would be unreadable (see Figure 2.2).

**PC-DOS, TopView, and the IBM PC AT**

With the introduction of concurrency, many old programming techniques can come back to haunt both the software vendor and, unfortu-



**Figure 2.1** BIOS versus direct screen access. An application program can either call the basic input output system (BIOS), which is resident in read-only memory, to display a character on the screen, or it can write the character directly to the screen memory buffer. The second approach may make the program run faster, but it also means that the program won't be compatible with all future IBM computers or compatibles. It also makes concurrency difficult to manage.



**Figure 2.2** Two programs attempting to use the same device. The two programs shown here running concurrently have not checked to see if the video display is being used by the other program, and the resulting screen display is garbage. Going through the BIOS or the operating system proper would have avoided this problem.

nately, the user. Whether you decide to use Concurrent PC DOS or IBM's TopView, some programs won't work the way they did under PC-DOS 2.1 or 3.0. As mentioned earlier, any program that writes directly to the video display could act strangely if run under Concurrent or TopView. Programs that directly control either the serial ports or the parallel printer ports could cause problems, while programs that cause the video display mode to be changed by issuing commands directly to the video controller circuit could cause major problems. Programs that use some form of copy protection that is incompatible with TopView's or Concurrent's compatibility rules may not run. Programs that modify internal interrupt vectors (BIOS or PC-DOS interface entry points) could cause the system to crash. A program that loads other programs or data into absolute memory locations might overwrite another program running concurrently.

What this means is that just on the basis of operating system and hardware dependencies, an application program may not run properly in either TopView or Concurrent PC DOS. Using an IBM-supplied operating system such as TopView is no guarantee that the program will work correctly.

The PC AT presents even more serious problems. At the time of this writing, neither PC-DOS nor TopView supports more than 640K bytes on the AT. To take advantage of the 3M byte extended address space of the AT, you must switch to the Xenix operating system from IBM.

In most respects, the PC AT is upwardly compatible with the PC XT and the original PC. However, programs that depend on the timing of instruction execution (such as executing a program loop 1000 times to produce an exact time delay) will not operate as the programmer intended. As discussed above, if a program bypasses the BIOS or the DOS for the sake of execution speed and assumes that certain fixed memory locations are used by the DOS, the program may not work in the AT because some memory locations have been redefined or altered in function.

Other, less noticeable differences can cause problems. For example, some software products use machine instructions that exist in the 8088 microprocessor but which are undocumented by Intel, the chip manufacturer. These instructions are present in all current versions of the 8088, but may not be present in the 80286 used in the AT. Since the manufacturer doesn't support these hidden instructions, no attempt was made to include them in the 80286. Certain other machine instructions act differently in the 80286 as well. In many cases, there is simply no way of determining which programs will run and which will crash when used with a new operating system or a new hardware design, except to test them thoroughly before trusting them.

### **Concurrent CP/M-86**

Concurrent PC DOS is really a superset of PC-DOS and Concurrent CP/M-86. This means that features were taken from both and combined to form the best of both systems. Naturally, in any merger of different programming approaches and standards, some compromises had to be made. Concurrent PC DOS has most of the features of PC-DOS, but some familiar features might be missing. The chief example is the lack of command line editing keys in Concurrent PC DOS. In PC-DOS 2.1 and 3.0, you can reissue the last command entered by pressing function key F3. Other function keys allow for editing of the command line character by character, which eliminates any retyping. Concurrent PC DOS doesn't support any of these advanced editing features, although other features, such as the File Manager, make up for the loss.

One important gain is that Concurrent PC DOS will read diskettes formatted for PC-DOS, Concurrent CP/M-86, or CP/M-86. When a program attempts to read a file from a diskette or a fixed disk, Concurrent first checks to see whether the disk was formatted for CP/M-86 or PC-DOS, so that the application program will be unaware of any difference in the media.

---

Chapter

# 3

## The Concept of Concurrency

Anyone who is familiar with a mainframe computer, such as an IBM 370 or VAX, has some notion of what concurrency means. In the mainframe world several other terms are used to describe it. The word timesharing is generally used when several interactive video display terminals or typewriter terminals are connected via direct coax or telephone line to a central computing facility. The terms multiprogramming or multiuser are often used when the primary purpose of the mainframe is to execute batch jobs that are submitted to it through card readers or other input devices. The most common type of system is usually a mix of timesharing and batch processing.

In a timesharing system, each user can pretend that he or she is the only person using the mainframe computer. In fact, a good timesharing system will not even let a user know what other work is in progress on the machine (see Figure 3.1). This is known as virtual operation. The term virtual means “not in actual fact” and refers to a mode of operation in which each user appears to have all of a computer’s resources at his or her disposal, but in fact is sharing them with many other users.

In many respects a personal computer running an operating system like PC-DOS looks and acts just like a timesharing system because it gives the user complete and sole access to the computer and all devices attached to it. Programs that run on the personal computer can be said

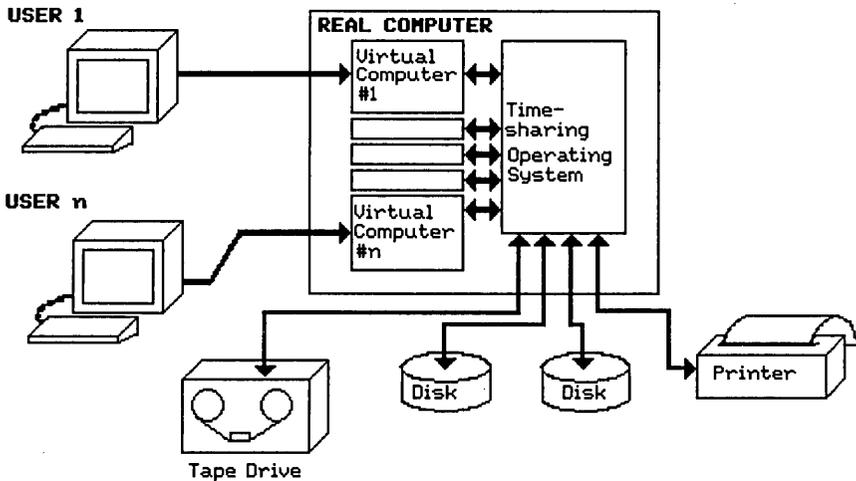


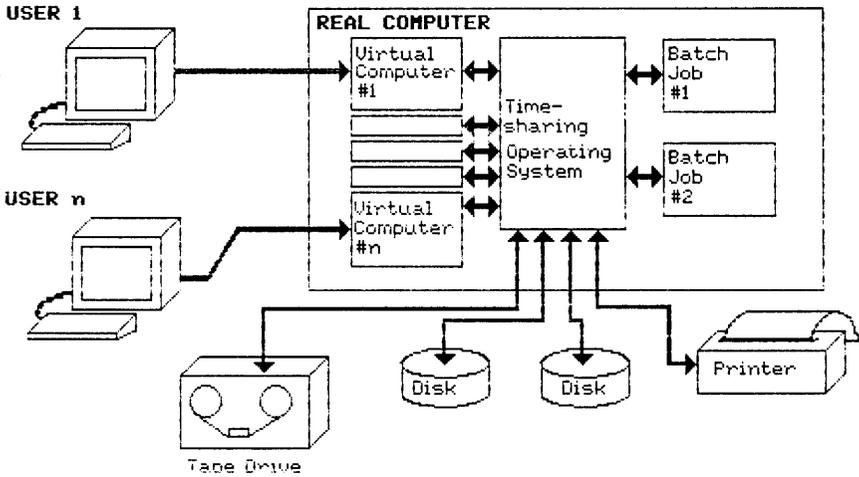
Figure 3.1 Timesharing. Each user on a mainframe has his or her own “virtual computer.” The user is generally unaware of the activities of other users.

to “own” the central processor, the disk drives, the printer, and the console (the video display and keyboard). Most personal computers were designed for this type of operation—there are no facilities in the hardware or operating system for the execution of multiple programs at one time (see Figure 3.2).

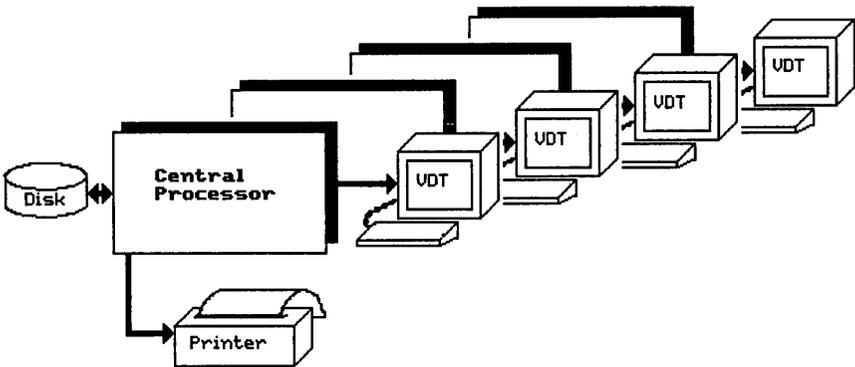
Concurrent PC DOS provides the features that allow a user to execute more than one program at a time. To do this, Concurrent takes control of all the hardware devices and intercepts all the requests that application programs make to the operating system or the BIOS (see Figure 3.3). By “owning” all of the attached devices, Concurrent controls which programs have access to the hardware and thereby avoids conflicts that might occur between programs (see Figure 3.4). For example, if there is only one printer, the user can’t send output to it from two different programs simultaneously; the result would be unreadable garbage. Concurrent PC DOS establishes the ownership of a printer and allows only one program to print at a time.

### Multiple Programs per Machine

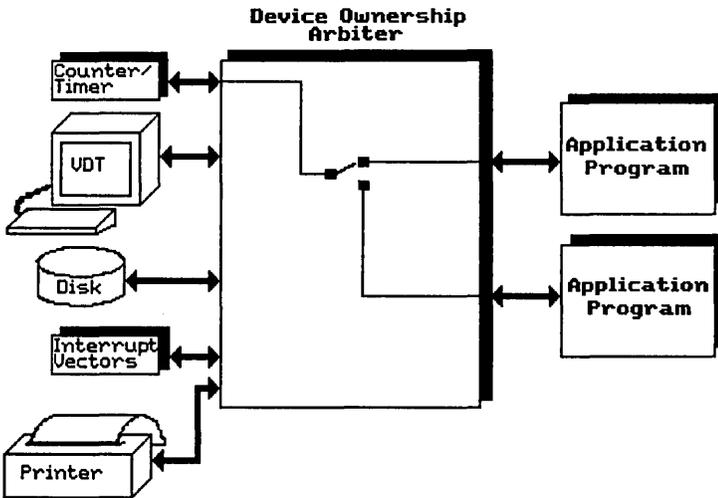
The simplest case for multiprogramming or concurrency is that of running two or more programs on the same microprocessor. A simple form of multiprogramming occurs every time you run the print spooler in PC-DOS 2.1 or 3.0. When the print spooler is loaded into memory, it first builds itself into the lowest available memory space just above the



**Figure 3.2** Timesharing plus batch jobs. Essentially identical to the system shown in Figure 3.1, this form of timesharing allows batch or noninteractive programs to run in the “background.” These programs have access to all system services, just as interactive users do, but can operate unattended.



**Figure 3.3** Timesharing in a virtual environment. In a completely virtual operating environment, each user appears to have the computer all to him or herself. The entire memory space and all disk space appears to be available. This is done by setting up separate virtual memories for each user.



**Figure 3.4** Device ownership. Within a timesharing or concurrent operating system, a table is kept for each physical device attached to the computer. A portion of the program scheduler acts as the arbiter by granting or refusing access to a device depending on its current ownership.

operating system itself. This is accomplished by a DOS function call that terminates the program but causes its memory space to be reserved so that no other programs can overwrite it. Before terminating itself, the print spooler sets up some special interrupts and redirects other BIOS entry points so that it can take control of the processor every few milliseconds; in effect, it steals time from the program you are running in order to print a file in "background" mode. For every tick of the internal system clock, a call is made to the print spooler which in turn reads characters from the file to be printed and send them to the current printer device.

This process is quite similar to what happens in Concurrent PC DOS except that the procedure is formalized to enable more than just one other program to run at a time. Concurrent accomplishes this by taking control of all the interrupt vectors and all the DOS function calls so it can manage the use of the microprocessor at all times. A program called a dispatcher is built into Concurrent, and it keeps track of all the programs that are running and decides which program should be allowed to run next, as shown in Figure 3.5.

Programs that are waiting for keyboard input or access to a disk file

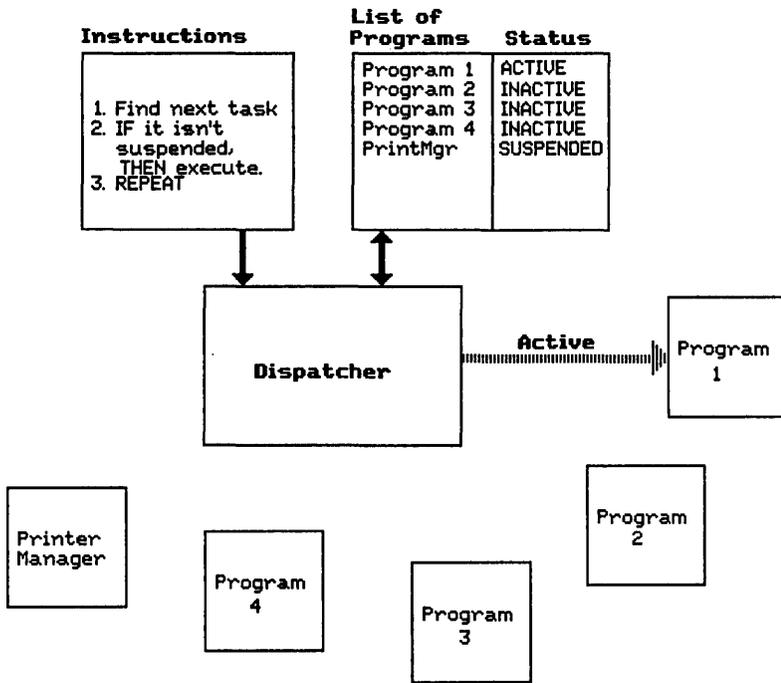
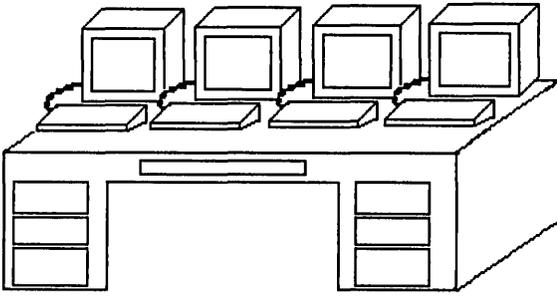


Figure 3.5 The dispatcher. Similar to the arbiter, the dispatcher controls the use of the central processor. A list of tasks or programs is kept along with the status of each task. If a task has not been suspended by some outside action, the dispatcher will activate it.

can be put on hold to keep them from slowing down other programs that don't need external input.

### Multiple Programs per User

In a mainframe timesharing system, multiprogramming is used to allow many people to share the resources of a large computer. The real advantage of concurrency on a personal computer is not to allow several people to share one computer, but to allow one person to be more productive by being able to use one computer for many functions simultaneously (see Figures 3.6 and 3.7). For example, you are using your word processor and are right in the middle of editing a file. You would like to refer to your calendar or address book, but to do this you must exit the word processor and start another program. In Concurrent PC DOS, you can simply switch to another window and run

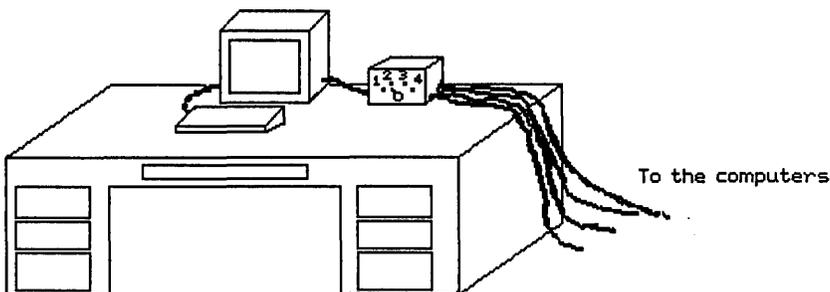


**Figure 3.6 Multiple programs per user.** This figure shows one rather clumsy method of obtaining concurrency. A separate terminal could be supplied for each program.

Cardfile, a program supplied with Concurrent that simulates a standard Rolodex card file. Alternately, you could load a spreadsheet to check the numbers from a sales report for use in your letter. It is even possible, using windows (which will be described in the next chapter), to cut and paste from one program to another. You could copy the totals column from the spreadsheet directly into your word processor instead of retyping the numbers.

### Multiple Users per Machine

Concurrent PC DOS even allows you the option of turning the two serial ports, COM1 and COM2, into ports for extra console devices.



**Figure 3.7 Multiple programs per user (another method).** This method uses one terminal, but it has a switch allowing the terminal to be connected to one of four data communications lines that connect it to a mainframe or minicomputer.

Using serial video display terminals, you could support up to three users on one IBM PC or compatible computer (see Figure 3.8). The disadvantage of this technique is that the response time for each user could increase drastically if each user puts heavy demands on the system.

One good use for a two- or three-user arrangement is order entry or light duty inventory applications that involve only small amounts of keyboard input and display. Obviously a PC AT will be able to handle a much greater work load than a PC or PC XT because of its faster processor.

### Background and Unattended Tasks

One of the main uses for concurrency in a single-user system is to run background programs that don't require user input or may run in unattended mode. Printer spoolers and long-running batch programs are the primary candidates for background execution. Actually, in Concurrent PC DOS no distinction is made between foreground and background as such; you can simply start a program in one window and switch to another window to continue your work. The Printer Manager that is supplied with Concurrent is a special case since it doesn't occupy a window when it executes. It actually installs itself into the operating system when it is loaded and executes concurrently with the other four windows. In this respect it can be called a background task.

Some programs that take a long time to execute, such as compilers and other development utilities, also work well in a concurrent system.



**Figure 3.8** The IBM PC AT can be used as a multiuser system under Concurrent PC DOS. The two serial ports, COM1: and COM2:, can be configured as additional consoles.

For example, if you are developing an application in Fortran, you could set up a batch file that contains all the parameters needed to tell the compiler where the file is and what options you desire and then set up another command line for the linker. You could start the batch file and switch windows to continue working (perhaps writing additional source code) while the compiler works on your program. Because your computer is working on more than one program at once, you may notice some speed degradation, but if the background program is fairly input/output bound (meaning that it spends more time reading and writing the disk than executing instructions) you may not notice much of a difference in speed.

# 4

## Windows in Concurrent PC DOS

### How Windows Work

The concept of windows, or screen segmentation, goes back to the late 1960s and had its origins at Xerox's Palo Alto Research Center. Simple windows are called tiled windows because the video or graphics display screen is divided into halves or some other number and split either vertically or horizontally. Each segment of the screen is treated as a separate logical display device by the computer. Figure 4.1 shows a typical split screen arrangement. The top and bottom halves can be scrolled independently. This format is often used in spreadsheets or word processors to allow the user to see and edit two files at once, or to look at two different parts of the same spreadsheet or text file.

Actually the split screen or tiled window concept has been around much longer than most people realize; some mainframe software packages split the screen even in the early 1960s.

The modern form of display windows involves overlapping or potentially overlapping windows on the same display. When text is written to a window that is covered by another window, it is referred to as writing to an "occluded" window (see Figure 4.2).

In most windowing systems, the operating system keeps track of virtual windows that are stored in reserved areas of main memory. Each virtual window is large enough to handle the entire real window display. For example, if the video display is capable of showing 80

columns by 25 lines, each virtual window must be allocated 2000 characters (see Figure 4.3).

When a program prints text characters to the screen, it normally hands them over to a program in the operating system (or the BIOS) which actually performs the chore of finding the right location on the screen and placing the characters there one at a time. If windows are being used, the application program still hands the characters over to the operating system, but instead of just writing them to the screen the operating system writes them to the virtual window in memory. If the window is currently being displayed on the screen the operating system will also copy the new characters to the real screen to update it, as shown in Figure 4.4. If the window isn't being displayed, the operating system returns control to the application program without displaying the text.

As mentioned earlier, windows can overlap each other and they can be less than the full size of the screen. If a character is written to a virtual window, the operating system must check to see if that

```

Version 4.1

Copyright (c) 1985, Digital Research Inc.
60 Garden Court
Box DRI
Monterey, CA 93942

C>$ysdisk C:\Con-Dos
C)
C)
C>suspend off

Suspend = OFF
C)
-----
Enter the Number to delete or RETURN to exit 1
C>prtsc

C)wmenu
Window Manager installed.

C)
  _____ Printer=0 C Win 0:02

```

Figure 4.1 Split screen display. Shown here is one form of windowing called "tiled" windows, because they are placed adjacent to one another like tiles. Each window scrolls independently of the other. Normally a special function key switches keyboard input from one window to the other.

```

C:\COMMAND COM : \DOS : \HB\C)
C:\PCTALK : \FONTRIX : \WS C)suspend off
C:\GENE : \DOC : \AS
C:\CONFIG BAK : AUTOEXEC OLD : FA Suspend = OFF
C:\CALDTA NBD : IDEAS NBD : AU C)cd \gendesk
C:\MCD NBD : OMNI NBD : QU
C:\PERSONAL NBD : LEDFORD NBD : \FO C)dir
C:\GEMSYS NBD : \GEMAPPS : GE
C:\PICTLIB2 : \PICTLIB3 : GE Directory of Drive C (DOS Media)
C:\CONFIG SYS : MEMORY DOC : \FS
C:\STARTUP2 BAT : STARTUP3 BAT : CC C:\. : \. : DESKTOP AP
C:\FONTS : AUTOEXEC BAT : C) C:\DESKLO ICN : DESKHI ICN
57 File(s) 1886336 bytes f 7 File(s) 1886336 bytes free
System files exist
C) C)

Enter the Number to delete or RETURN to exit 1
C)prtsc

C)wmenu
Window Manager installed.

C)
Printer=0 C Min 8:04

```

Figure 4.2 An example of occluded windows.

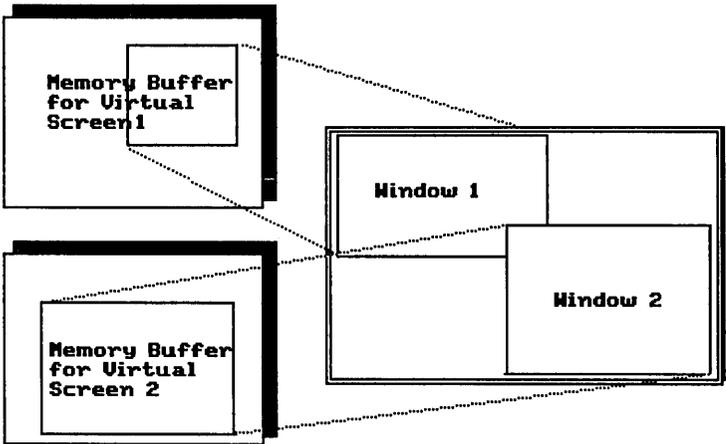
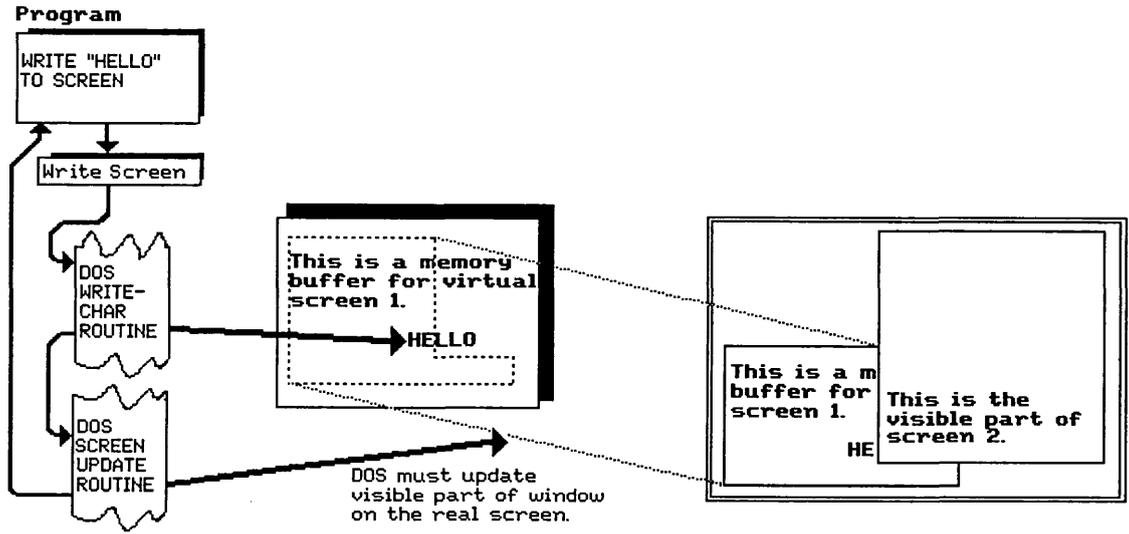


Figure 4.3 Virtual to real display mapping. In most windowing systems, separate "virtual" windows are kept for each program. All or part of each virtual window is then mapped (through software or hardware) onto the one "real" display window.



**Figure 4.4** Writing to an occluded window. When virtual windows are used, the problem of keeping the screen up to date is complex. The simple method would be to rewrite the entire screen each time a character is displayed, but this method would involve substantial overhead for the processor and cause the screen to flicker. Therefore, additional software is added to test for window boundaries. If the character will be seen on the real display, its location is computed and the screen updated. If the character will be hidden under another window, there is no reason to write it to the real display so it is simply stored in the virtual window for later use.

particular character is occluded by another window. For example, if only one corner of a window is covered by another window, the screen would have to be updated except for the covered corner. Updating windows can become a complex problem if they are allowed to overlap each other.

### **Using Windows to Run Two or More Programs**

Windows are essential to the operation of Concurrent PC DOS. Four virtual consoles are provided, so you can run four different programs simultaneously. Each virtual console is assigned a window. The window can display the entire virtual console or just a segment of it, depending on your needs. At any time you can switch from displaying the entire window to displaying just a small part of it, or vice versa. This allows you to set up a window for a background program that you want to observe, but do not need to watch all the time. You could set up the window for that program as a small two- or three-line window in the upper corner of the screen, while your word processor or other program occupies the remainder of the screen. You can now keep an eye on the other program and, if necessary, expand its window to cover up what you are working on to see more.

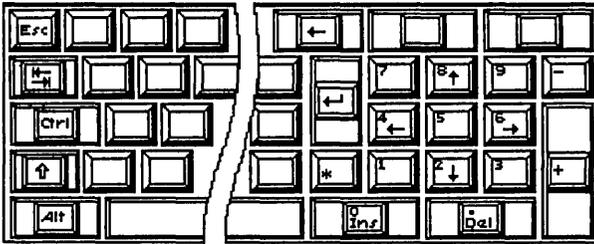
Another good use for multiple windows would be to monitor a telecommunications line. DR Talk, a useful serial communications and data transfer program, is included with Concurrent and is described in Appendix II. It allows you to turn your personal computer into a smart terminal and dial up information services, mainframes, or other personal computers using either a telephone modem or a direct wire RS-232 connection. You could start DR Talk in a window in the corner of your screen, log on to another computer or information service, and issue commands to download your electronic mail or other files. While this is happening, you could switch to another window and continue your work. When the downloading process is complete, you will see this in the small window and can switch back to DR Talk and bring the window up to full size again. None of these actions will cause your other work to be disturbed. It is just like having four computers or four terminals on your desk at once.

### **Window Control Keys in Concurrent PC DOS**

Since windows are part of Concurrent rather than part of an application program, there must be some way of controlling their placement

and selection without disturbing the program that is running. This is accomplished by using function key combinations that are normally not used by other programs. The default assignments are given in Figure 4.5. It is possible to reassign these functions if they conflict with another program's use of the keyboard; the Resetting Window Switching Keys section in Chapter 6 explains how to do this.

These five key combinations allow you to select the window you want to work in and to switch between a partial and the full display of the window. You will also notice that when you select a window as the current one, the status line at the bottom of the screen will show the current window and program name (if a program is running) by highlighting it.



**NOTE: Only numeric keypad keys may be used to select windows.**



**Select Window One**



**Select Window Two**



**Select Window Three**



**Select Window Four**



**Switch between Full and Partial Window Display**

**Figure 4.5** Window control keys. Concurrent PC DOS uses five keys to activate and control windows. These keys can be reassigned if they conflict with function keys in other application programs. Holding down Ctrl and pressing 1, 2, 3, or 4 on the numeric keypad will cause windows 1, 2, 3, or 4 to be activated respectively. Other windows return to the background mode. Ctrl-Del will switch the active window between full screen size and partial screen, if the window is smaller than the whole screen.

## The WINDOW Command

When Concurrent is loaded into memory, it sets up the four windows as full screens, with window 1 displayed and the other three “hiding” behind it. As discussed earlier, if you press Ctrl-2 you will see window 2, Ctrl-3 will bring up window 3, Ctrl-4 will display window 4, and Ctrl-1 will return you to the first window. The WINDOW command is one way of modifying the characteristics of a window, although for most practical purposes you will want to load the resident Window Menu Manager, which is described in the next section, and use it to alter window characteristics.

From the A> or C> prompt, you can (assuming that the WINDOW command is on the current disk) type WINDOW VIEW to see the current settings for all four windows.

<i>No.</i>	<i>PRow</i>	<i>PCol</i>	<i>NRows</i>	<i>NCols</i>	<i>VRow</i>	<i>VCol</i>	<i>Tracking</i>	<i>Display</i>	<i>FGColor</i>	<i>BGColor</i>
1	3	5	15	70	1	1	Row	Color	White	Magenta
2	40	30	10	40	1	1	Row	Color	White	Black
3	20	50	5	30	1	1	Row	Color	Blue	Black
4	1	1	24	80	1	1	Row	B&W	White	Black

The fields denote the physical row and column, the number of rows and columns, the virtual row and column, whether cursor tracking is enabled, which display the window is on, and the foreground and background colors of the window.

Figure 4.6 shows a typical window and how all the above parameters are related. For example, the physical row and column represent the location of the window with respect to the physical display screen. If PRow=5 and PCol=10, then the window will appear on the screen at row 5, column 10. The virtual position of the window is an imaginary point in the lower left corner of the window that defines the window’s position with respect to the full screen display of that window. This allows Concurrent to keep track of the window’s position as the display underneath it scrolls.

Tracking is normally enabled, and it determines whether the cursor is always displayed in or allowed to leave the visible part of a window.

It is possible to use both a color and a monochrome display at the same time under Concurrent. For example, windows 1 and 2 could be displayed on the monochrome monitor while windows 3 and 4 could be assigned to the color monitor. This ability can be used for several purposes; it allows you to monitor two full screens at once instead of having small or overlapping windows on one screen, as well as exploit certain application programs that may be able to use a dual screen configuration.

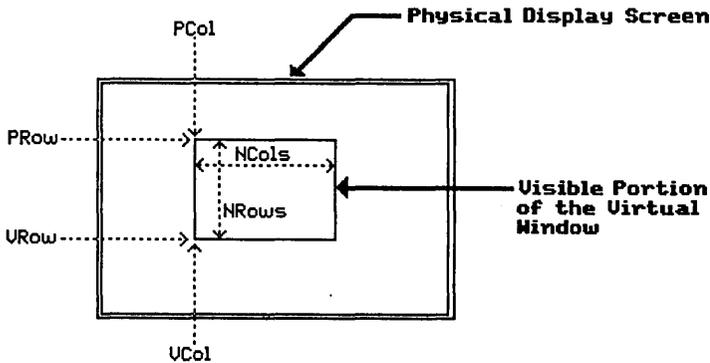
Foreground and background colors may be changed for each window. When using the color monitor, you have the choice of black, white, red, yellow, cyan, magenta, blue, and green. When using a monochrome display, you are of course limited to one color.

Several other commands are available in the WINDOW utility. To change any of the above parameters, you can enter the WINDOW CHANGE command followed by the keywords and the number of the window to alter. For example,

```
WINDOW CHANGE NUMBER=1 NR=20
```

would change the number of rows displayed in window 1 to 20 rows. The TOP option will bring the specified window to the top, which means that it will become the active window. The FULL option will cause the selected window to occupy the full screen, and the WRITE option can be used either to write the contents of the screen to a text file or to create a batch file called WSETUP.BAT that may be used to reload the window parameters to their current state at a later time.

The TYPE=C parameter causes the complete screen to be written to a file specified by the FILE= parameter. The TYPE=W option will write only the visible part of the window to the specified file. TYPE=S will write a batch file that contains four WINDOW command lines



**Figure 4.6** Window parameters. In Concurrent PC DOS, the Window Manager keeps track of several variables for each window. First are the physical row and column (PRow, PCol), which determine the location on the real display of the virtual window's upper left corner. The number of rows and columns (NRows, NCols) tells Concurrent how much of the virtual window to display. The virtual row and column (VRow, VCol) indicate what portion of the virtual window to display.



you may invoke them simply by moving the cursor to highlight the command and then pressing the Enter key. In place of the main menu line, you will then get a submenu or prompt line that will ask you to use the arrow keys to position the window or adjust the parameter values.

You can easily adjust the size of a window by selecting the SIZE menu item and pressing Enter. Figure 4.8 shows the SIZE command prompt line, which displays the number of rows and the number of columns. The arrow keys may be used to reduce or expand the screen size.

Similarly, you can easily change the placement of a window by selecting the PLACE menu entry and pressing Enter. Figure 4.9 shows the full screen and the PLACE command being selected. After pressing Enter, the menu line will show the current placement of the screen. By using the arrow keys, you can "drag" the window around on the screen until it is where you want it. To return to the main window menu line, press the Escape key (ESC).

As you can see in Figure 4.10, reducing the size of a window causes the windows that appear behind the current window to become visible.

```

C: PRISC  EYE : PRISC2  BAK : BADSPOT  3  : PRISC2  ASM : PRISC  ASM
C: SCREN  BAS : PRISC  BAK
      12 File(s) 1712128 bytes free

C>prisc

C>cd \con-dos

C>dir

Directory of Drive C (DOS Media)

C:\.          : \.          : STARTUP1 BAT : MENU      DAT : PREPMENU DAT
C: FDSTART  1  : HDINSTAL BAT : AUTOEXEC  NEW : LOADCCPM COM : LETTER  TXT
C: HDAUTO   BAT : STARTSUB  HD  : START     HD1  : HDDOSSUB  BAT : HELP    CMD
C: CARDFILE DIS : CARDFILE  TRM : CARDDECK  BAT : CLEAR    BAT : DRTALK   EXE
C: DRTALK   DIR : DRTALK    DEF : DRTALK    KEY : FDMENU   DAT : HDMENU   DAT
C: CONSOLE1 TXT : WINDOW1   TXT
      27 File(s) 1712128 bytes free
System files exist
C>

```

WINDOW CHANGE N=1 NRows=21, NCols=63 (Pressing ↑, ↓, →, ← sizes window)

Figure 4.8 Adjusting the size of a window using the Window Manager.

```

Concurrent DOS 4.1
Co
      12 File(s) 1712128 bytes free

C>prtsc

C>cd \con-dos

C>dir

C) Directory of Drive C (DOS Media)

Su C:\.          : \.          : STARTUP1 BAT : MENU      DAT : PREPMENU
C) C: FDSTART  1  : HDINSTAL BAT : AUTOEXEC  NEM : LOADCCPM COM : LETTER
C: HDAUTO   BAT : STARTSUB  HD : START     HD1 : HDDOSSUB  BAT : HELP
C: CARDFILE DIS : CARDFILE  TRM : CARDDECK BAT : CLEAR    BAT : DRTALK
C: DRTALK   DIR : DRTALK   DEF : DRTALK   KEY : FDMENU   DAT : HDMENU
C: CONSOLE1 TXT : WINDOW1  TXT
      27 File(s) 1712128 bytes free
System files exist
C)

WINDOW CHANGE N=1 PRow=34,PCol=34          (Pressing ↑,↓,→,← places window)

```

Figure 4.9 Moving a window using the Window Manager.

By pressing the window switching keys (i.e., Ctrl-1 through Ctrl-4) you can bring each window to the foreground for viewing or entering commands. Figures 4.10 through 4.12 show the three remaining windows.

Figure 4.13 shows what happens when the Ctrl-Del key combination is pressed. This combination will switch between the full window display and a partial window, if a partial window has been set up. For example, if you start up Concurrent but don't define any windows, pressing Ctrl-Del will have no effect because the virtual window is the same size as the actual screen. If you use the Window Manager or menu to change the size of a window, Ctrl-Del will cause the window to zoom out to full screen size or back to a partial window each time it is pressed.

### A Cut-and-Paste Example

One of the most useful features of the window system is its cut-and-paste facility. At first it might seem clumsy, but unless a program is written to handle cut-and-paste (very few are), a two-step process is necessary.



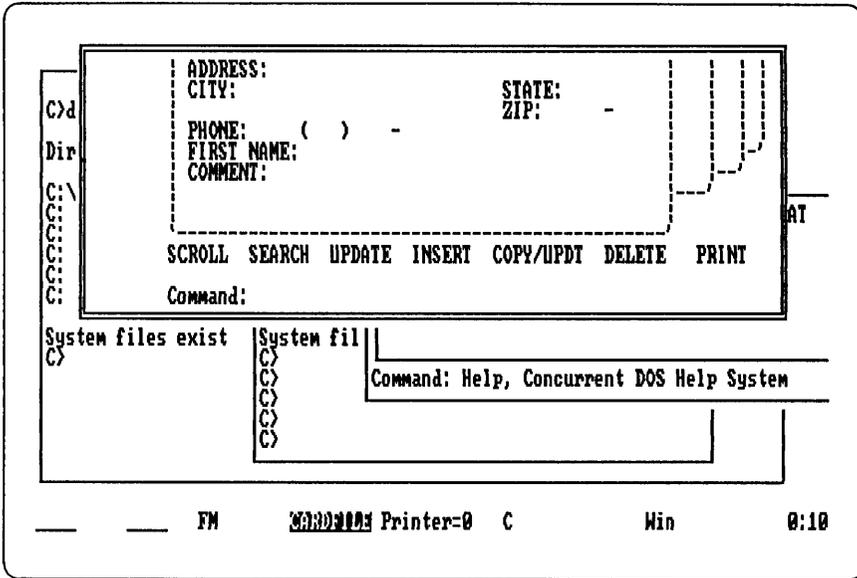


Figure 4.12 Four overlapping windows; window 4 is active.

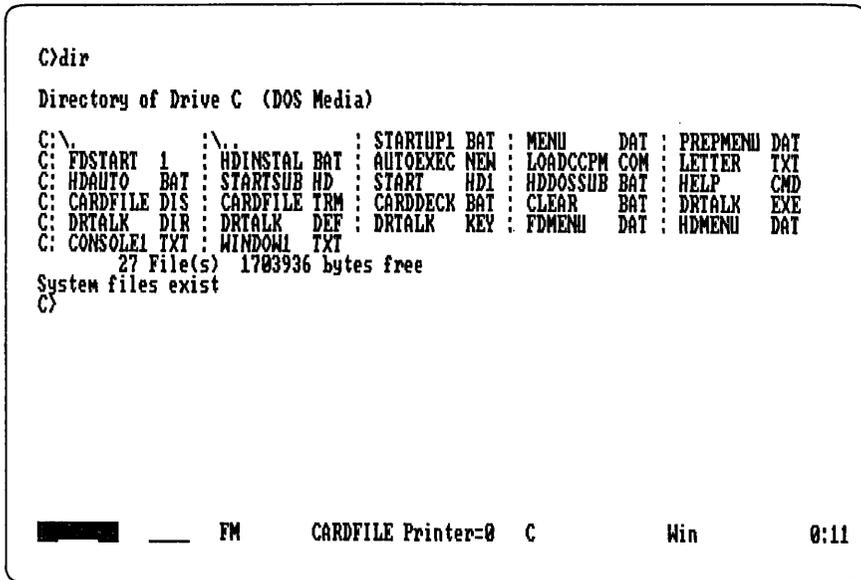


Figure 4.13 Window 1 is brought out to occupy the full screen by pressing Ctrl-Del. It can be returned to its reduced size by pressing Ctrl-Del again.

```

D12: (C0) -$C$3                                READY
C>

      B          C          D          E          F
3      $1,000,000
4          13.50%
5      $100,000
6          10
7      $100,000
8          YEAR
9          0          1          2
10
11
12 Cost of projector          ($1,000,000)
13 Investment tax credit      $100,000
14 Lease payment              $100,000          $100,000          $100,000
15 Tax rate for year          46.00%           46.00%           46.00%
16 ACRS Schedule              15%              22%              21%
17 Depreciated value          $950,000
18 Depreciation tax shield     $65,550          $96,140          $91,770
19 TOTAL                      ($734,450)       $196,140         $191,770
20 NPV                        ($734,450)       $172,811         $148,864
21
22
    _____ Printer=0 C Win 0:06
    
```

Figure 4.14 A sample Lotus 1-2-3 spreadsheet.

```

C:SAMPLE.LTR PAGE 1 LINE 8 COL 01
      <<<  M A I N  M E N U  >>>
--Cursor Movement--  -Delete-  -Miscellaneous-  -Other Menus-
^S char left ^D char right ^G char ^I Tab ^B Reform (from Main only)
^Q word left ^F word right ^DEL chr lf ^U INSERT ON/OFF ^J Help ^K Block
^E line up ^X line down ^T word rt ^L Find/Replce again ^O Quick ^P Print
--Scrolling--        ^Y line ^RETURN End paragraph ^Q Onscreen
^Z line up ^W line down ^N Insert a RETURN
^C screen up ^R screen down ^U Stop a command
L-----!-----!-----!-----!-----!-----!-----R
Dear Elaine:
      I have run a sample cash flow for the Omnitheatre project.
      Shown here is a lease/purchase plan for the projector. It
      appears that we will be able to give an investor a good rate of
      return, while keeping the lease payments within reason.
    
```

Figure 4.15 Wordstar in the second window being used to compose a letter.



```

C:\SAMPLE.LTR PAGE 1 LINE 16 COL 01
      << M A I N M E N U >>>
--Cursor Movement--  -Delete-  -Miscellaneous-  -Other Menus-
^S char left ^D char right ^G char ^I Tab ^B Reform (from Main only)
^A word left ^F word right ^DEL chr lf ^V INSERT ON/OFF ^J Help ^K Block
^E line up ^X line down ^T word rt ^L Find/Replce again ^Q Quick ^P Print
--Scrolling-- ^V line ^RETURN End paragraph ^O Onscreen
^Z line up ^W line down ^N Insert a RETURN
^C screen up ^R screen down ^U Stop a command
L-----R
      I have run a sample cash flow for the Omnitheatre project.
Shown here is a lease/purchase plan for the projector. It
appears that we will be able to give an investor a good rate of
return, while keeping the lease payments within reason.

```

Cost of projector	(\$1,000,000)		
Investment tax credit	\$100,000		
Lease payment	\$100,000	\$100,000	
Tax rate for year	46.00%	46.00%	
ACRS Schedule	15%	22%	
Depreciated value	\$950,000		
Depreciation tax shield	\$65,550	\$96,140	
TOTAL	(\$734,450)	\$196,140	
NPV	(\$734,450)	\$172,811	

Figure 4.17 Switching back to Wordstar, the WIN-DOW1.TXT file written by the Window Manager and containing the spreadsheet information is read back and attached to the letter.

## Diskettes and Fixed Disks

### DOS Media Versus CP/M Media

Diskette compatibility has long been a problem in the microcomputer world. The last diskette format that could be called a standard was the IBM 3740 single-sided single-density 8-inch floppy disk. This format gained popularity because CP/M-80 (CP/M 2.2) for the 8080 and Z80 microprocessors used it as the first interchange standard. CP/M-80 was and still is popular on the Osborne 1, the Kaypro, the S-100 bus machines, and the Apple II. Most CP/M-based systems could read and write the 8-inch format, although many computers also supported their own double-density formats that were incompatible with other computers even if they all ran CP/M.

In the 5¼-inch diskette world, two mutually incompatible diskette formats have arisen: the Apple II format and the IBM PC PC-DOS format. The original PC-DOS diskette format only worked on single-sided double-density diskettes and allowed up to 160K bytes per diskette. Double-sided disk drives were soon added to the IBM PC and the disk space went up to 320K bytes. With the introduction of PC-DOS (and MS-DOS) 2.0, the number of sectors written per track was increased by one to permit a diskette to hold either 180K or 360K bytes of information. IBM PC AT high-capacity diskettes are the latest incompatibility, at 1.2M bytes per diskette.

As mentioned earlier, PC-DOS and MS-DOS were based in concept on Digital Research's CP/M-86. The internal programmer's function calls are almost identical and the command language looks very similar, but the PC-DOS disk format for 5¼-inch diskettes is quite

different. The DOS format allows for directory entries that can keep track of the time and date a file is written, among other things. Because of these major incompatibilities it is impossible to directly read or transfer files from CP/M-86 to PC-DOS, even though many programs will run correctly on either operating system.

Concurrent PC DOS can best be described as a superset of both PC-DOS and CP/M-86 in that it can run programs from either operating system and read and write diskettes formatted for either operating system. Concurrent even tells you which disk media you are operating with when you display the directory for a disk. A CP/M-86 format diskette can be loaded into one drive and a PC-DOS diskette can be loaded into the other drive and files can be copied in either direction. Concurrent will recognize which media it is working with, write the appropriate type of directory entry, and follow all the conventions for the selected format.

### **Subdirectories and User Numbers**

Two other fundamental incompatibilities do exist between the CP/M-86 and PC-DOS media. Starting with PC-DOS 2.0, subdirectories were supported to make information storage and organization easier on a fixed disk. CP/M-86, on the other hand, uses directory user numbers which are similar in concept to subdirectories, but more limited.

Figure 5.1 shows the two different directory formats. Subdirectories may be created several levels deep and have names assigned to them. If necessary, you can create a hierarchy, or tree structure, of directories by breaking down groups of files into categories. This way they are easier to keep track of on a fixed disk, which may contain hundreds of files. CP/M-86 user numbers are similar to subdirectories, but only one level can be allocated. Each user directory has a number associated with it. Because of the difference in disk formats, subdirectory path names used by PC-DOS (i.e., \correspondence\proposals) will not work on CP/M media, and user numbers will not work on PC-DOS media. Also, you must specially format a CP/M-86 disk to use time and date stamping on files, whereas PC-DOS does this automatically.

One advantage of CP/M media is password protection, which allows you to assign passwords to files to prevent other users from reading or writing them.

### **Logical and Physical Disk Drives**

In PC-DOS, you can normally have up to four disk drives, and they may be for either floppy or fixed disks. They are labeled A:, B:, C:, and

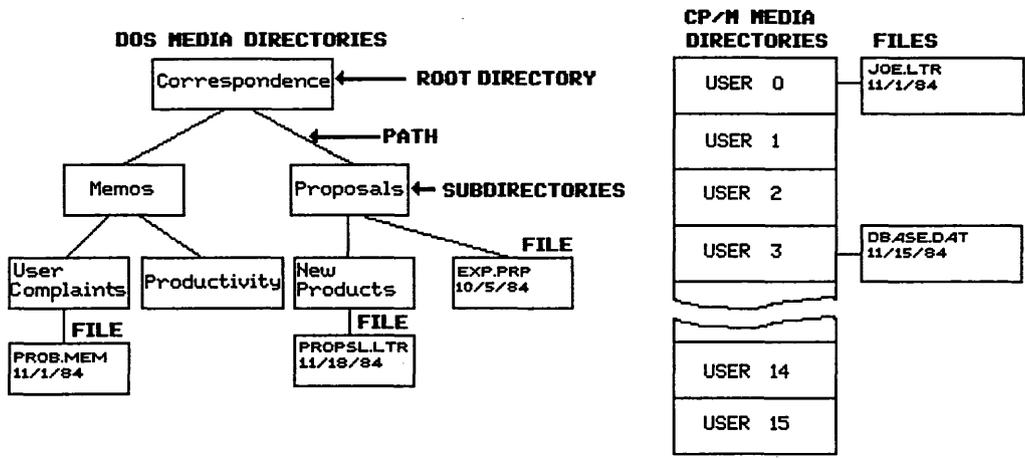


Figure 5.1 Subdirectories versus user numbers. PC-DOS and MS-DOS use a technique of splitting the directory into a hierarchy of small manageable directories. This is needed primarily on fixed disks, where the number of files stored could rapidly become unmanageable. The primary or root directory can contain files and other directories. Each of the other directories can themselves have files or directories. CP/M disks do not support this structure, but offer an alternative strategy: user areas. The directory is broken into 16 independent directories per disk, each of which may contain files.

D:. If you load DOS from a floppy, the A> prompt will come up, indicating that the currently selected drive is the A: floppy disk drive. In a fixed disk system, the C> prompt will appear to indicate that you are currently logged into the fixed disk.

Concurrent PC DOS supports this method of referring to disk drives, but provides some additional features that simplify the use of your personal computer.

If you are using a PC that has a fixed disk, the C: drive will be assigned to the DOS-formatted partition (if one exists) and the D: drive will be assigned to the CP/M-86-formatted partition or section of the fixed disk. This means that the fixed disk will be separated into two "logical" disk drives (both contained in one "physical" disk drive), each of which is formatted for a different medium.

As shown in Figure 5.2, many possible arrangements could be set up, depending on the number of floppy and fixed disks in your system. Note that unlike PC-DOS, if a second floppy disk is not installed,

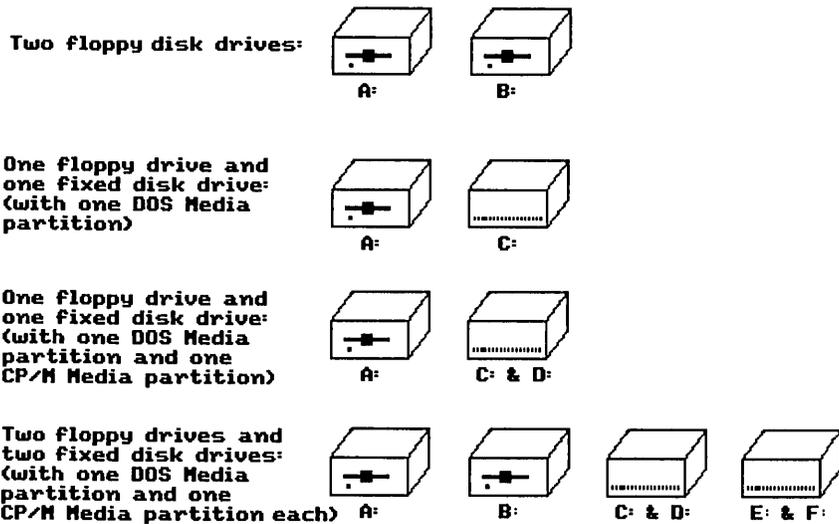


Figure 5.2 Drive designations for different hardware configurations. Concurrent PC DOS assigns drive letters for floppy and fixed disks depending on the number of drives in the system. As shown, a two floppy drive system would have the first drive labeled A: and the second B:. Since A: and B: are reserved for floppy drives, a system with one floppy and one fixed disk would be configured with drives A: and C:. Fixed disks with more than one partition would have each partition labeled as a separate logical disk drive.

Concurrent will not create a phantom drive for the B: designation. An attempt to reference the B: drive will result in a soft error and an error message. The A: drive will not be used in place of the B: drive with the message "insert B: disk."

### **Floating Drives**

In addition to the above possibilities, Concurrent supports two additional logical drives that are called N: and O: respectively. These floating drives may be assigned to any other normal disk drive or partition such as A: or C:, or, in the case of PC-DOS media, they may be assigned to a subdirectory such as `\inventory\payable`. In this case, every reference to drive N: would actually cause Concurrent to search the `\inventory\payable` subdirectory for files. This feature can be extremely helpful when you are working with programs that don't support subdirectory path names.

### **The System Disk**

PC-DOS supports a feature called PATH, which can be used to make the operating system search another path or set of subdirectories for a file if the file isn't found in the currently selected directory. In Concurrent, the SYSDISK command allows you to assign a drive, floating drive, or subdirectory path as the "system disk" where Concurrent will look for files if they aren't found in the current drive.

### **Memory Disk**

A memory disk can be thought of as a disk that is stored in the main memory, or the random-access memory (RAM), of your personal computer. The primary purpose of a memory disk is to load frequently run programs at internal memory speeds instead of slow floppy disk or fixed disk speeds. Concurrent has built-in support for a memory disk which will be created and formatted each time you power up and load Concurrent PC DOS (see Figure 5.3). Since the memory disk is kept in the main memory, it will be erased when the power is turned off or when you perform a system reset; therefore, it isn't advisable to store important files or the work in progress on it. The best use for it would be to quickly load system tools, such as a spreadsheet program or a word processor, or program development tools. Since you generally keep a copy stored on a diskette or fixed disk, it becomes a simple process to load the file or program into the memory disk at the start of each session and then execute it from the memory disk to speed loading time.

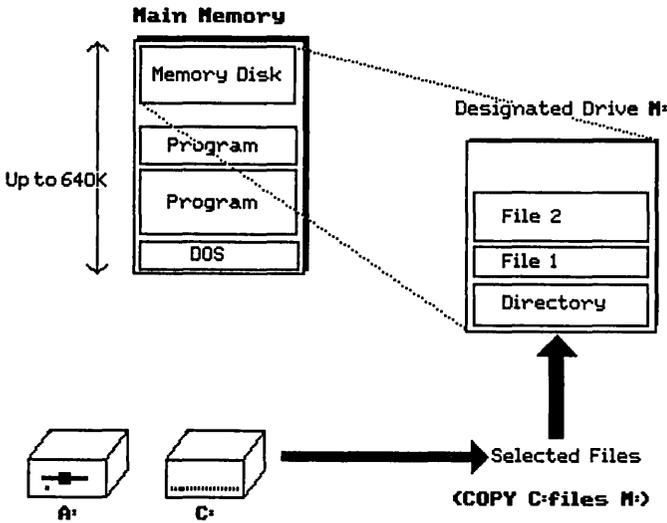


Figure 5.3 The memory disk. It is possible in PC-DOS and Concurrent PC DOS to create a memory disk. Part of the main random-access memory is allocated as a pseudo-disk that acts just like a floppy or fixed disk but works about ten times faster.

# 6

## The File Manager

The File Manager is one of the most powerful utilities provided with Concurrent PC DOS. It puts all the resources and commands available in Concurrent in menu form so you don't have to remember the syntax or options for each command. With the File Manager, you are presented with the file directory for the currently selected disk drive or subdirectory and a list of commands such as COPY FILE and RUN PROGRAM. The File Manager can be run by typing "FM" at the command prompt. Figure 6.1 shows the main menu of the File Manager.

If you are using a monochrome display, you will see the menu as it is shown here. If you have a color display, each of the regions of the menu will show up in colors.

As you can see, the main menu is broken into three major sections, or panels. The largest is the Object Panel, which contains a list of the objects that can be manipulated with the File Manager. Objects include files, disk drives, and subdirectories. The right side, known as the Command Panel, is a list of all the commands that can manipulate objects such as files. The bottom section is referred to as the Prompt Panel. It contains any prompts that may be required by commands as they are selected and shows the commands available through the use of the function keys F1 through F10.

The File Manager can greatly simplify the management of resources on a personal computer by allowing you to see the available commands and existing files all at once on the screen. It supports the file backup and restore facilities and allows you to configure your system or change its configuration at any time. The File Manager can format diskettes and create new system disks as well.

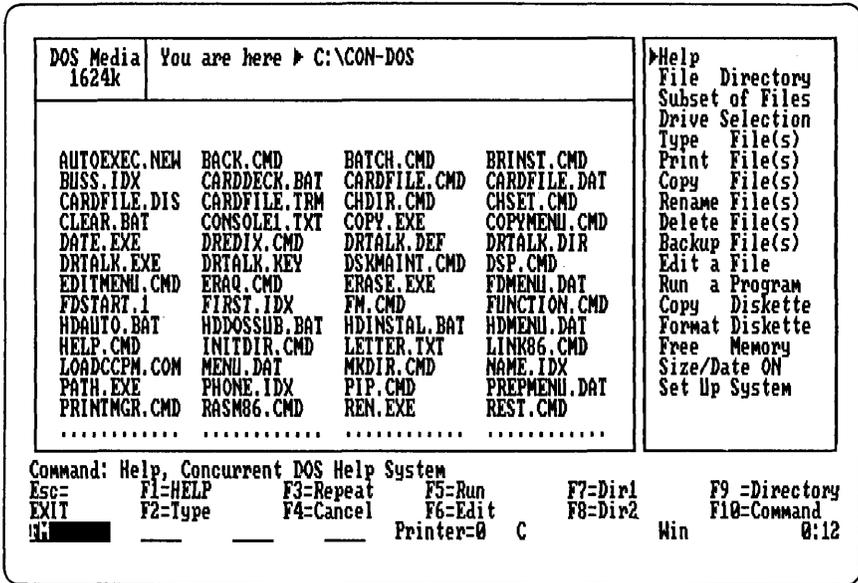


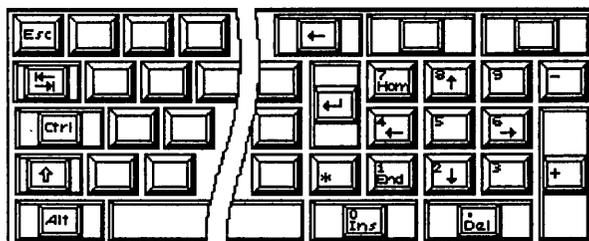
Figure 6.1 The File Manager main menu. Function keys are shown at the bottom of the screen.

### The Special Function Keys

Most of the function keys used by the File Manager are listed at the bottom of the screen; however, some additional function keys are used including the Tab key, the arrow keys, Insert, Delete, Home, Page Up, Page Down, and End. Figure 6.2 explains their use.

### Selecting Groups of Files

It is possible to copy, print, delete, backup, or rename multiple files at one time by selecting a group of files before executing the appropriate command. This means that you can move to the object panel, select 10 files, and issue either a COPY, PRINT, DELETE, TYPE, BACKUP, or RENAME command to affect all 10 files. Using the Insert and Delete function keys, you can select or deselect files in the object panel. Pressing Insert when pointing to a file will select it, while pressing Delete will deselect the file (see Figure 6.3). When you have selected the desired group of files, you can use the Tab key to return to the command panel and then select the COPY FILE(s) command. The



-  Stops any operation in progress and returns to the last menu.
  -  Moves the cursor between the object and command panels.
  -  In the command panel, the cursor is moved to the next command. In the object panel, the cursor is moved to the right. If this isn't possible, it is moved to the first item in the next row.
  -  Selects the object at the cursor and highlights it.
  -  Cancels selection of the object at the cursor and removes the highlighting.
  -  Selects the object at the cursor and/or executes the command for the currently selected objects.
  -  Moves the cursor to the top of the list (command or object panel).
  -  Moves the cursor to the bottom of the list (command or object panel).
  -  Moves cursor to top left in list (command or object panel).
  -  Moves cursor to bottom right in list (command or object panel).
- 



- The arrow keys control cursor movement within the command and object panels.

Figure 6.2 File Manager cursor movement and control keys.

secondary copy menu will request a destination disk drive (see Figure 6.4). After pressing Enter, the files will be copied automatically.

Another method for selecting commands involves typing in the first one, two, or three letters of the command. As you press each key, the command that most nearly matches the characters you type will be highlighted. For example, if you wish to enter the COPY command,

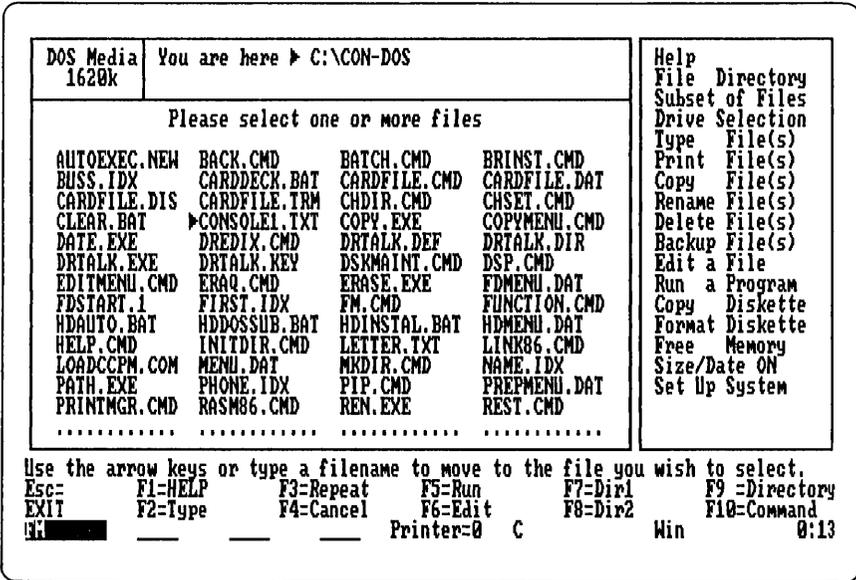


Figure 6.3 Selecting files with the Ins and Del keys.

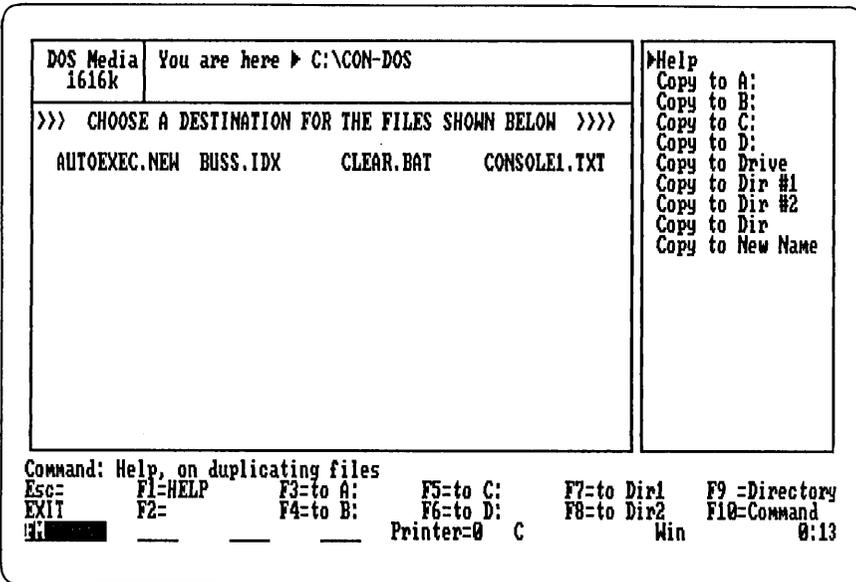


Figure 6.4 Copying files that have been selected with the Ins key.

you would merely have to type "C". The cursor would then point to it, and by pressing Enter you would invoke it. If the command you want starts with the same character as another (e.g., SETUP and SELECT), you will have to enter as many characters as it takes to define the command uniquely. To choose the SELECT command you would have to enter SEL to distinguish it from SETUP.

### The File Directory

The File Directory option in the main File Manager menu allows you to change the current directory (the one displayed in the object panel and to which all commands automatically refer if an alternate directory or disk isn't specified). As shown in Figure 6.5, this submenu contains seven options: Change Directory, Help, View Files, Set Directory 1, Set Directory 2, Make Directory, and Remove Directory. This subdirectory is reached by positioning the cursor next to the file directory entry of the main menu and pressing Enter (see Figure 6.6). If the currently selected disk is formatted as DOS media, the object panel will show the available subdirectories, including the root [which is represented as a backslash (\) just as in PC-DOS]. The currently

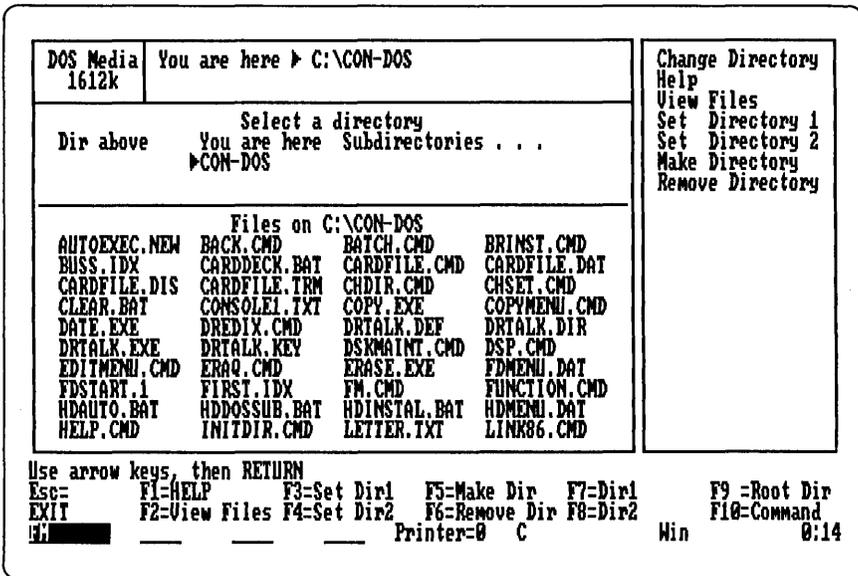


Figure 6.5 The File Directory submenu.

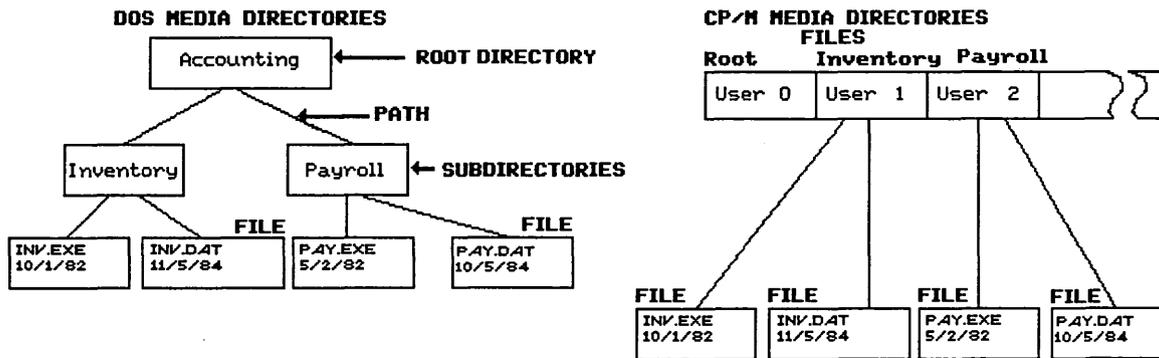


Figure 6.6 Subdirectories and user numbers. Different file storage strategies are required when working with DOS and CP/M media.

selected directory is shown in reverse video (i.e., the foreground and background colors are reversed). You can use the arrow keys to select the new current directory, and then press Enter. If you are already in a subdirectory, your display will only show the current subdirectory and the root, so you must first press the F9 function key to move to the root. The rest of the first level of subdirectories will then be visible. If the currently selected disk drive is formatted as CP/M media, then invoking the file directory command will cause a list of the 16 user numbers (root plus 1st through 15th) to be displayed (see Figure 6.7). As discussed in Chapter 5, user numbers can be thought of as predefined subdirectories one level deep. If the CP/M disk has a USER.TXT file that contains the names of the 16 directories, these names will be displayed in the File Manager menu instead of the numbers 1 through 15.

### The Subset of Files Command

When a directory is displayed in the object panel, all files are listed. For many applications it may be desirable to limit the display to a group of files, perhaps based on the file type extension of the file name. This can be done with the SUBSET command. When invoked, the

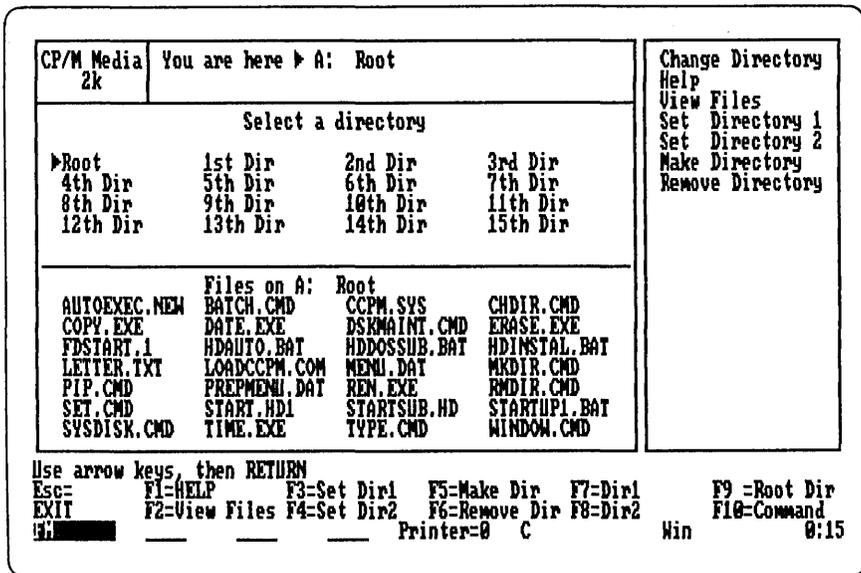


Figure 6.7 CP/M-media File Directory submenu.

following prompt will appear in the panel:

Enter wildcard (? matches any character, \*.\* matches all):

You can choose the group of files you want by the use of the ? and \* characters, just as you would with the DIR or COPY commands in CP/M-86 or PC-DOS. For example, if you want to display all files that start with the letters "PC" you could enter the wildcard "PC\*.\*". Similarly, "\*.DOC" would list all files with the file type .DOC.

The subset selection will be retained for each disk drive separately. The wildcard \*.DOC will not be carried along if you select it for drive A: and then switch to drive C:. If you then switch back to drive A:, the \*.DOC wildcard name will still be in effect for the directory display.

### The Drive Selection Menu

The current disk drive can be selected through the menu in a manner that is analogous to the selection of the file directory. When invoked, this menu option displays a list of the available disk drives and allows you to choose one by using the arrow keys or the space bar (see Figure 6.8). After pressing Enter, the object panel is updated to show the new

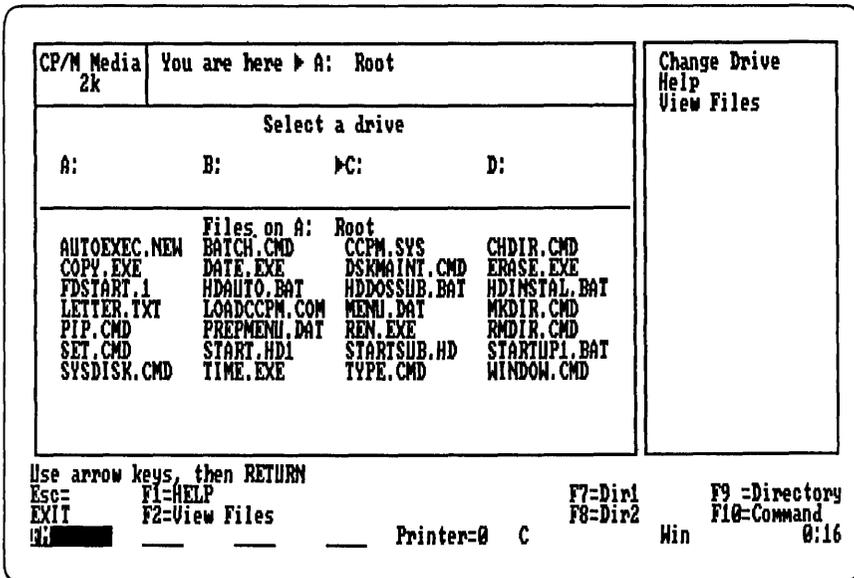


Figure 6.8 The Drive Selection submenu.

directory, with wildcard and path or user number in effect if one was previously selected for that drive. For example, if you select the \inventory subdirectory on the C: drive and switch to the A: drive, the \inventory subdirectory will still be selected when you return to the C: drive.

The Drive Selection menu also includes the option of viewing the disk directory of a drive before selecting the drive. Function key F2 invokes this feature. As with most File Manager commands, context-sensitive help is always available.

### Manipulating Files

The File Manager supports all the traditional PC-DOS and CP/M-86 commands such as TYPE, PRINT, COPY, RENAME, and DELETE, and it adds a new built-in feature: BACKUP. All of these commands allow direct manipulation of files without having to remember the command's format and options.

When any of these commands are invoked, they will prompt you to select one or more files on which to operate. You may use the arrow keys, the space bar, and the Insert and Delete keys to select as many files as you want. You may also select files before invoking the command by pressing the Tab key to move to the object panel first. Regardless of how you select files, each command will ask you for all required information, such as the destination disk (in the case of the COPY command) or the new file name (in the case of the RENAME command).

### Context-Sensitive Help

The HELP facility is more than just a list of commands that you can refer to when you can't remember the syntax or the options in the parameter list of a command like WINDOW. HELP is used by the many menus in Concurrent and by the File Manager to provide help that is selective to the command you are trying to use at the moment that you press the F1 or designated help key. If you are trying to use the File Manager to copy a file but can't remember how to select multiple files, pressing the F1 key will give you specific information related to what you have done already and what is currently on the screen. This facility effectively eliminates the need for a user's guide. PC-DOS and MS-DOS have no comparable facility.

Figure 6.9 shows the screen that is displayed if you press F1 while viewing the File Manager main menu, and if the cursor is in the object panel. This list of function keys is all you really need to successfully

```

HELP

Syntax:
HELP {topic} {subtopic1 ... subtopic8} {[NOPAGE|LIST]}

Explanation:
HELP displays a list of topics and provides summarized information about
Concurrent commands.
HELP topic displays information about that topic.
HELP topic subtopic displays information about that subtopic.
From a main topic's display, you must type a period before a subtopic name
to view a subtopic.
information for your topic, it displays the special prompt HELP> on your
screen, followed by a list of subtopics.

- Enter ? to display list of main topics.
- Enter a period and subtopic name to access subtopics.
- Enter a period to redisplay what you just read.
- Press the Enter key to return to the Concurrent system prompt.
- [NOPAGE] does not stop at 24 lines; window display keep

HELP> _____ Printer=0 C 0:11

```

Figure 6.9 The Help facility.

navigate the File Manager. Pressing Enter will cause HELP to show you the next page of information, as shown in Figure 6.10.

To return to the original File Manager display, press the Escape key (ESC). Each command on the main menu has a separate help section that can be referred to by moving the cursor to the desired menu entry and pressing the F1 key. Figure 6.11 shows the display for the drive selection help entry.

The first menu entry, Help, will take you to the title screen of the help facility where you can select your own topic from a list. One useful help entry is HOWTO?. The first page of HOWTO? is shown in Figure 6.12. It provides general information on how to find your way around the system and which commands and utilities should be used to perform a desired service. If you don't know where to look, consult this useful index. Generally speaking, all functions listed in HOWTO? are available through the File Manager menus and submenus.

## The TYPE and PRINT Commands

TYPE and PRINT are used to list text files on either the current window or the printer. Both are selected by moving the cursor to the desired command and pressing Enter. If a file is not selected prior to

```

HELP

Syntax:
HELP {topic} {subtopic1 ... subtopic8} {[NOPAGE!LIST]}

Explanation:
HELP displays a list of topics and provides summarized information about
Concurrent commands.
HELP topic displays information about that topic.
HELP topic subtopic displays information about that subtopic.
From a main topic's display, you must type a period before a subtopic name
to view a subtopic.
One or two letters is enough to identify the topics. After HELP displays
information for your topic, it displays the special prompt HELP> on your
screen, followed by a list of subtopics.

- Enter ? to display list of main topics.
- Enter a period and subtopic name to access subtopics.
- Enter a period to redisplay what you just read.
- Press the Enter key to return to the Concurrent system prompt.
- [NOPAGE] does not stop at 24 lines; window display keeps scrolling.
- [LIST] removes extra lines between headings; use with Ctrl-P to print.
Press RETURN to continue
HELP> _____ Printer=0 C                                0:11

```

Figure 6.10 The second page of the Help facility.

```

FILEMANAGER
DRIVE

The DRIVE SELECTION Menu displays all the drives present on the Concurrent
system. This menu allows you to select a different drive from the one on
the Main Menu. After the DRIVE SELECTION command is selected, a new menu
appears. Move the cursor around the top of the object panel under Select a
Drive to choose a new drive. Press Enter when the cursor points to the file
directory you want to select.

The following table summarizes the commands for the DRIVE SELECTION Menu.

Command          Action
CHANGE DRIVE     Allows you to select another drive.
HELP             Offers help on DRIVE SELECTION.
VIEW FILES       Allows you to display the directories in a drive without
                 selecting it. It works the same way as the F2 key.

The following table lists the function key assignments for the DRIVE
SELECTION Menu.

Function Key     Action
Press RETURN to continue
HELP> _____ Printer=0 C                                0:12

```

Figure 6.11 Context-sensitive help; the Help message for drive selection.

```

HOWTO?

For help on performing the following tasks, consult the listed HELP topic:

TASK                                     HELP TOPIC
Back up hard disk files                 BACKREST
Boot automatically                     HDMAINT
Boot from hard disk                    HDMAINT
Change user numbers                     USER
Check on free space on a disk           SHOW
Copy diskettes                          DSRMAINT
Copy files                              PIP, COPY
Create a memory drive                   SETUP, MEMORYDISK
Define function keys                    FUNCTION
Delete a file                           ERASE, ERAQ
Execute a series of commands            BATCH
Format diskettes                       DSRMAINT
Initiate time stamping                  INITDIR
Prepare a hard disk                     HDMAINT
Print files                             FILEMANAGER PRINT, PRINTMGR, PIP
Read a file                             TYPE
Rename a file                           REN
See a directory of files on a disk      DIR, SDIR
Press RETURN to continue
IBM: _____ Printer=0 C                                0:13

```

Figure 6.12 The Help facility showing the HOWTO? option. HOWTO? is an index to features and facilities and directs you to the appropriate Help section.

invoking the command, the File Manager will prompt you for a file name selected from the object panel directory display, as shown in Figure 6.13. Once given a file name, TYPE will open the file and display it in blocks in the object panel part of the screen (see Figure 6.14). If a PRINT command is given, the file will be sent to the printer shown in the status line at the bottom of the screen.

### The COPY Command

The COPY command performs the function of the familiar PC-DOS COPY command, but in Concurrent you can select one or more files from the directory list and specify their destination from a submenu of destinations. If files aren't selected before invoking the COPY command, the screen will look like the one in Figure 6.15.

After files have been selected using the Insert or Enter key (you must press the Enter key to select the last file), the File Manager will display a list of possible destinations (see Figure 6.16).

Note that the files you selected are displayed in the object panel. If the list is incorrect, you can press the ESC key and return to the main

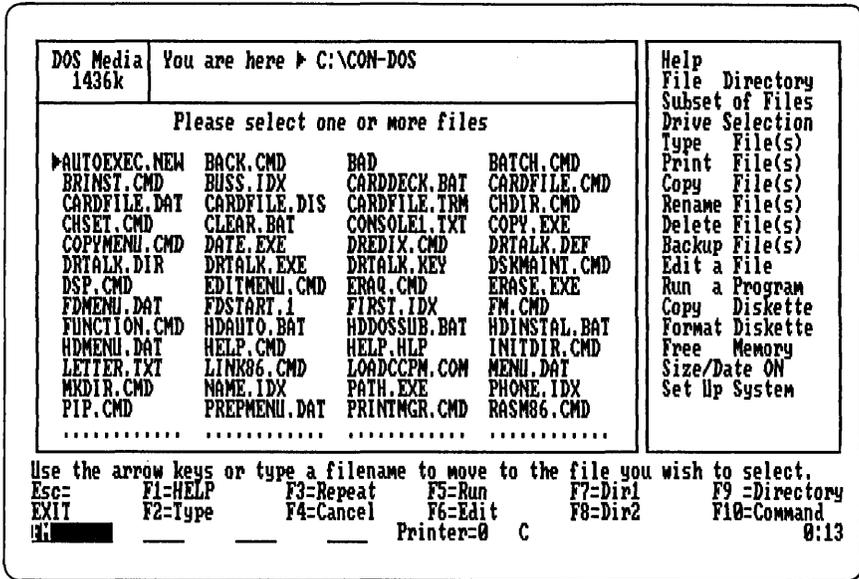


Figure 6.13 The TYPE command. Filename prompt.

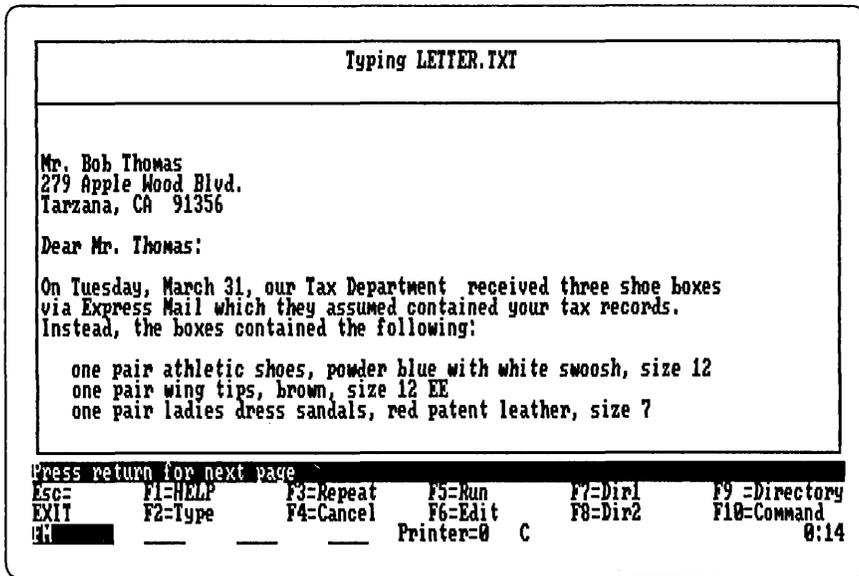


Figure 6.14 The TYPE command. Formatted output display.



menu. The selected files will again be shown in the object panel and you can invoke the COPY command to select or deselect other files.

Returning to the destination selection menu, you can choose any of the disk drive destinations by pointing to the desired line and pressing Enter. The first four lines reference the drives A through D. The COPY TO DRIVE command will cause the list of available drives to be displayed in the object panel where you can select the destination with the arrow keys. Dir#1 (N:) and Dir#2 (O:) may be selected, or a subdirectory within a PC-DOS media disk may be chosen with the COPY TO DIR command. COPY TO NEW NAME allows you to copy and rename a file or files in one operation. The File Manager will request the new name of each file that is copied.

As discussed earlier, it is possible to select all the files for a copy operation prior to invoking the COPY command. Whether files are selected before or after COPY is invoked is simply a matter of personal preference. Figure 6.17 shows files being selected from the object panel. Pressing the Tab key will move the cursor from the command panel to the object panel. The arrow keys can then be used to point at a file, and pressing Insert will select the file. In addition, files can be selected by typing in the file name or a name with \* and ? wild cards.

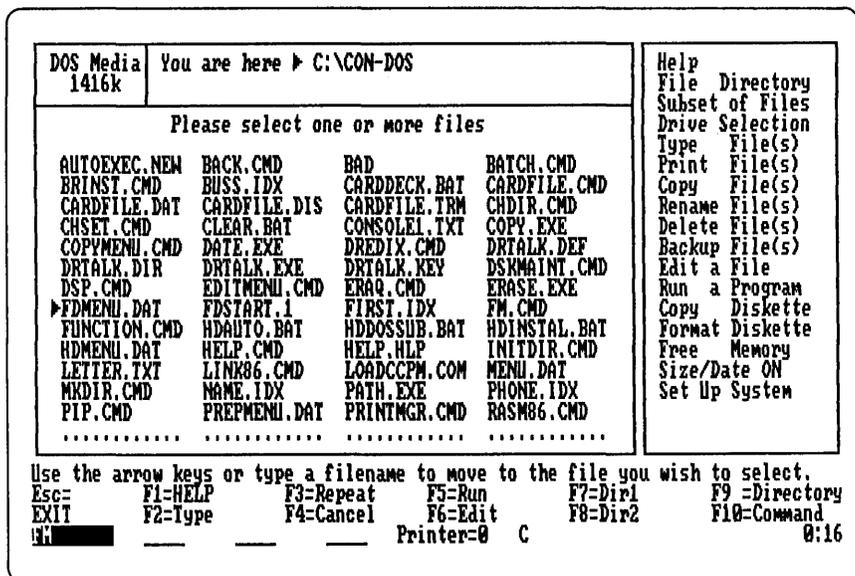


Figure 6.17 Selecting files in the File Manager.

All matching files will be selected when you press either the Insert or Enter key. When finished, pressing the Tab key again will move the cursor to the command panel where the COPY (or other) command can be invoked.

### The FREE MEMORY Command

The FREE MEMORY command displays a list of all the programs currently running, how much memory is allocated to them, and the amount of free memory left in the system. After displaying the active programs, the program will ask you to either press Enter or type the number of the program you wish to stop. Figure 6.18 shows a typical free memory display. It is sometimes necessary to stop or abort a program if it is unresponsive or doing something unexpected. Since there may be other work in progress or other users on the system, it is not advisable to use the normal system reset (Ctrl-Alt-Del). The FREE MEMORY command allows you to restart a window without disturbing other programs. Note that aborting a program in this way will cause any output files being written by the program to be lost, but input files will not be disturbed. This command is also available from the normal DOS command prompt C> as the STOP command.

```

Entry  Program  Memory  Window
Number Name    Used    Number
-----
**    STOP      16K     1
1.    FM         122K    1
2.    PRTSC      133K    1

Contiguous Blocks of Free Memory
4C00:0      192K

Free Shared Code == 0K
Total Free Memory == 192K Bytes

FH _____ Printer=0 C 0:16

```

Figure 6.18 The Free Memory display.

### The SETUP SYSTEM Command

The File Manager allows you to perform all system setup functions, including formatting disks and loading the operating system onto disks. The Setup System menu allows you to assign function keys, set attributes for the serial ports, set up a memory disk, and set all other default system attributes (see (Figure 6.19). These submenus are also available at the C> system prompt by entering the SETUP command.

#### COPY TO SYS DISK

The first command in the Setup System submenu, COPY TO SYS DISK, will copy selected files to the defined system disk to make them more accessible to programs in all four windows. Since the system disk is searched by the operating system for files that can't be found in the current directory and/or disk, program files placed in the system disk will be readily available no matter what your current disk or directory is. Files that are copied with this command are also assigned the "System" and "Read/Only" attributes.

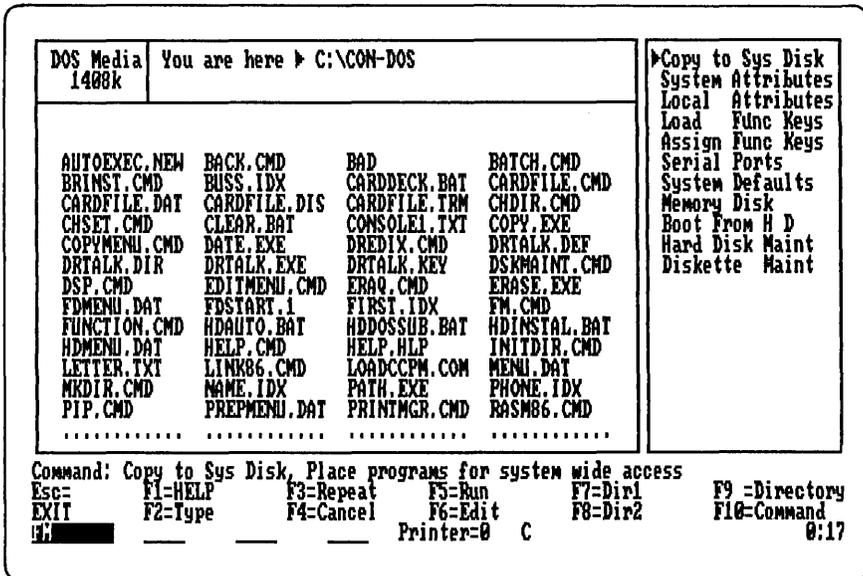


Figure 6.19 The SETUP command. This feature allows you to configure and customize the Concurrent operating system.

## System/Local Attributes

The SYSTEM ATTRIBUTES and LOCAL ATTRIBUTES commands give selected files either the "system" or the "local" attribute. The system attribute allows programs or files to be used simultaneously by multiple windows, without regard to the directory or disk currently in use. The local attribute is generally given to all other local or nonshared files, but system attribute files must be changed to local in order to rename or delete them. Files selected when either of these commands is invoked will receive the new attribute.

## Assign Function Key Submenu

The definitions of the function keys, F1 through F10; Alt-F1 through Alt-F10; Ctrl-F1 through Ctrl-F10; Shift-F1 through Shift-F10, and the arrow keys, Home, PgUp, PgDn, End, Ins, and Del can be easily modified with the Assign Function Key submenu.

As you can see in Figure 6.20, three options are available: Assign function keys, Save function keys, and Reset window switching keys.

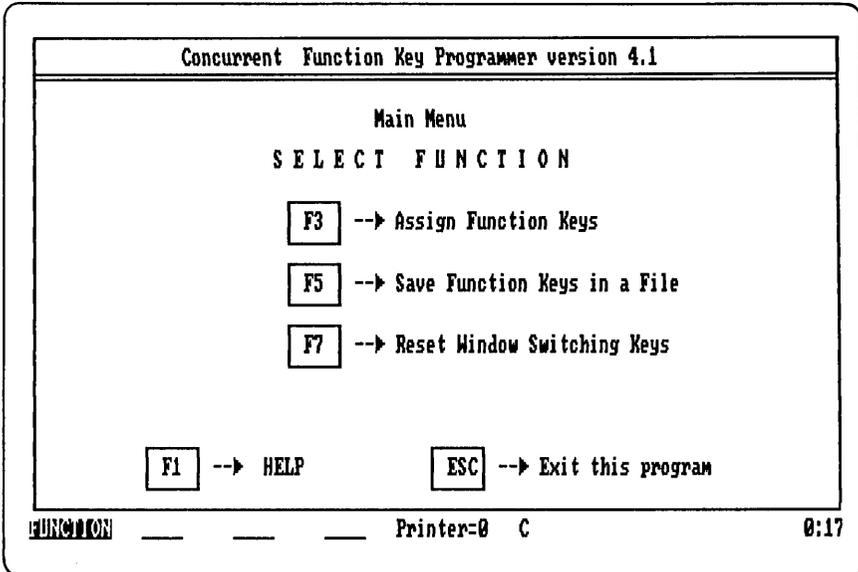


Figure 6.20 The SETUP command. Assigning function keys.

### Assigning and Saving Function Keys

To alter the function key assignments, press the F3 key; you will be presented with the first of three pages of key definitions (see Figures 6.21 through 6.23). To change one of the default entries, simply press the corresponding key on the keyboard. The entry will be highlighted and the cursor will be positioned for your input. You may enter any ASCII character by typing it on the keyboard. To enter nonprintable characters, such as the delete character (which is 7F in hexadecimal or 127 in decimal), you can press the backslash (\) character followed by the hexadecimal value of the ASCII keycode. This happens automatically when you press the Enter key (\OD) or hold down Ctrl and press any other key. The Backspace key will erase any unwanted input. To conclude the entry of a new function key definition, press the Ctrl-@ key combination.

When you have finished, you can save the redefined keys by returning to the main menu (by pressing ESC) and then pressing the F5 function key, as shown in Figure 6.24. Function keys are normally saved in a file called KEYS.PFK which will be automatically reloaded, but you may also save different sets of function keys for different applications. The Setup System menu in the File Manager has a LOAD

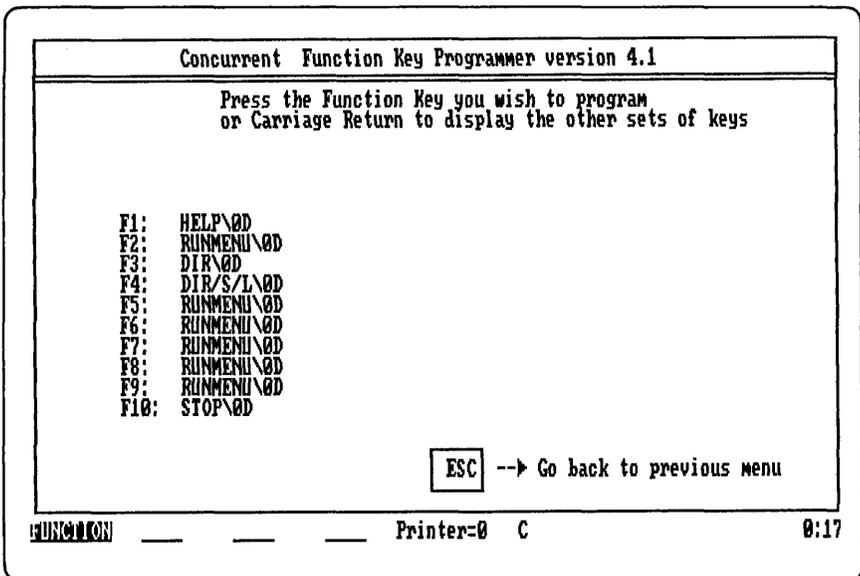


Figure 6.21 The SETUP command. Assigning functions for the unshifted F1 through F10 keys.



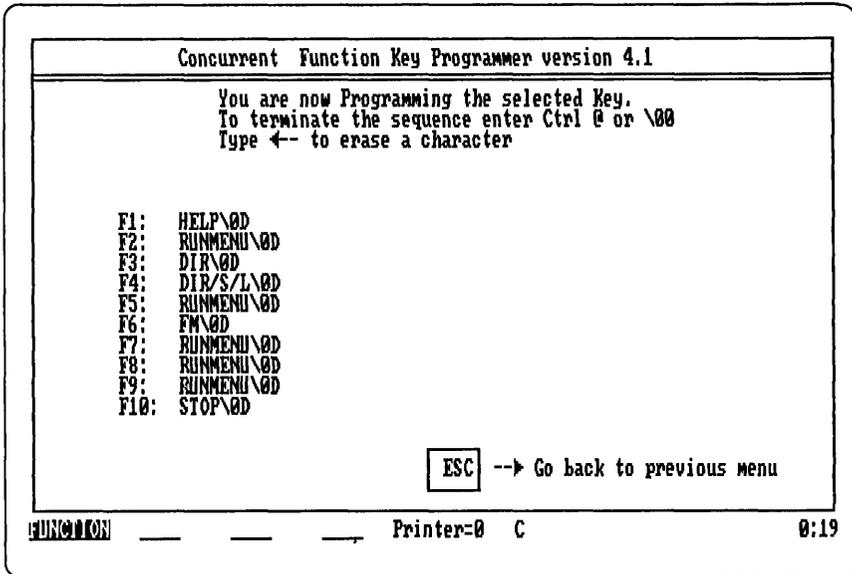


Figure 6.24 Reassigning the meaning of function key F6.

FUNCTION KEYS command which will ask for a file from the object panel.

### Resetting Window Switching Keys

The third option on the Assign Function Key submenu allows you to change the default window switching shift-key combination. Normally, the Control key in conjunction with the numeric pad keys is used to switch windows, but if a program you choose also makes use of these function keys, you must reassign them to Alt-1 through Alt-4 or some other shift-key combination.

Figure 6.25 shows all the possible combinations of the shift-keys. The choices are: keypad only (no shift key), Shift, Ctrl, Alt, Shift-Ctrl, Shift-Alt, Alt-Ctrl, and Shift-Alt-Ctrl. If you were to choose Shift-Alt, you would have to hold down the Shift and the Alt keys while pressing the appropriate numeric-pad key (1 through 4, Del, or +) in order to select a window, toggle (switch) between full or partial windows, or invoke the Window Manager.

### System Defaults

Other system defaults include: selecting the serial port usage, setting the color monitor scrolling mode, defining a memory disk, and setting

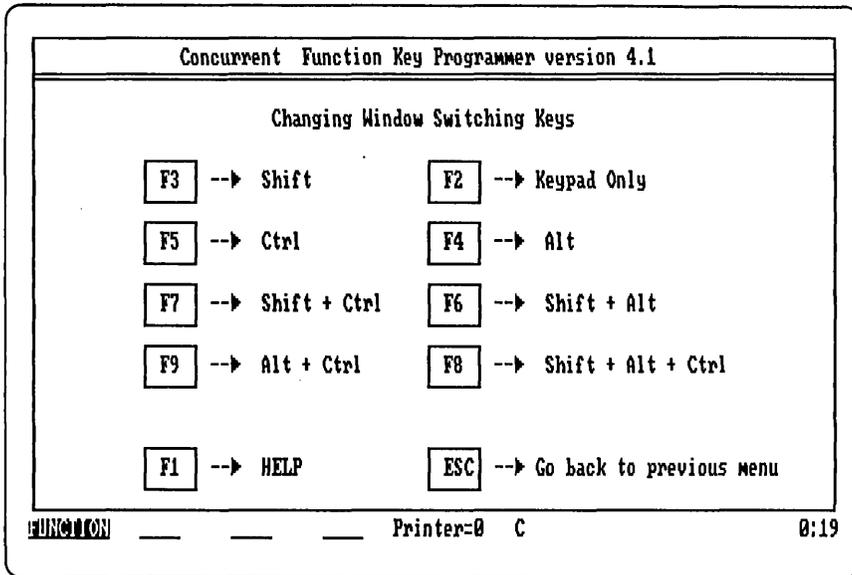


Figure 6.25 This submenu allows the window control keys to be reassigned.

the maximum memory per program. These and other options may be selected by invoking the System Defaults entry of the File Manager submenu.

Figure 6.26 shows the main menu for the SETUP command. You may assign function keys (as described above), set the diskette head step time (if your drives are faster or slower than normal), define the uses of the two serial ports (peripheral port or serial console), set the color monitor scroll mode (flicker-free or fast), set up a memory disk, set the verify-after-write option for disks, and set the maximum memory a program can allocate (to keep a program from allocating all of your main memory). Examples of the submenus are shown in Figures 6.27 through 6.33.

### The SIZE/DATE Command

The SIZE/DATE command lets you select the way the File Manager displays the disk directory in the object panel. Normally SIZE/DATE is turned off so only file names are displayed; this allows more of the directory to be shown at once. If SIZE/DATE is turned on, you will see the file name, file size in bytes or characters, and the date the file was

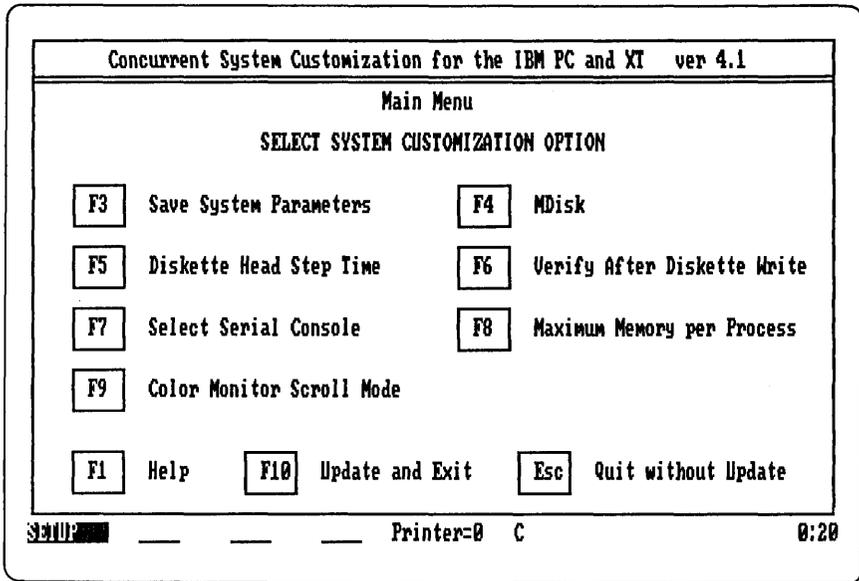


Figure 6.26 The SETUP command Main menu.

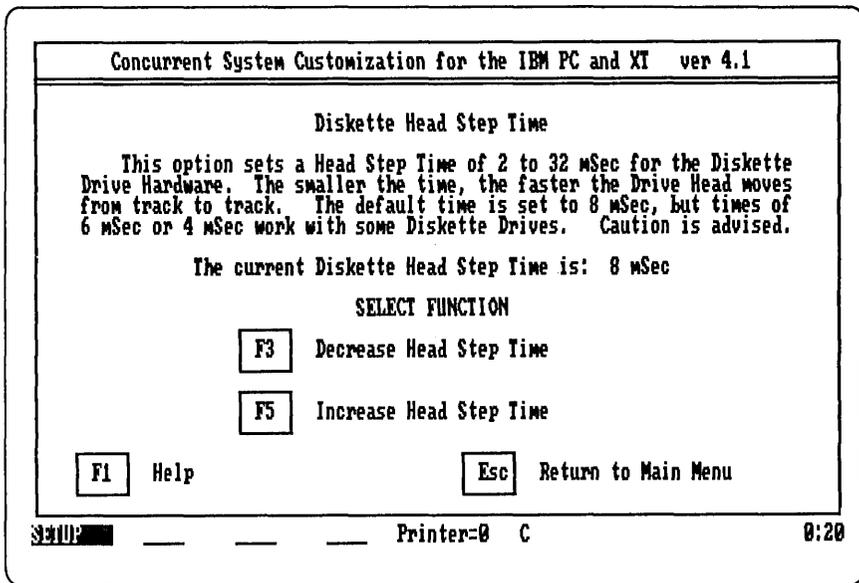


Figure 6.27 The SETUP command. Disk head Step Time menu.

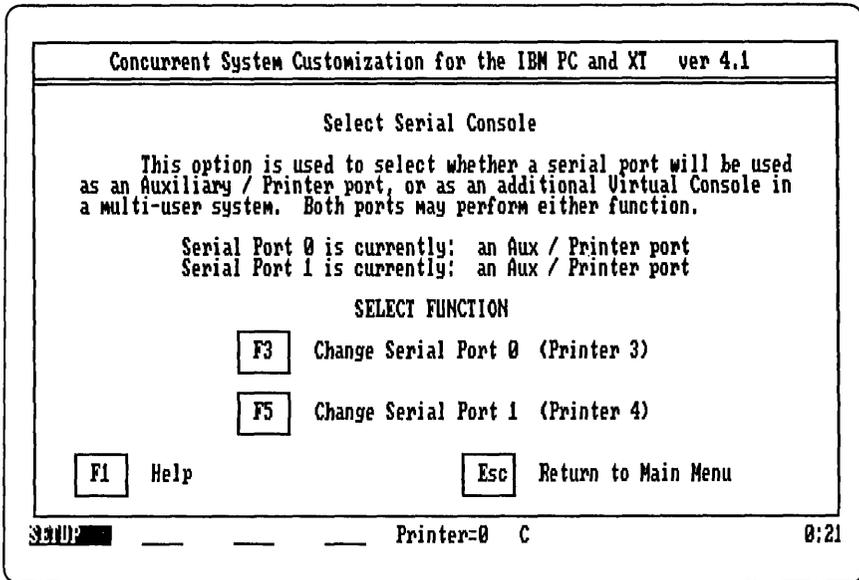


Figure 6.28 The SETUP command. Serial port defaults.

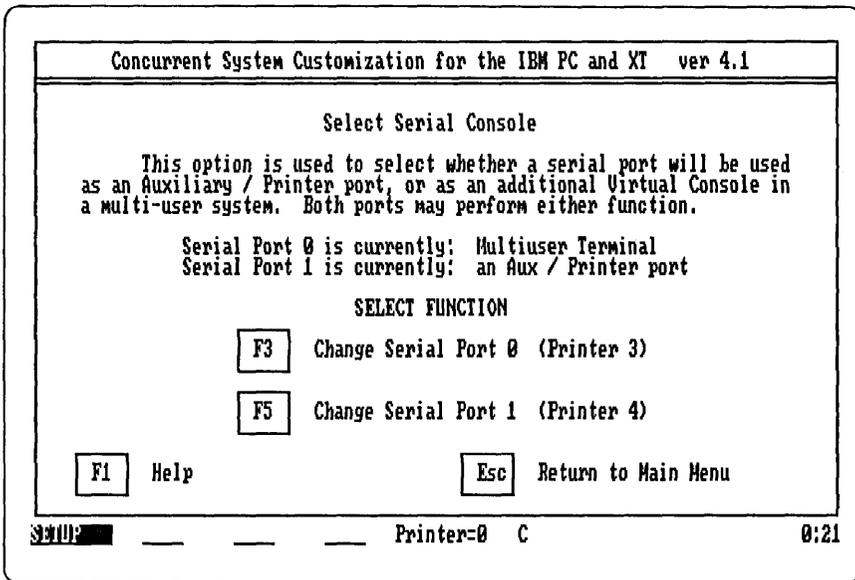


Figure 6.29 The SETUP command. Serial port reassignment.

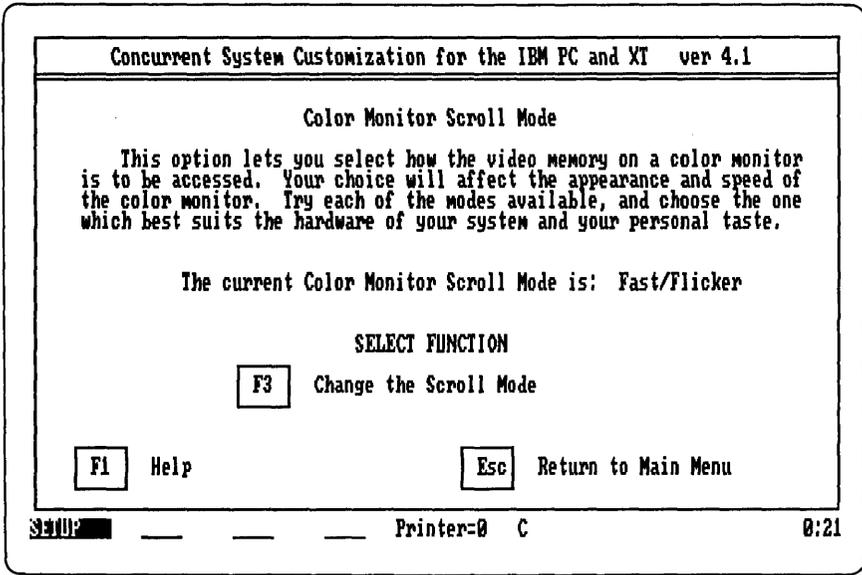


Figure 6.30 The SETUP command. Color Screen Scroll Mode menu.

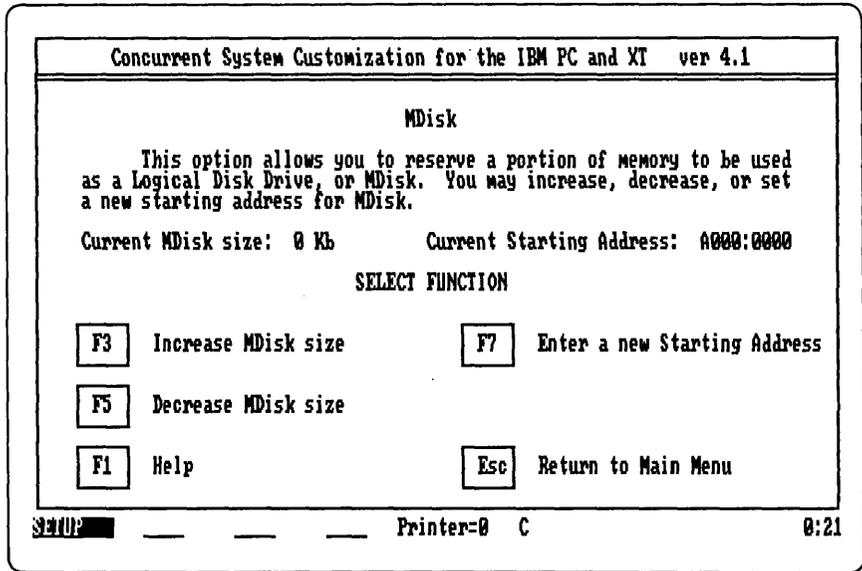


Figure 6.31 The SETUP command. The Memory Disk menu.

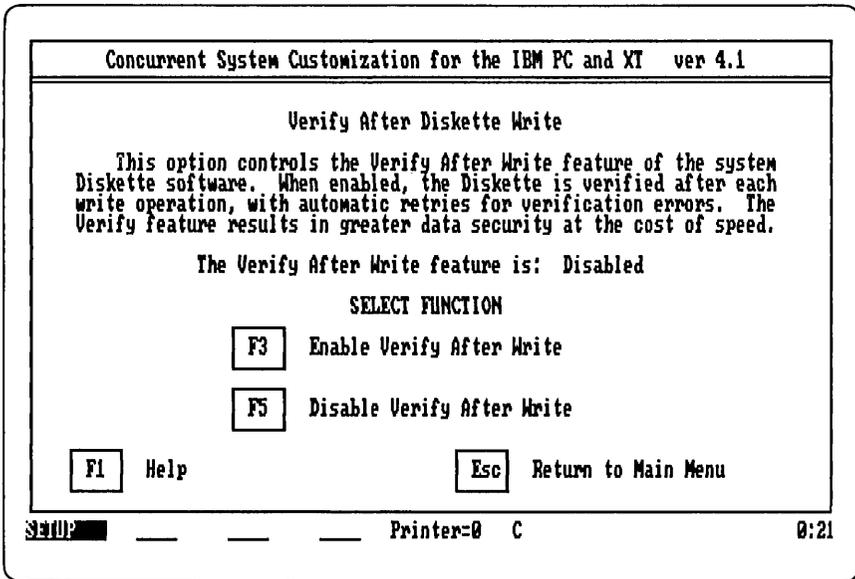


Figure 6.32 The SETUP command. Setting the disk verify-after-write status.

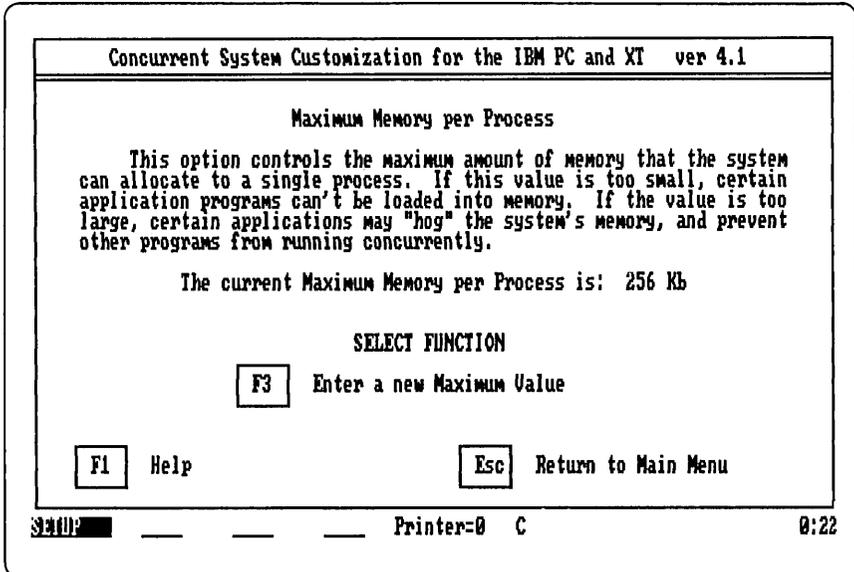


Figure 6.33 The SETUP command. Setting the maximum memory allowed per program.

created or last updated, as shown in Figure 6.34. When using CP/M media, you have the option of activating time/date stamping. On PC-DOS media, time/date stamping is always active. If the directory of a CP/M-media disk is displayed and time/date stamping is not enabled, you will only see the file name and the file size.

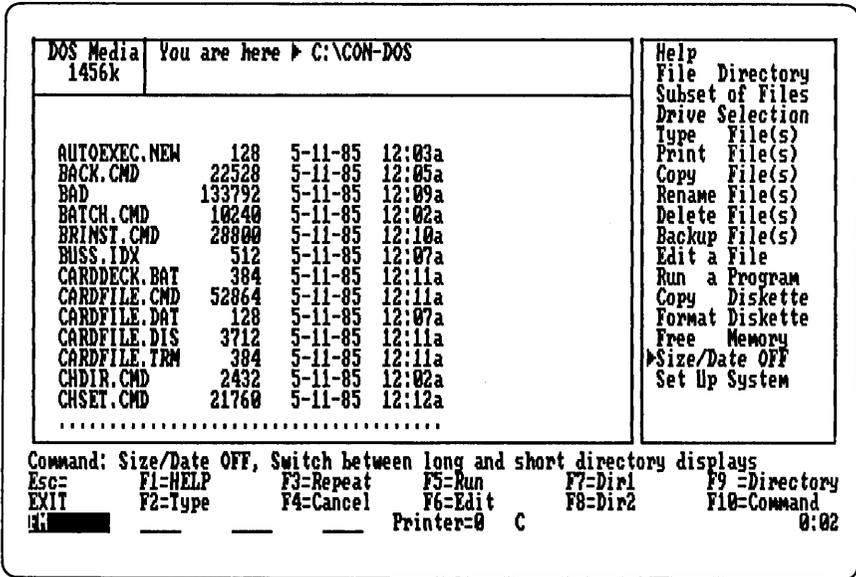


Figure 6.34 The directory display size/date option. When activated, this option shows all information regarding the files in a directory.

## Creating and Modifying Menus

### What are Menus?

In Concurrent PC DOS, menus are special control files that contain text and commands which are interpreted and displayed by a built-in program called RUNMENU. You can use the menu editor, EDITMENU, to create and modify menu screens to make a friendly “shell” or package for your application programs. This menu facility is of use to software developers, but it is also a useful and handy tool for the average user.

For example, you could create a menu that would appear when your system performs a power-on reset, or warm restart, and display a list of the programs you use most frequently. Pressing one function key would cause the desired program to be loaded with any standard options automatically set.

A programmer could create a menu that would automate the three most common functions in program development: editing, compiling, and linking a source file. Any frequently entered series of commands can be automated and assigned to one of the 10 function keys.

An office manager could set up a series of menus to simplify the training of employees. Standard office procedures, such as loading a word processor, a spreadsheet program, or the payroll program, could be set up on one main menu, thus making it easier for employees to learn the system.

### Menu File Organization

Each menu file can contain one or more menus. The file can contain individual menus or a series of linked menus. It is important not to confuse the menu file name with the names of the menus within a file. Figure 7.1 shows a sample menu file with its individual menus, and Figure 7.2 shows the logical hierarchy of these menus.

The first screen of the EDITMENU utility is shown in Figure 7.3. From this screen you can create a new menu or change, copy, or delete an existing menu.

Using the Create Menu function, you can add new menu screens to an existing menu file. Figure 7.4 shows the Menu Creation screen. If you have trouble remembering the contents of a menu file, pressing F2 will display the list of menus. As always, ESC takes you back to the previous screen.

The Menu Change screen looks just like the Menu Creation screen. You are asked for the name of an existing menu and can look at the list of menus as before.

Figure 7.5 shows the Menu Editing screen, which is quite similar in

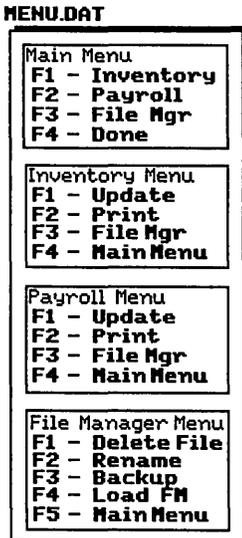


Figure 7.1 The menu file. EDITMENU allows you to set up one or more menus that contain DOS commands to execute. This feature allows complex tasks to be automated without resorting to a programming language.

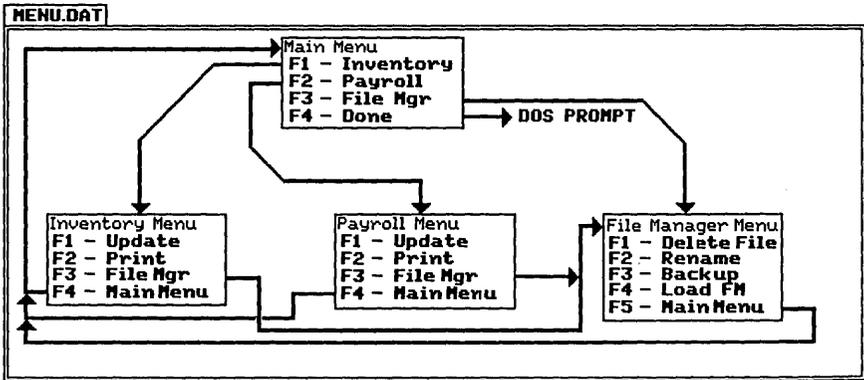


Figure 7.2 Sample menu file hierarchy. The menu file shown in Figure 7.1 can be set up with commands and branching capability. For example, pressing F1 at the main menu will cause control to be transferred to the Inventory Menu, and pressing F4 will return the user to the main menu.

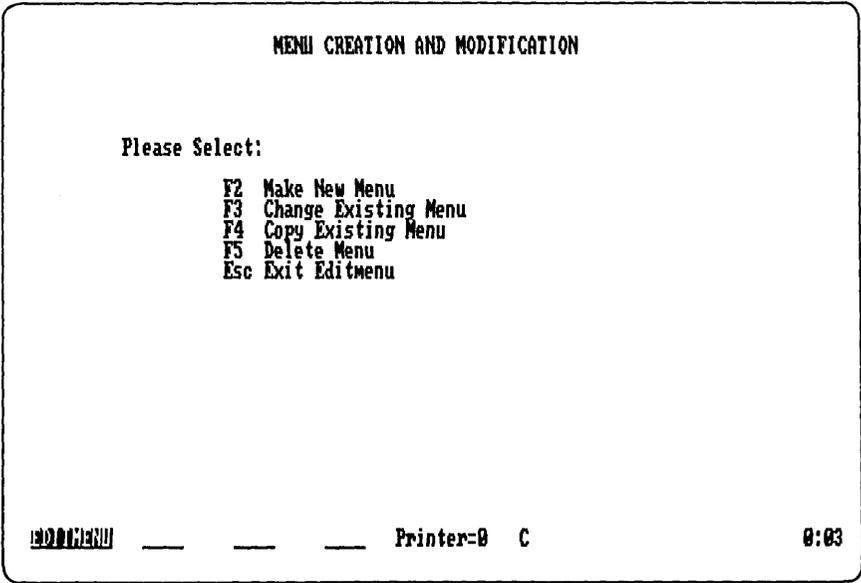


Figure 7.3 The EDITMENU "Main Menu" screen.

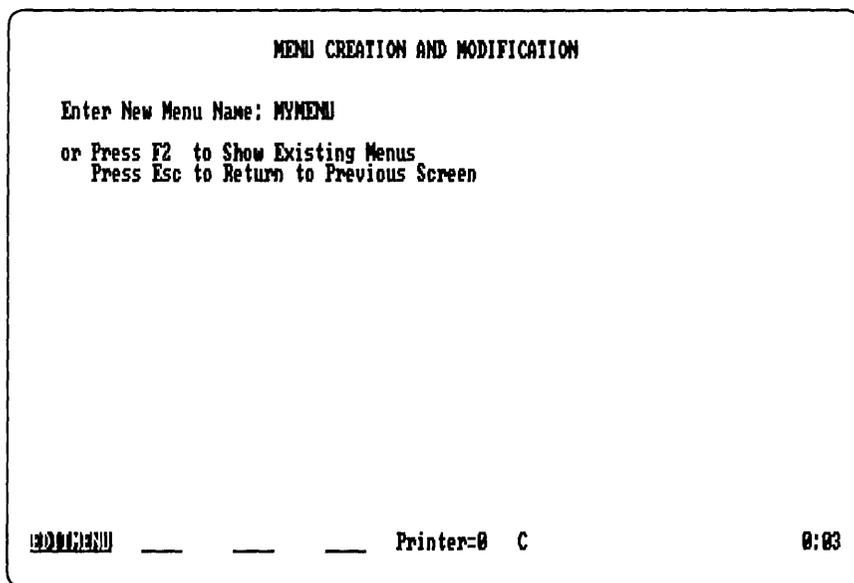


Figure 7.4 The EDITMENU "Menu creation" screen.

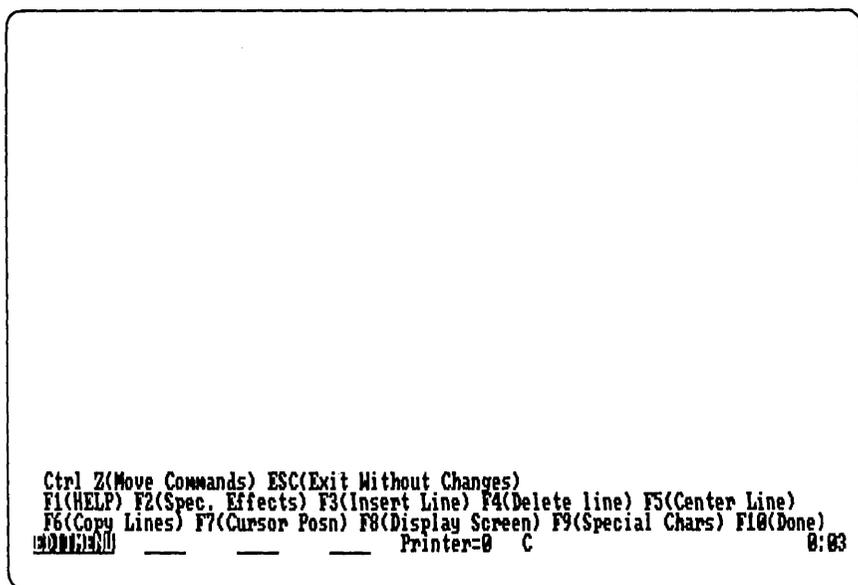


Figure 7.5 The EDITMENU "edit" screen.

function to a typical full-screen word processor editor. You can type text or place special characters anywhere on the screen. Special functions are available, and they are listed in the prompt menu at the bottom of the screen.

The function keys used in EDITMENU are similar to those found in many word processors, but additional screen formatting commands allow you to create menus with color, highlighting, and blinking text. Figure 7.6 shows the special effects screen.

Characters that aren't directly available from the keyboard, such as the block graphics characters, may be used in a menu by pressing the F9 key (see Figure 7.7).

The arrow keys can be used to select a special character. Pressing Enter will select the character and place it on the menu at the current cursor location. The F7 through F10 keys may be used to place a character on the menu and move to another adjacent character location.

The F1 function key is available from the Editing screen; pressing the key displays two Help screens that show all function and editing keys, and the special effects commands.

A sample menu is shown in Figure 7.8. Note that three function keys

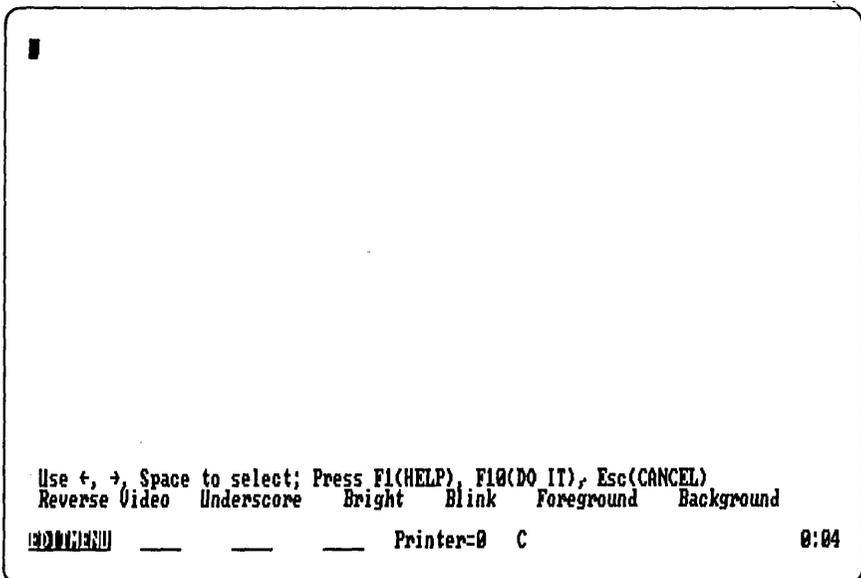


Figure 7.6 EDITMENU special screen effects.

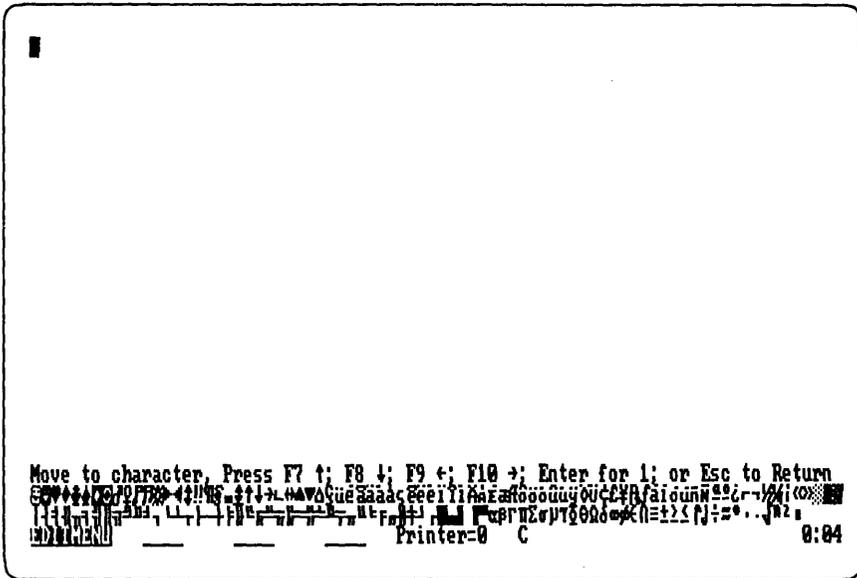


Figure 7.7 EDITMENU special characters.

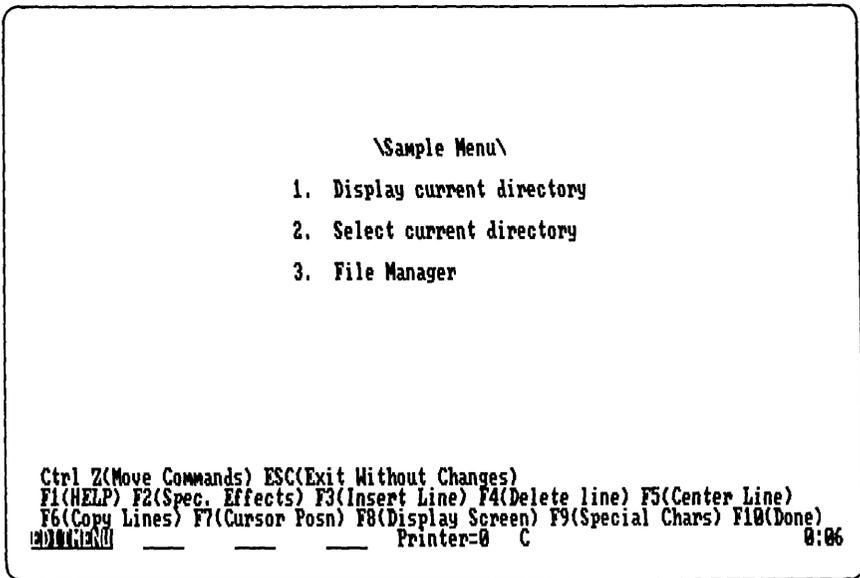


Figure 7.8 A sample menu created with EDITMENU. Function keys F1, F2, and F3 will invoke DIR, CHDIR, and FM respectively.

have been used in this example. When you are finished editing the Menu screen, pressing F10 will take you to the Assign Function Key screen. This screen lets you assign commands to each of the function keys listed in your menu. For example, when the first function key in Figure 7.7 is pressed, the current disk directory will be displayed. On the Function Key screen, you would enter the command DIR in the field labeled "Menu Name or Command Text." This tells the RUNMENU program that when F1 is pressed, the DIR command should be processed just as if you had typed it in by hand. The "Type" field tells RUNMENU what kind of command you are entering. The allowable types are:

- A Complete command line for an application program.
- O Command line requiring the user to enter options to complete the command.
- M Name of another menu in the menu file.
- C Command that the user must enter when prompted.
- E Exit to the Concurrent PC DOS command prompt C>.
- Blank No command is assigned to the function key.

The "A" option will treat the command text as a complete command that will be executed by Concurrent just as if you had typed it in manually. The "O" option will display the command text you have entered and then wait for you to complete the command and press Enter. If "M" is specified, RUNMENU will assume that the entry is the name of another menu and try to find it in the menu file and display it. This lets you create a series of interconnected menus that can dramatically simplify the operation of your personal computer and save you from having to remember frequently used commands.

As shown in Figure 7.9, when you have completed filling in the fields of the Function key screen, you may press the F2 key to assign further attributes to the menu. Figure 7.10 shows the options screen. Normally you should not need these options, but they are provided for advanced users who need greater control over their menu systems.

Within menus, the Escape key is used to return to a previous menu. If this capability is not desired, the Special Options screen can be used to turn off the Escape key. This could be needed in a case where operations must be performed in a precisely defined sequence, such as updating a database. You might not want the user to be able to back up through the menus and execute a function more than once.

Normally, if you press Enter while a menu is displayed, RUNMENU will display a Concurrent PC DOS command prompt to let you enter any command you want, even if it isn't in the menu. This default

MENU CREATION AND MODIFICATION		
Fcn	Type	Menu Name or Command Text
1	A	DIR
2	O	CD \
3	A	FM
4		
5		
6		
7		
8		
9		
10		

Press ENTER or TAB for next field, Shift TAB for prior field  
 F1(HELP), F2(Disable ENTER, ESC Options) F10(Done), Esc(Prior Screen)


 Printer=0 C Caps 0:07

Figure 7.9 Assigning the function keys for use with the menu in Figure 7.8.

MENU CREATION AND MODIFICATION	
Disable ESC to return to prior menu level? (Y/N)	N
Note: The prior menu level is the system prompt if this is the main menu.	
Disable ENTER to enter a command in Concurrent DOS Format? (Y/N)	N
Save and restore default drive? (Y/N)	N
Save and restore user number ? (Y/N)	N
Command to execute at start of menu:	
Command to execute at end of menu:	
Press F10 to continue	
 Printer=0 C Caps 0:07	

Figure 7.10 Special options screen.

feature can be overridden by typing "Y" in the second field on this screen.

For certain applications, it may be necessary or desirable to save and restore the value of the default or current disk drive and the current user number (for CP/M media). If the menu issues any commands that could alter these values, it is generally a good idea to protect them so that you won't lose your place upon exiting the menu.

If a menu is being used to automate a series of operations, such as a database update or the end-of-month inventory report, you may have to preload certain programs (such as a print spooler) or issue a command to set the current subdirectory. This can be accomplished by specifying startup and terminate commands for each menu. These commands will be executed automatically before the menu is displayed and when you leave the menu.

## **COPYMENU**

The COPYMENU utility allows you to make backup copies of menu files, condense space in menu files, and copy menu files that you can later modify. If you don't enter any options after the command, you will be prompted for source and destination files.

---

## Chapter

# 8

## The Printer Manager

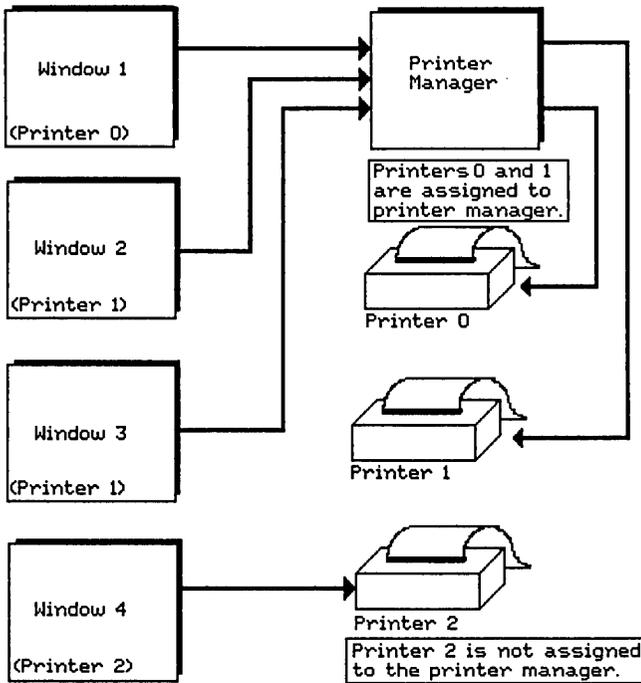
Concurrent PC DOS comes with a Printer Manager that runs in background mode just like the PC-DOS PRINT spooler command, but unlike that command, the Printer Manager lets you use more than one printer. In the extreme case, you could have printers attached to all five external interfaces (three parallel and two serial ports) and use all of them simultaneously.

The Printer Manager also lets you specify options for formatting the output of a file. For example, you can specify how many copies of a file to print, whether to format the file for a certain number of lines per page, and to which printer the file should be sent. You can also have the pages numbered as well as the left margin and tab stops set.

Figure 8.1 shows how the Printer Manager interacts with the windows and the printers. As with the Window Manager, the Printer Manager runs separately in the background so it doesn't occupy a window. It waits for print jobs to be sent to it by the PRINTMGR command or the Printer Manager menu. As files are received, it stores the file names and other formatting options in a queue. All print requests are serviced on a first-come first-serve basis for each printer assigned to the Printer Manager.

### Starting the Printer Manager

There are two methods of starting and controlling the Printer Manager. If you have installed Concurrent on a fixed disk system, the



**Figure 8.1** The Printer Manager. Each window can be assigned a default printer device. The Printer Manager can accept input from several programs at once and deliver output to as many as five printers at once.

HDMENU startup menu will have a Printer Manager entry, which can be selected by pressing F6. If you are using a floppy disk-based system, you will probably have to set up the Printer Manager on a separate disk that can be inserted when you want to start the Printer Manager. The Concurrent PC DOS User's Guide explains how to do this. Figure 8.2 shows the startup menu's Printer Manager menu.

To start the Printer Manager you can press F2, which will give you a list of printers from which to choose (see Figure 8.3). Depending on how your system is configured, you may have one parallel port or a serial port, or up to three parallel ports and two serial ports. Each of these five devices is assigned a number from 0 to 4. If you have more than one printer, you should choose the F7 key and enter the numbers of each of the printers you want to assign to the Printer Manager. Pressing ESC will return you to the previous screen.

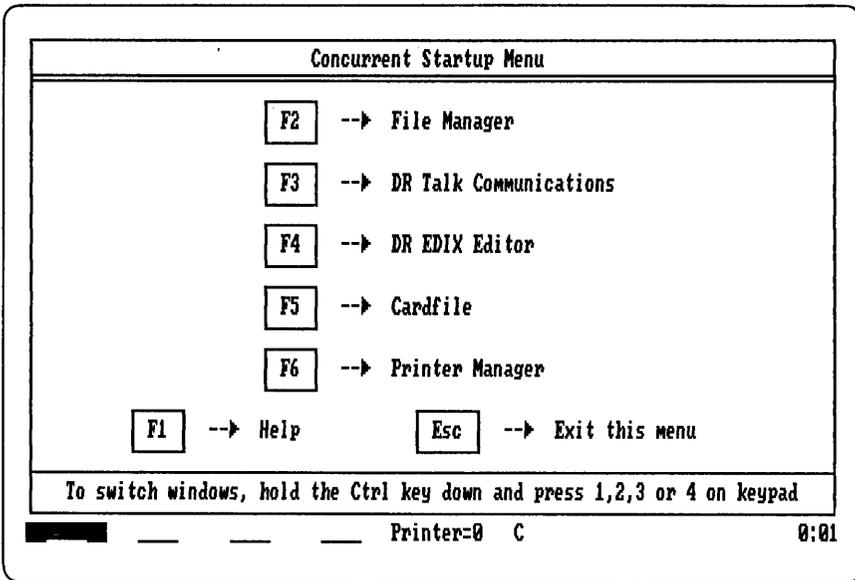


Figure 8.2 The HDMENU Printer Manager submenu.

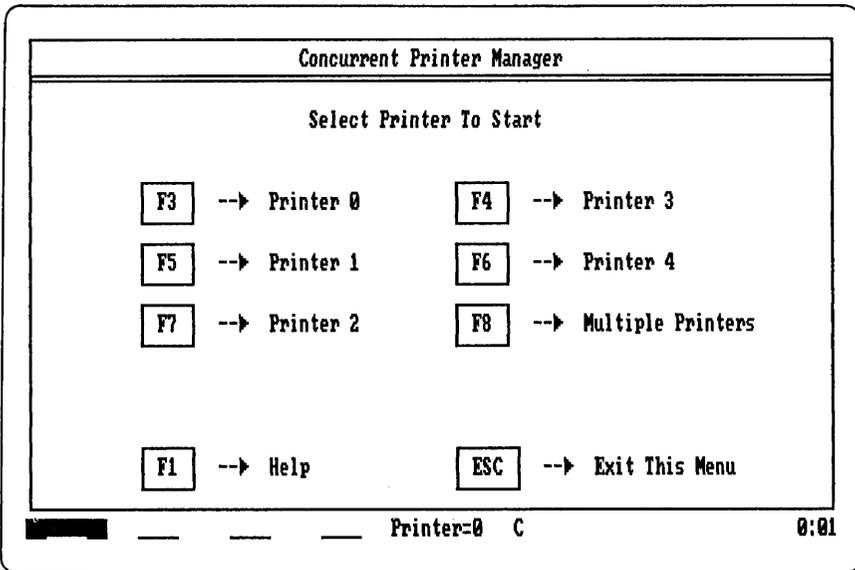


Figure 8.3 The Start Printer submenu.

If you want to start a print job, you can press the F3 key and the screen in Figure 8.4 will appear. Pressing F2 will allow you to type in one or more file names to print, along with formatting options if needed. Figure 8.5 shows a command being entered which will cause three copies of the LETTER.TXT file to be printed on the default printer. The default printer is the printer assigned to the current window. This may be overridden in the command by adding PRINTER=n, where n is the desired printer number. Before you issue the Printer Manager command (or load the menu) you may set the default printer for a window by typing PRINTER=n at the Concurrent command prompt.

You can also display the status of the Printer Manager at any time either through the Startup menu shown in Figure 8-2 or through the direct command: PRINTMGR STATUS. Figure 8.6 shows a status display with several print jobs in the queue.

The Printer Manager also offers a help screen which may be displayed by typing PRINTMGR HELP. Figure 8.7 shows this display.

Other help screens are available in the Startup menu and in the HELP command.

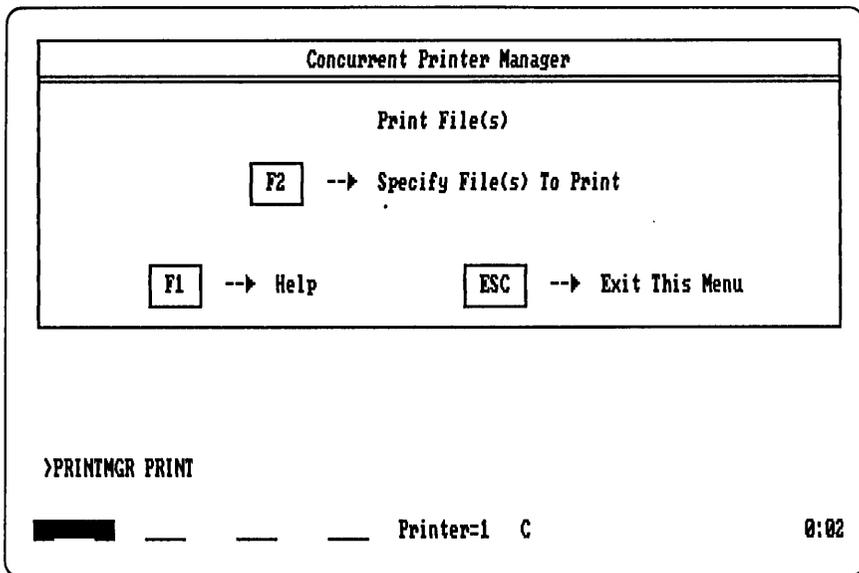


Figure 8.4 The Print File submenu.

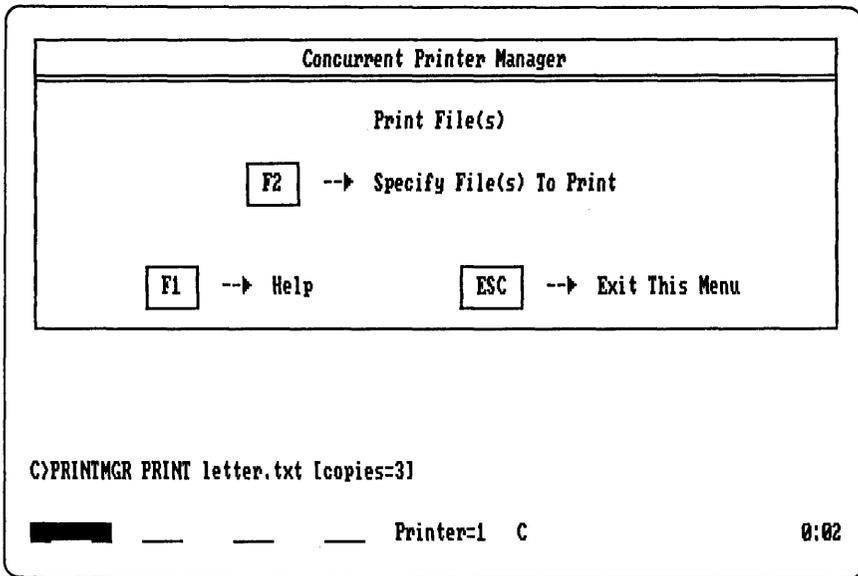


Figure 8.5 Sample command. "PRINT LETTER.TXT [COPIES=3]"

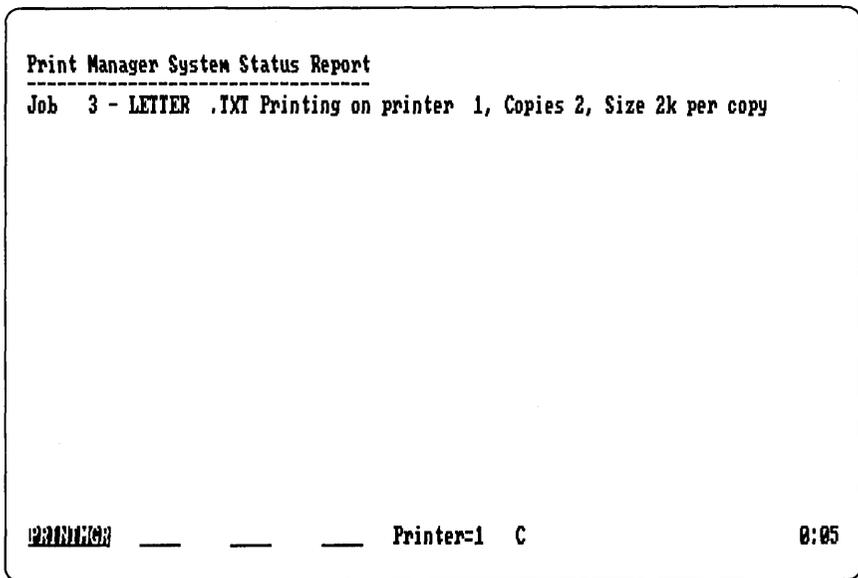


Figure 8.6 Printer Manager status display with a job printing.

```

-----
Print Manager Command Summary
-----
Use the PRINTMGR command and one of the options listed below:
PRINT filename filename filename ... [option, option, ...]

Spools each file with the specified options.

OPTION      DEFAULT      DESCRIPTION
-----
COPIES=n    1             Sets the number of copies to print.
              (n is 1 to 250).
TAB=n       8             Sets tab position to every nth space.
MARGIN=n    8             Sets the left-hand margin.
NOFEED      FORM FEED     Prevent formfeeds before each copy.
PRINTER=n   CURRENT LST: Specifies the printer to print the file.
FORMAT      NO FORMAT     Inserts page breaks.
FORMSIZE=n  66           Sets the length of the paper in lines.
PAGESIZE=n  66           Sets the number of lines printed per page.
NUMBER      NO NUMBERS    Numbers each page sequentially.

Type any key to continue.

PRINTER=1  C
0:06

```

Figure 8.7 The Printer Manager help display.

```

-----
Concurrent Printer Manager
-----
Suspend/Stop Printer Manager

[F2] --> Suspend The Printer Manager
[F3] --> Stop The Printer Manager

[F1] --> Help          [ESC] --> Exit This Menu

Printer=1  C
0:06

```

Figure 8.8 The Printer Suspend/Stop menu.

### **Stopping the Printer Manager**

It is possible to stop a print job before it is completed, just as in PC-DOS. This is done with the `DELETE` command in the Startup menu or with the `PRINTMGR DELETE` command. You must specify the job to be deleted by its number instead of by its name. If you specified multiple copies when you submitted the print job to the Printer Manager, all copies will be deleted.

### **Suspending the Printer Manager**

You may at some time find yourself with a full print queue, but for some reason it is necessary for you to shut down your personal computer or perform a system reset. This would cause the print queue and all the jobs in it to be lost (although the actual files will be safe). This situation can be avoided by using the `SUSPEND` command. `SUSPEND` may be invoked through the Startup menu by pressing `F6` (see Figure 8.2) or by entering the `PRINTMGR SUSPEND` command. Pressing `F6` will cause the screen in Figure 8.8 to be displayed. If you select `SUSPEND`, the job queue will be saved so it can be restarted at a later time. If you select `STOP`, the job queue will be deleted. With both `SUSPEND` and `STOP`, the Printer Manager modules are deleted from the main memory and must be started again to resume printing.

---

## Chapter

# 9

## The Backup/Restore Facility

The problem of making backup copies of important files has been compounded by the widespread use of 10M byte and larger fixed disks. Directory segmentation (the use of subdirectories in a tree structure) makes organizing files into logical groups easier, but it also frustrates efforts to make backups. One standard method is to back up files by subdirectory. A simple mass copy using a wildcard filename such as \*.\* (all files) or \*.dat (just data files) will transfer all files in a subdirectory to a floppy disk for safe keeping.

Problems do arise though. First, copying all files in a directory is very inefficient. Most files will probably not have changed, while some, such as program (.EXE, .COM, .CMD, and .BAT) files, don't need to be backed up periodically if you keep the original or master diskette separately in a safe place. Also, a mass copy cannot handle special situations, such as when the destination or backup disk becomes full before all files are transferred. In the latter case, you would have to know which files have already been backed up and manually copy the remainder to another disk.

Ideally, a backup facility should allow you to choose groups of files to copy, specify files that shouldn't be copied, and allow for files that are too large to put on one floppy disk. It should also keep track of where the backup copies are (by volume labels) and allow you to restore a file in case the working copy has been damaged or lost. However, some backup/restore utilities give you an all-or-nothing choice: restore all files on the hard disk, or none at all. Restoring just one file or a group of files means reloading the entire hard disk and losing other files in the process.

Concurrent PC DOS comes with Back and Rest, two programs that will perform a backup or restore operation according to a customized control file that you can create. These utilities, collectively referred to as BackRest, make it easy to keep your hard disk organized and all your files protected.

### BackRest Files

BackRest needs three files to operate: BACK.CMD, REST.CMD, and CONTROL.BR. The .CMD files are the actual backup and restore programs that get their instructions from the CONTROL.BR file. A sample control file is shown in the following listing. The control file tells BackRest what files you want to back up; what files, directories, or user numbers are exceptions to the rule; how to report the activities of BACK and REST; and so on.

```

Files Backed Up
Drive C Backup Volume 00001

Path \CDOS
BACK .CMD K BATCH .CMD K CHDIR .CMD K CONTROL .BR K COPY .EXE K
DSKMAINT.CMD K ERASE .EXE K FM .CMD K PIP .CMD K REN .EXE K
REST .CMD K SYSDISK .CMD K TYPE .CMD K

Files Skipped
Drive C

Path \CDOS
CONTROL .BAK K DIR .BR K PATHS .BR K REPORT .BR K

Statistics Report

Path \CDOS
Drive C: 232K in 17 files. 13 files backed up.

Drive C: Total of 13 files backed up out of 17.
Drive C: 348K available.
```

In operation, BackRest creates several additional work files, all with the .BR file type. Because BackRest uses this file type, none of your files should have a .BR file type. The three types of work files are: a directory file, a path file, and a report file. The directory file, DIR.BR, is used to keep track of the files that have been copied to each of the backup diskettes. The path file, PATHS.BR, records all the path names encountered on PC-DOS-media disks during a backup operation. Two

types of report files are created: REPORT.BR and RESTRPT.BR. These files report the results of backup and restore operations respectively. Other files are created that are temporary and have the .BR@ file type. These files should not be erased, renamed, or changed to read-only status.

BackRest will only back up files not previously backed up and files that have changed since the last backup. All files are written to one or more diskettes which must have been previously formatted according to the disk media being backed up. For example, if BackRest is to back up a CP/M-media fixed disk partition, it will request that you insert CP/M-media floppy disks into the destination drive. Each backup disk will be given a volume label which is written as a disk file with a length of zero bytes. For example, the label -C-00001.VOL indicates that the files came from the C: drive and that this is the first diskette in the sequence of backup diskettes.

BackRest keeps track of all files that have been backed up in the DIR.BR file. It contains the name of each file, the date the file was backed up, and the volume number of the backup diskette it was written to.

### **Disk Media Considerations**

BackRest can handle files stored on either PC-DOS or CP/M media, but because of the media differences discussed in Chapter 5, different procedures are often required for each media. For example, CP/M media allows for password protection of files. If you are using passwords, you must specify them in the CONTROL.BR file. All protected files will be backed up, but the backup diskettes will not be password-protected; therefore, these diskettes should be stored in a place where unauthorized personnel will not have access to them.

### **Control File Options**

The listing in the BackRest files section shows a typical CONTROL.BR file used with BackRest. Multiple control files may be created for various applications. For example, you could set up special control files to back up files in groups by application. These control files must have an identifier character attached to the file name, such as CONTROLA.BR. To use the special control file when invoking BACK or REST, you would specify CONTROL=A in the command line.

The first group of control statements in CONTROL.BR defines console controls, such as the commands that will clear the screen and

set color attributes for errors and prompt messages. The next group defines the printer control sequences that tell BackRest what your printer is capable of. These controls are described in detail in the Concurrent PC DOS User's Guide and the BACKREST.DOC file that is included on the distribution diskettes.

Report generation is controlled by the REPORT PRINT, SHOW SKIPS, and ID commands. If REPORT PRINT is followed by the keyword TRUE, a report will always be generated when BackRest is finished. SHOW SKIPS TRUE adds a list of files not backed up to the report. ID allows you to put a title at the top of the report. This feature is handy when you have several CONTROL.BR files or several personal computers to keep track of. The ID field could be "Inventory File Backup" or "Accounting Dept PC Backup," for example.

The next group of control records tells BackRest what options are desired in backup and restore operations. SPLIT: TRUE allows files to be split across two diskettes if there is insufficient space on one diskette. BELL REPEAT: TRUE enables beep prompts. DEST DRIVE: lets you specify the destination disk drive to which the backup files are to be written. SOURCE: likewise selects the drives to be backed up. CONTROL DRIVE: defines the drive to be used by BackRest for work files. VERIFY: TRUE enables read-after-write verification of all files to ensure complete accuracy. REUSE: TRUE allows backup diskettes to be reused. ERASE: TRUE causes diskettes used for backup to be erased before being reused. USERS: lets you select which CP/M-media user numbers (disk areas) are to be backed up and restored.

Paths for PC-DOS media may also be specified. If no paths are given, only the current directory will be backed up. For example:

```
PATH: C:\
PATH: C:\inventory
PATH: C:\accounts
```

will cause the files in the root directory and the inventory and accounts directories to be backed up.

## Exceptions

The above control records define the normal operation of BackRest. The following commands let you specify exceptions to the above rules; for example, you may not want to back up certain files. Exception records also let you delete files after they have been copied. The format

of an exception record is:

EXC: user/D,d,process,disposition,filename.ext[,password]

where:

user/D	specifies a user number for CP/M media or the letter "D" for DOS media.
d	is the source drive.
process	is the process to be used: "A" for always back up the specified files, "C" for conditional backup, and "N" for never back up the files.
disposition	can be either "D," to delete the files from the source drive once backed up, or "K," to keep the source drive files.
filename.ext	can be any file name, with or without wildcard characters (? or *), to match one or more file names. The last field is optional and for CP/M media only, if a password is required to read the file.

For example, if you want to back up all files of the type \*.DAT on the C: disk, the command EXC: D,C,A,K,\*.DAT would perform this function. While you are running backups, it is also possible to free some disk space by getting rid of the .BAK files that are created by many programs and word processors. This can be accomplished by the command EXC: D,C,A,D,\*.BAK. All files on the C: drive (DOS media) will be backed up and then deleted if they have the .BAK file type.

### Invoking BackRest

Two methods are available to run BACK or REST. As usual, the BACK or REST commands can be entered at the C> prompt, or they can be put in a batch file. The File Manager also has a Backup/Restore menu entry (see Figure 9.1).

From the C> prompt, the CONTROL.BR file will be the sole source of commands for a backup or restore operation. You may specify three parameters, FULL, CONTROL=n, and REPORT, for the BACK command, and two for the REST command, CONTROL=n and REPORT.

The FULL option will force a full fixed disk backup, which causes BACK to ignore references to previously backed up files. All files that match the commands and exceptions in the CONTROL file will be backed up. CONTROL=n lets you select an alternate control file for the backup operation. REPORT forces generation of a report if the REPORT PRINT: command was set to FALSE. Normally a report is always generated.

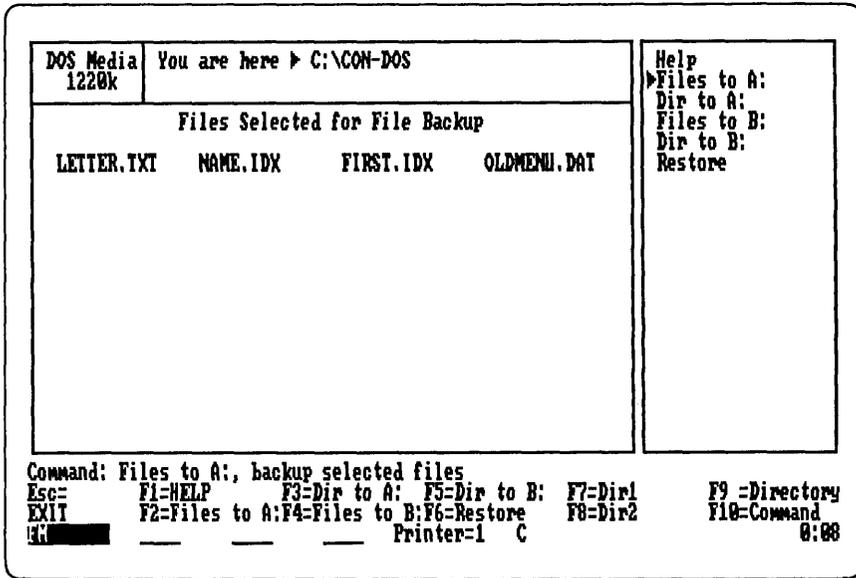


Figure 9.1 The File Manager BackRest (Backup/Restore Facility) submenu.

The REST command also lets you give the CONTROL=n and REPORT overrides of the CONTROL file.

The second method of invoking BackRest is from the File Manager. From the main menu panel of the File Manager you can select a group of files to back up by pressing the Tab key and using Insert and Delete to choose files from the menu. If you then invoke the Backup Files menu, you will be presented with a list of all the files you selected. If you then specify the destination drive and the Backup Files command, the File Manager will create a CONTROL.BR file and call BACK for you. All the files you specify will be included in the control file by name in EXC (exception) command records. From then on, BACK will work as described above.

# 10

## Customizing Your Personal Computer

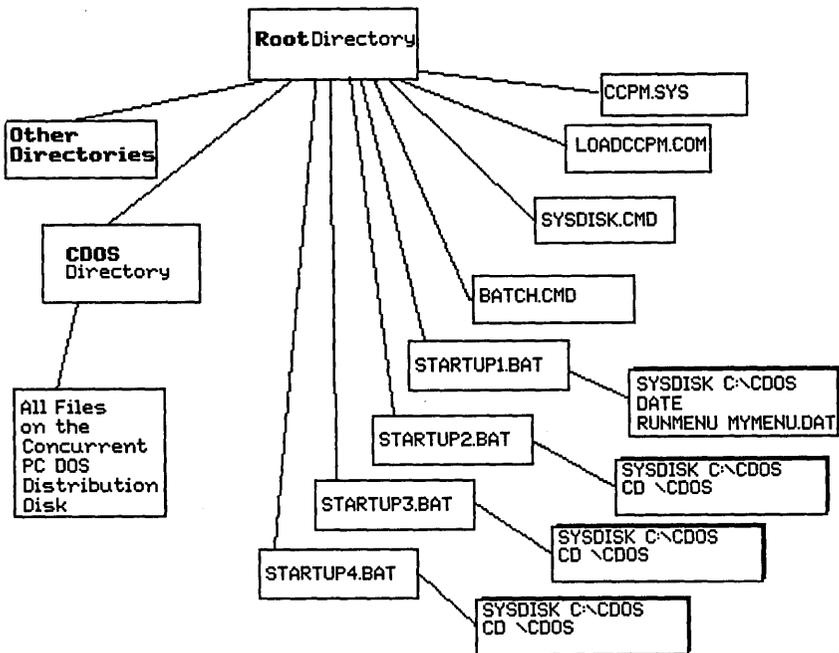
Concurrent PC DOS has a wide variety of features that can be used to customize your personal computer working environment. You can create batch files that will be executed upon start-up of Concurrent, change the colors used in the File Manager and other programs, and cause programs like the Window Manager or Printer Manager to be loaded and set up automatically when you cold-start or restart your computer. Other commands, such as SYSDISK, let you choose a disk or directory as your system disk. As a result, no matter what disk or directory you are currently in, all system commands and utilities are always accessible. The Backup/Restore facility can be customized to back up just the files you are working with on any given project or application, and files may be assigned passwords to protect them from unauthorized access. The function keys (F1 through F10 and keypad) may be customized for your convenience. You can also set up your system to load PC-DOS or MS-DOS first, and then ask you whether you want to load Concurrent PC DOS.

### **Autostart Batch Files**

In PC-DOS and MS-DOS you can create a file called AUTOEXEC.BAT which will be executed after the DOS has been loaded into memory. This lets you configure your system by executing commands and

setting options that can only be done each time you power up or restart your system. For example, you might want to enter the current date and time or load a special device driver. Another good use for the AUTOEXEC.BAT is to set a PATH command in PC-DOS so that if a file isn't found in the current directory, the DOS will check an alternate "path" for it.

Concurrent PC DOS lets you set up AUTOSTART batch files, but since there are four windows, you may have a STARTUPn.BAT file for each of the windows (where n is a number from 1 to 4). The arrangement many people use is shown in Figure 10.1.



**Figure 10.1** Sample fixed disk setup for Concurrent PC DOS. In this example, the root directory contains only the files required to load and start running Concurrent. CCPM.SYS, LOADCCPM.COM, SYSDISK.COM, and BATCH.COM are required to start it. The four batch files (.BAT) contain commands that initialize each window. The first command in each is the SYSDISK command, which sets the default system volume and path, allowing access to all other Concurrent PC DOS files and commands. The CDOS directory is used to hold Concurrent, partly to keep nearly one megabyte of files out of the root directory and partly to keep them organized separately from PC-DOS system files and commands, which may also be on the disk.

Often it is convenient to keep all the Concurrent PC DOS commands and utilities in a separate directory, so that the root directory of a DOS-media disk will not be too cluttered. In Figure 10.1, a directory called CDOS has been created and a SYSDISK command has been added to each of the STARTUP batch files to allow each window access to this directory. Any commands not found in the currently selected directory or disk will be found in the CDOS directory.

Additionally, a file called WSETUP.BAT can be created that contains a group of WINDOW commands used to alter the locations, sizes, and colors of each of the four windows. The WINDOW command has a WRITE option that will build the WSETUP file for you, based on the current window configuration (see Chapter 4). Once the parameters are saved in the batch file, the window configuration will be restored to its former arrangement upon start-up of Concurrent.

### Changing Colors

As discussed in Chapter 4, window colors can be changed to make each window stand out from the others when they are overlapped on the screen. The technique of color-coding windows makes them easy to identify and remember. Additionally, the File Manager and BackRest facility colors can be tailored according to personal preference. The File Manager configuration file, FM.CFG, contains keywords that let you specify the colors to be used in each panel of the display. The following table shows the configuration options for the File Manager.

SAMPLE FM.CFG FILE:

```
OBJECT BG COLOR=0
OBJECT FG COLOR=8
COMMAND BG COLOR=4
COMMAND FG COLOR=0
PROMPT BG COLOR=7
PROMPT FG COLOR=0
KEY BG COLOR=14
KEY FG COLOR=15
```

### COLOR SPECIFICATIONS:

<i>Number</i>	<i>Color</i>	<i>Number</i>	<i>Color</i>
0	Black	8	Gray
1	Blue	9	Light blue
2	Red	10	Light red
3	Magenta	11	Light magenta
4	Green	12	Light green
5	Cyan	13	Light cyan
6	Brown	14	Yellow
7	White	15	High-intensity white

Colors are specified by number, as shown. The foreground and background colors of the command, object, and prompt panels can be set, as well as the function key legend colors. If the default colors cause you eyestrain, this feature lets you change them to whatever combination you find pleasing.

### **The Printer Manager**

The Printer Manager is discussed in detail in Chapter 8, but several other issues need to be considered. First, the Printer Manager is not automatically loaded when you start up your system—this is accomplished by either invoking the Printer Manager from the HDMENU or FDMENU, or entering PRINTMGR START at the C> prompt. Also, each window defaults to printer 0 (zero) unless you add a PRINTER command to each of the four STARTUP batch files. You may have one printer that is shared by all of the windows, or you may want to assign a separate printer to each window. As discussed in Chapter 8, the Printer Manager can accept print jobs from any window and direct output to more than one printer at a time. This feature gives you considerably greater flexibility than the PC-DOS Print Spooler.

### **Function Keys**

Unlike PC-DOS or MS-DOS, Concurrent PC DOS lets you easily define character string assignments for all the function keys, including the numeric keypad. Using the SETUP utility you can define new values for the unshifted F1 through F10, shifted F1 through F10, control- and alt-shifted F1 through F10, the arrow keys, Home, PgUp, PgDn, End, Ins, and Del keys. You can also select the shift key assignments for the window switching and control keys in case they conflict with an application program. PC-DOS cannot begin to match this ability.

You may create a function key layout that includes each of the most frequently used commands for your application. If you run Lotus 1-2-3 or a word processor, you can simplify frequently used commands to one keystroke.

### **Cardfile**

The cardfile program lets you create an online index-card file for names, addresses, and phone numbers. It can even print mailing labels. If your personal computer is left on most of the time, an online card file could be very helpful. Cardfile is discussed in detail in Appendix I.

## The HELP Facility

The HELP command eliminates the need to refer to the User's Guide and supports "context sensitive" help in many programs, such as the File Manager. HELP may be altered for your particular needs. The HELP EXTRACT feature lets you edit the HELP.HLP file and add entries for your application programs or special requirements. This can be of great value to a hardware vendor who prepackages a system for a particular purpose, such as accounting or computer-aided design. Help messages can be added for the end user or employee who will actually be operating the system.

## Passwords

CP/M-media floppy and fixed disks support password protection of files. If the PC is being used in a multiuser environment, files can be made secure from unauthorized use or tampering. This is a feature that PC-DOS does not support at all and is therefore a major advantage of Concurrent.

## The Memory Disk

Concurrent can be configured to automatically create a memory disk when the system is started up. The disk will be cleared and ready for use immediately after Concurrent is loaded. One of the STARTUP batch files can be set to copy frequently used system files and programs to the memory disk and the system disk can even be set to search the memory disk for commands. If enough memory is available, a memory disk can greatly speed up the execution of many programs, especially those that perform numerous disk read-and-write operations.

## Using Menus to Create a Turnkey System

Two methods are available for creating a turnkey system (a system that either runs an application or displays a menu every time the computer is turned on or restarted). The old method of creating an AUTOEXEC.BAT batch file may still be used. This is a batch file that PC-DOS searches for and executes automatically after the DOS is loaded into memory. (In Concurrent, the corresponding file is called STARTUP1.BAT; Concurrent will not recognize an AUTOEXEC.BAT file). This method requires that you use a text editor to enter the command or commands that you want to run. Each command will be executed as it appears in the batch file and just as if you had entered it manually. In order to create a menu of options, you would have to

write a short program that would display a list of program names or tasks and transfer control to the program of your choice. The menu creation facility described in Chapter 7 can be used to simplify the task. Figures 10.2 through 10.4 show the steps you would take to build a Startup menu. You can either build yours from scratch or copy the HDMENU.DAT file and alter it to suit your needs.

To make your menu come up when Concurrent is loaded, you will need to make a small batch file called STARTUP1.BAT, which will be executed in window 1. Similar files for windows 2 through 4 can also be made. The file could contain the following statements:

```
SYSDISK C:
DATE
RUNMENU MYMENU . DAT
```

where MYMENU is the sample shown in Figures 10.2 through 10.4.

### Your PC as a Multiuser System

Concurrent PC DOS can support two additional consoles or users through the two serial ports, COM1: and COM2: However, many problems come up when you try to do this because most programs are

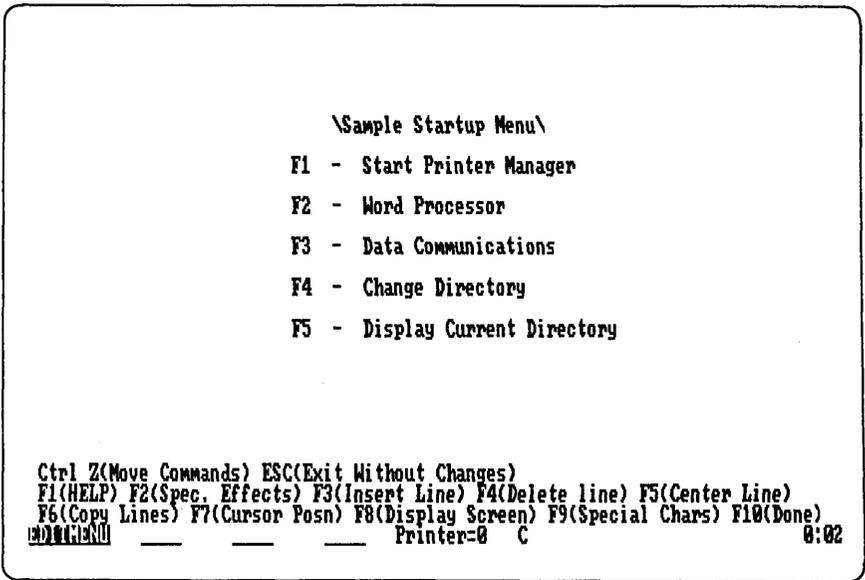


Figure 10.2 EDITMENU sample screen.

```

                                MENU CREATION AND MODIFICATION
Fn  Type  Menu Name or Command Text
 1    A    PRINTMGR START 1
 2    A    WS
 3    A    DRTALK
 4    O    CD \
 5    A    DIR
 6
 7
 8
 9
10

Press ENTER or TAB for next field, Shift TAB for prior field
F1(HELP), F2(Disable ENTER, ESC Options) F10(Done), Esc(Prior Screen)
EDITMENU  _ _ _ _ Printer=0 C           Caps 0:04
    
```

Figure 10.3 EDITMENU sample function key assignments.

```

                                Sample Startup Menu
                                F1 - Start Printer Manager
                                F2 - Word Processor
                                F3 - Data Communications
                                F4 - Change Directory
                                F5 - Display Current Directory

                                _ _ _ _ Printer=0 C           Caps 0:04
    
```

Figure 10.4 The final product of EDITMENU.

not designed to work correctly when used with a console device other than the color or monochrome adapter card. This is due to the problems described in Chapters 2 and 3. Many application programs assume that the console device is a full screen that is actually addressed as random-access memory (RAM), instead of as a video display terminal that is connected through a serial interface adapter. Most serial terminals do have the ability to address the full screen and position the cursor at will, but the codes that allow this aren't compatible with the commands used by the color or monochrome display cards. Also, since many word processors and spreadsheet applications address the screen directly and bypass the operating system, there is no way they could possibly work with a serial terminal. Some programs do work with a serial terminal, and you can generally tell if they do by checking to see if they have an installation program that lets you select the type or brand of terminal. For example, some programs will give you a list that includes the Lear Sigler ADM 3A, the IBM 3101, Soroc, or other terminals. Some programs even let you enter the codes to define your own terminal if it isn't on the list.

This process is similar to the printer installation questions asked by many word processors. Many printers use different escape (command) character sequences to turn on and off special printing features. If you are unsure whether or not a program will work with a serial console, ask your dealer or call the vendor who wrote the program. One word processor that will work on a serial console is Wordstar from Micropro.

Some of Concurrent's own utilities, such as File Manager, DR Talk, DR EDIX, DSKMAINT, EDITMENU, HDMAINT, and RUNMENU, will not run on a serial console. The menu versions of FUNCTION, PRINTMGR, SETPORT, and SETUP also will not run.

One good use of the multiuser capability is to attach a terminal and write a custom program for tasks like inventory or order entry. A query workstation will not demand much of the processor's attention and will not degrade the response time too much. By writing a custom application, you can take advantage of Concurrent's features and make certain that the other users are protected from running programs that might crash the system or trying to use the internal hardware in ways that circumvent the operating system.

It is also possible to turn your computer into a minitimesharing system by hooking an autoanswer modem to one or both serial ports. This allows you or others to phone in from any location to transfer files or run programs. Since Concurrent supports password protection of files, such a system would be easy to install and maintain.

## Other Considerations

Any program that communicates directly with the serial interface hardware could cause problems when it tries to support multiple users. The BasicA language reprograms the serial ports whenever a program tries to initialize COM1 or COM2. Therefore, no interpretive BasicA programs should be run in a multiuser environment.

Naturally, the more tasks you have in memory at once, the slower your personal computer will run. This problem is compounded by one or two additional users. In the single-user case, if you have two or three programs running, not much will be happening in the windows you aren't looking at (assuming that you aren't compiling a program or using DR Talk in the background). If other people are trying to do things at the same time that you are, you will have to wait longer than normal. Not much can be done about this except to get a faster computer, such as the DeskPro or IBM PC AT.

---

## A p p e n d i x



# Cardfile

Cardfile is a utility program included with Concurrent that simulates an address book. It displays "electronic" index cards on the screen and allows you to store names, type of business or company name, addresses, phone numbers, and comments. Functions such as search, delete, insert, and update are provided. If your personal computer is within easy reach of your desk, and if you leave it on most of the time, this electronic address book could be quite handy.

### Starting Cardfile

Cardfile can be invoked from the startup menu, as described for the Printer Manager in Chapter 8. It can also be started by typing `CARDFILE` at the Concurrent `C>` prompt. Entering

```
CARDFILE          (when using a monochrome monitor)
or CARDFILE COLOR (when using a color monitor)
```

will load the Cardfile program and also automatically load the Cardfile data files containing the names and addresses you have stored.

For Cardfile to operate correctly, the following files must be present on the current drive and directory: `CARDFILE.CMD`, `CARDFILE.DIS`, `CARDFILE.DAT`, `CURRENT.TRM`, `NAME.IDX`, `PHONE.IDX`, `FIRST.IDX`, and `BUSS.IDX`.

Once loaded, you will see a screen display similar to the one in Figure I.1. Three index cards are shown, along with a line of com-

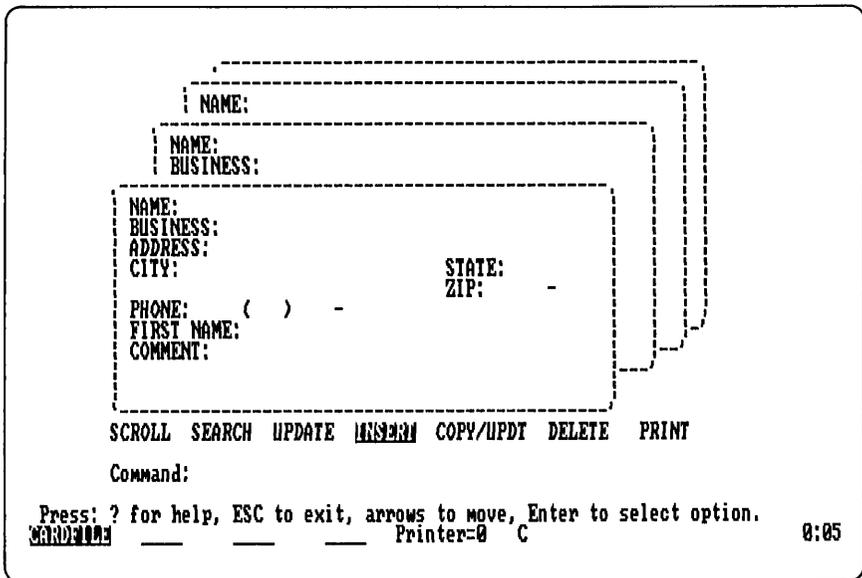


Figure I.1 Cardfile screen display.

mands and a prompt line "Command:". You may press the question mark (?) character for help at any time.

### Selecting Commands

Cardfile commands may be selected by pressing either the left or right arrow keys to highlight the command you wish to use. For example, Figure I.1 shows the SCROLL command highlighted. To add a new card, press the right arrow three times; this highlights the INSERT command. To actually invoke the command, simply press Enter (see Figure I.2). The insert mode lets you type new information directly into the card shown on the screen. After pressing Enter, the front-most card will be cleared of its contents (if any) and the cursor will be positioned to accept new input. You may then type in the person's name. The best format is to enter the last name first, so Cardfile can perform searches more effectively. When you are satisfied with the name field, you can press Enter to go on to the business field. After you press Enter, Cardfile will not let you go back and edit the field while in the insert mode. If you make a mistake, leave it for later correction. Also, if you want to leave a field blank, just press Enter and go on to

```

NAME:
BUSINESS:
NAME: Mark Dahmke
BUSINESS: MCD Consulting, Inc.
ADDRESS:
CITY: Lincoln STATE: NE
PHONE: (402) 999-9999 ZIP: 68501-
FIRST NAME: Mark
COMMENT:
*
SCROLL SEARCH UPDATE INSERT COPY/UPDT DELETE PRINT
Command:
ESC to exit card, Enter for next line, Arrow up for previous line.
CARD#0001 _____ Printer=0 C 0:06

```

Figure 1.2 A new card being entered into Cardfile.

the next one. The "State" field will only accept two characters and it is not necessary to press the Enter Key to move on. This holds true for the "Zip code" and "Phone Number" fields as well. The "Comment" field allows you to enter one 40-character line of text for any purpose.

After you have entered or skipped all the fields, Cardfile will ask if you want to save the information you have entered. If you do, use the right arrow key to highlight the YES choice and press Enter. Your new index card will be added to the file. If you want to quit and start over without saving, just leave the NO choice highlighted and press Enter.

### Editing, Updating, and Deleting Cards

Old information may become dated, or perhaps you made a mistake when you entered a new card. The update feature lets you go back into a card to put in new information, such as a new phone number or address, or to correct mistakes.

Use the arrow keys to select the Update function and press Enter. You will see the cursor move up to the "Name" field of the card displayed. Any field may be changed, in the same manner as the information was entered with the INSERT command. When you have reached the comment line and pressed Enter one more time, Cardfile

will ask you if you want to save the changes. Select the YES option and press Enter.

The COPY/UPDT command works in precisely the same manner but creates a new copy of the displayed card that can be modified and saved as a completely new record.

The currently displayed card may be deleted by selecting the DELETE command and pressing Enter.

### Searching for a Card

Two methods are provided for finding a card. First, you may scan through your card file. There are two methods to do this; you may either use the up and down arrow keys at any time to manually go through your card file, or you may use the SCROLL command, which will automatically display one card after another until the entire card file is displayed.

Second, the SEARCH command lets you search for a card based on the last name, first name, phone, or business fields. After invoking SEARCH, you will see the following prompt line appear:

```
Select:  NAME   phone  firstname  business
```

In this prompt line, the "Name" field is shown in uppercase letters. By pressing the right and left arrow keys you can cause the phone or other choices to be shown in uppercase too. When you press Enter, the choice that is in uppercase will be the selected search method. The cursor will be positioned so you can enter the number or name to search for. If you enter only one or two characters, the nearest match will be found.

### Printing Labels and Cards

With Cardfile, you can print one or all of the index cards on paper or mailing labels. Only one label is printed across the width of the page. The following listing shows sample labels.

```
NAME:
BUSINESS:  BYTE Magazine
ADDRESS:   70 Main St.
CITY:      Peterborough
STATE:     NH      ZIP: 03458
PHONE:     (603) 924-9281
FIRSTNAME:
COMMENT:
```

```
NAME:      John Doe
BUSINESS:
ADDRESS:   1234 '0' St.
CITY:      Lincoln
STATE:     NE      ZIP: 68501
PHONE:     (402) 123-4567
FIRSTNAME: John
COMMENT:
```

```
BYTE Magazine
70 Main St.
Peterborough, NH
03458
```

```
John Doe
1234 '0' St.
Lincoln, NE
68501
```

**Listing I.1:** Sample Cardfile "cards" and labels.

You can print the contents of a card formatted as labels or you can print one or more cards for hard-copy backup. When you invoke the PRINT command, you will be asked for the type of output desired, labels or cards, and whether to print one card (the one displayed), or all cards.

---

## A p p e n d i x



# DR Talk

DR Talk is an application program that allows you to communicate with other computers. Actually, DR Talk is the same as PC Talk III, a product that is quite popular among IBM PC owners. DR Talk also allows you to gain access to computer services such as Dow Jones, The Source, Compuserve, Dialog, and other timesharing systems. In addition, it can be used to communicate with most other microcomputers, whether the other computer is running DR Talk or not. As long as the computer at the other end recognizes the ASCII character set, you should be able to communicate at either 300 or 1200 bits/s (baud). DR Talk is written to be able to use many of the features of the Hayes Smartmodem products, and should work with other internal or external modems as well. DR talk will work if you use a nonintelligent modem (i.e., a modem with no built-in commands), but many of the features, such as the dialing directory, will not be functional.

### Starting DR Talk

DR Talk loads like any other application program. All of its special functions will default to assumed values, which should allow DR Talk to be used immediately. You may need to reassign it to the COM2: serial port if you are using COM1: for some other purpose. See the section below on Defaults and Configuration Parameters for more details.

If you plan to use DR Talk in the foreground or current window only,

you can just type DRTALK at the command prompt, select it from the startup menu, or run it via the File Manager. After it is loaded, you should see the message:

```
===Proceed. . .
```

in the upper left corner of the screen, which means DR Talk is ready to accept your commands. If this message doesn't appear, you should consult the User's Guide.

If you plan to start DR Talk and run it as a background task, you will first need to issue a SUSPEND=OFF command at the DOS command prompt. This is necessary because Concurrent defaults to SUSPEND=ON, which means that any task that is switched into a background window will be suspended until it is brought back to the foreground. The suspend feature is included primarily for programs that directly modify the screen. If they are allowed to operate in the background, they could cause strange characters to appear on the screen if they bypass the BIOS and window support software in the operating system. Once the SUSPEND command is issued, DR Talk will operate in a background window at up to 1200 bits/s without dropping characters.

If you start DR Talk from the startup menu, the SUSPEND command is automatically issued for you. You may, however, want to create a batch file with the SUSPEND=OFF and DRTALK commands in it to simplify start-up and to avoid having to always turn off the suspend feature.

### Using DR Talk as a Dumb Terminal

DR Talk's primary function is to turn your personal computer into a serial video display terminal that can be used to access other computers. If you don't need all the "bells and whistles" of a smart terminal, DR Talk will do the job without requiring you to learn all the advanced features. After DR Talk loads and displays the "Proceed" message, you may simply start typing (see Figure II.1). If you are using an external, manually dialed modem connection (i.e., an acoustic coupler) you can dial the number, establish the connection, and start communicating without further ado. If what you type isn't displayed on the screen (when you are connected to another computer), you may press the ALT-E key combination to activate the echo feature. This feature, which must be done locally on your computer, is used when the computer you are talking to doesn't echo your characters back to your screen.

```

===Proceed ...

===DR TALK      COMMAND SUMMARY===
↑PrtSc = print screen contents
^PrtSc = contin. printout (or ^PgUp)
Alt-R = Receive a file (or PgDn)
Alt-T = Transmit a file (or PgUp)
transmit: pacing '-p' binary '-b'
tran/recv: XMODEM '-x'
Alt-U = View file Alt-V = delete
Alt-D = Dialing directory
Alt-Q = redial last number
Alt-K = set/clear Func keys (Alt-J)
Alt-= = set/clear temp Alt keys
Alt-E = Echo toggle Alt-M = Message
Alt-S = Screendump Alt-C = Clearsc
Alt-P = communications Parameters
Alt-F = set program defaults
Alt-L = change Logged drive
Alt-W = set margin Width alarm
Alt-Z = elapsed time/current call
Alt-X = eXit to DOS
Ctrl-End = send sustained Break signal

^PrtSc=print Alt- I=tran R=recv U=view D=dial E=echo M=msg X=exit <Home>=help

```

Figure II.1 DRTALK with on-screen help.

There are times when you may need to change the configuration information for DR talk, even when using it just as a dumb terminal. For example, it may be necessary to set the data rate to other than the default 300 baud. This can be done in two different ways, using either the Parameter menu or the Program Defaults menu. The first method can be used by typing the ALT-P combination, which will give you the menu shown in Figure II.2.

If you are using a 300-baud modem, you will have to use either option 1 or 2, depending on the protocol used by the computer you are communicating with. If the protocols shown here don't correspond to those used by the remote computer, you will have to use the ALT-F menu. As you can see, you can select 300- or 1200-baud options.

The ALT-F menu lets you set other DR talk defaults. Figure II.3 shows this menu. As on the ALT-P menu, you can select the data rate, but here you can choose any rate from 75 to 9600 in 10 steps. You can also select the parity-checking option: even, odd, space, mark, or none. The character size can be set to 4, 5, 6, 7, or 8 bits per character, and either one or two stop bits may be selected. When using 8 bits per character, you must set the parity option to "none".

```

===COMMUNICATIONS PARAMETERS===
Present parameters: 300,N,8,1
Echo-N Mesg-N Strip-N Pace-N
Options:
  1 - 300,E,7,1 (text)      2 - 300,N,8,1 (binary)
  3 - 1200,E,7,1 (text)    4 - 1200,N,8,1 (binary)
      F - reset params to defaults
      X - exit to terminal
Choose:

```

Figure II.2 The ALT-P command provides a fast way to select 300- and 1200-baud communication speeds and protocols.

```

===SET NEW DEFAULTS===

```

```

Present program defaults:

```

Baud rate	300	Screendump file	B:SCRNDUMP.PCT
Parity	E	Redial delay	20
Data bits	7	Connect prompt	CONNECT
Stop bits	1	Line 25 help	Y
Echo	N	Foreground	7
Messages	N	Background	0
Strip #1	0	High inten.	15
Replace #1	0	Print port	LPT1:
Strip #2	0	Print init.	,
Replace #2	0	Print width	80
Strip #3	0	Comm. port	COM1:
Replace #3	0	Comm. init.	,CS,DS
Pacing p=	,	Modem init.	,
Logged drive	B:	C/R subst.	}
Margin width	70		

```

Enter <↑> to leave unchanged - <space><↑> for 'null' value - <ESC><↑> to quit
*** Enter new values

```

Figure II.3 DRTALK showing the Default Parameter menu.

## Receiving and Transmitting Files

DR Talk's most useful feature is its ability to send and receive files. Information from a session on a remote computer system such as Dialog can be captured and stored on disk for later analysis or printing. Files can be uploaded to or downloaded from another computer. DR Talk also supports the XMODEM protocol used by many computer communications programs and computer bulletin board systems to transmit files with error-checking protocol. Files transmitted with the XMODEM option are broken into 128-byte blocks or records which are transmitted and checked for errors. If a transmission error is found, the receiving computer notifies the transmitting computer that the block should be sent again. This is the safest method of sending any file, but most importantly for sending binary or program files.

The pacing option can be used when you communicate with remote computers that don't support any kind of error-checking or start-stop protocol. When pacing is activated, you must specify a time delay in seconds. After each line of text is sent, DR Talk will wait this number of seconds before sending the next line. You may also use the pacing option to specify a pacing character. For example, if the remote computer sends back a question mark after a line is received, indicating that it is ready for the next line, you could tell DR Talk to send when a question mark comes back.

## The Dialing Directory

DR Talk supports a dialing directory which is capable of storing 60 names and phone numbers. This feature is most useful when you have a modem that supports autodialing commands. Most modern internal and external modems support this feature. All numbers and other information are stored in a DRTALK.DIR file so they can be used anytime DR Talk is loaded. ALT-D will display the dialing directory as shown in Figure II.4.

As you can see, you may enter the name of the entry, whether it be a friend's phone number or the access number of a remote computer system or value-added service like Tymnet or Telenet. The phone number field can handle up to 36 characters, including dashes and commas. You may also customize DR Talk for the dial-up session by entering the data rate, number of stop bits, parity option, and number of data bits to use with the computer system to be accessed. Also, the character echo (full or half duplex) mode, stripping functions, messages, and pacing may be enabled or disabled according to your needs

```

===DIALING DIRECTORY 1=== Modem dialing command = ATDT
                          Long distance service +# =
                          -# =
Name      Phone #      Comm Param      Echo Mesg Strip Page
1-TYMNET  123-4567  1200-N-8-1      N      N      N      N
2-NUROS   765-4321  300-E-7-1      N      N      N      N
3-----  -----  300-E-7-1      N      N      N      N
4-----  -----  300-E-7-1      N      N      N      N
5-----  -----  300-E-7-1      N      N      N      N
6-----  -----  300-E-7-1      N      N      N      N
7-----  -----  300-E-7-1      N      N      N      N
8-----  -----  300-E-7-1      N      N      N      N
9-----  -----  300-E-7-1      N      N      N      N
10----- -----  300-E-7-1      N      N      N      N
11----- -----  300-E-7-1      N      N      N      N
12----- -----  300-E-7-1      N      N      N      N
13----- -----  300-E-7-1      N      N      N      N
14----- -----  300-E-7-1      N      N      N      N
15----- -----  300-E-7-1      N      N      N      N

Dial entry #:           ! or...   Enter: R to revise or add to directory
                               M for manual dialing
                               F / B to page through directory
                               X to exit to terminal
                               ! For long distance service, precede entry # with +/-
    
```

Figure II.4 DRTALK showing the dialing directory.

with regard to the computer system with which you plan to communicate.

DR Talk must also be told how to dial the phone number through your modem. Most Hayes-compatible modems use the ATDT command to initiate dialing. This command means "dial with Touchtone." You may want to change this to ATDP, or "dial with pulses" (rotary dial mode). Other modems may use a different command entirely. If you plan to use a long distance service such as MCI or Sprint, the access number can be added to the dialing directory. Two access numbers can be stored and are referred to as -# and +#. To use these in a directory entry, you can put a + or - character ahead of the number and then enter the actual number to dial.

### Defaults and Configuration Parameters

Some of the DR Talk defaults were discussed earlier, in regard to setting the data rate and parity options. The ALT-F Program Defaults menu allows you to further customize your communications.

The echo option lets you set the DR Talk default mode to either echo on or off. Some computers operate in full duplex mode, in which all the characters you send are echoed back to you. If you are communicating

with a computer that uses half duplex, you must echo characters locally in order to see what you are typing. The strip and replace fields allow DR Talk to scan for certain characters and remove or replace them with other characters. This feature can be used to implement a simple translate table. You may also set the default disk drive, the screen margin width, the foreground and background screen colors, and other features having to do with the way DR Talk displays information on the screen. The printer port used for screen dumps and simultaneous printing of the data coming in from the remote computer can be set as well.

All of the values shown in Figure II.3 can be saved to a disk file for later use. Once this configuration information is entered, you won't have to reenter it unless you need to make changes.

The function keys F1 through F10 may be programmed to simplify communications with another computer. For example, your sign-on sequence, including the account number, could be set up under one function key, making the log-on procedure a one-key-stroke operation. All Alt-, Shift-, and Ctrl- function keys may be programmed. These values will be stored in a file called DRTALK.KEY so you won't have to reenter them the next time you use DR Talk. Temporary function keys, ALT-1 through ALT-0, may also be assigned. They may be used to set up frequently entered commands for the current communications session.

---

## A p p e n d i x



# DR EDIX

Concurrent PC DOS is the first operating system to come with a true full screen editor instead of a line-oriented editor like EDLIN, which is included with MS-DOS and PC-DOS. In fact, DR EDIX is a very powerful full screen editor that can be used to create and edit memos, letters, computer programs, and other text files. It also gives you the ability to work with four separate files at once, which makes cutting and pasting operations very simple, and to split the screen into two independent sections so you can see two files at once.

A tutorial is supplied with Concurrent PC DOS that explains all the functions of DR EDIX. This tutorial may be invoked by typing STUDENT from the command prompt. The files required to run DR EDIX and the tutorial are: DREDIX.COM, STUDENT.BAT, LSN.EDX, HELP.EDX, and TEXT.

### Creating a Document

DR EDIX can be loaded for normal operation (not the tutorial mode) by invoking DREDIX.COM from the command prompt or the startup menu, or from the File Manager by selecting the EDIT file option.

Since several buffers and two screens or windows (not to be confused with Concurrent's four windows) are available, the status line at the bottom shows which buffer (B1) and which window (W1) are in use at the moment.

You can create a new document by just typing. All your text goes

into the selected buffer and will stay there until you write the text to a file on disk. If you try to leave DR EDIX without saving the text (with an ALT-W keystroke) you will be asked if you want to lose your changes (or additions). If you save the file with an ALT-W keystroke, you will be asked for the file name, since you are creating a new file.

### **Editing a Document**

Similarly, if you want to edit an existing file or document, you can enter the ALT-R or read command. The file (or the first part if it is large) will be loaded into the current buffer and displayed in the current window.

The cursor can be moved around in the window by pressing the directional arrows on the numeric keypad. Also, the Home, PgUp, and PgDn keys will move the cursor to the first line of the buffer, and will page forward and backward one screen page respectively. Other cursor controls are listed in Figure III.1.

Figure III.2 shows DR EDIX with two windows opened for editing. The ALT-1 and ALT-2 keys are used to select the desired window. Alternately, the F1 and F2 keys can be used to switch from one window to the other. ALT-O (the letter O) will restore DR EDIX to single-window operation. ALT-P will display the status screen, which shows what files are open, which buffers are in use, whether the buffer has unwritten changes, how many lines of text are in each buffer, and whether the buffer has a marked block in it.

Another useful feature of DR EDIX is its undo command. All deleted lines and deleted blocks are stored on the undo "stack." If you press the ALT-U key, the last delete operation will be undone and the text will be restored to the place where it was deleted. Up to 50 nonblank lines can be stored in the stack. If the stack is full and you delete lines, the oldest text is deleted from the stack.

Text lines can be up to 255 characters long, including the end-of-line character. Tabs are treated as just one character, so it is possible to enter a line that is considerably longer than 255 characters, if necessary.

### **Block Commands**

In DR EDIX, you can copy or move blocks from place to place within a buffer or between two buffers. You can also delete blocks in any buffer. The ALT-K key combination is used to mark the beginning and end of a block to be moved, copied, or deleted. Partial lines can't be marked

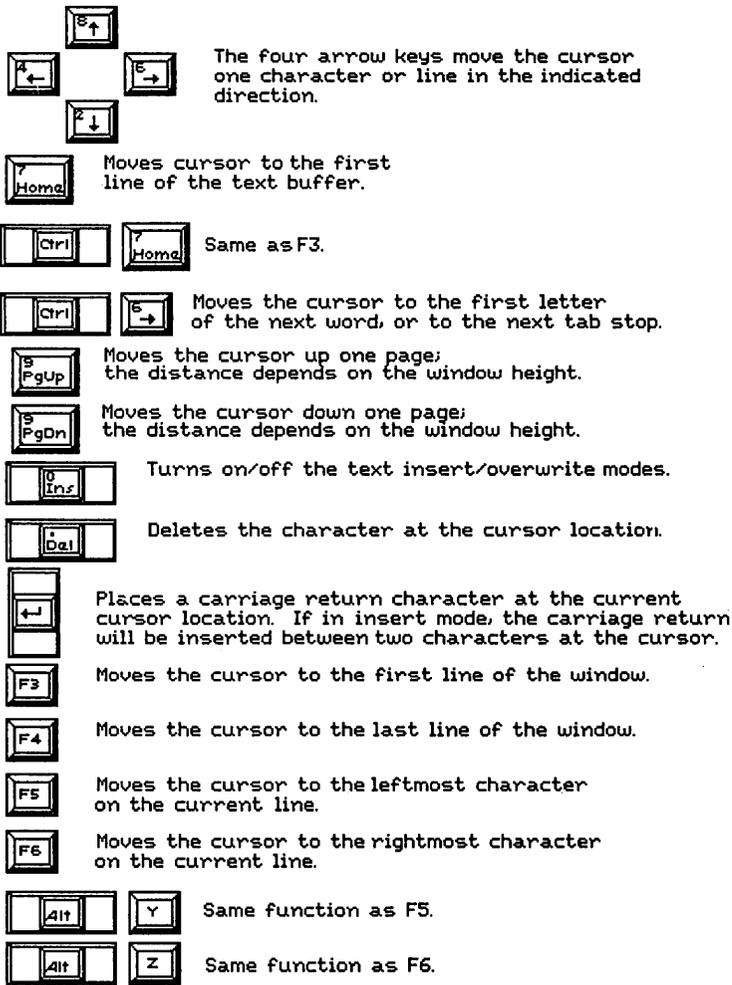


Figure III.1 DREDIX cursor control and editing function keys.

for a block operation. ALT-C will copy the marked block to wherever the cursor is currently located, even if in another buffer. The copy command doesn't reset the block marks, so it is possible to do multiple copies quite easily. When the ALT-M or move command is issued, the text is copied, but the source text is deleted. The block marks follow the copied text, so you can issue another move or copy without having to remark the block.

The block delete or CTRL-D command is equivalent to using ALT-D (line delete) repeatedly. The deleted text is copied to the undo stack, and the block markers are turned off. If more than 50 lines are deleted, the undo stack will not be able to store all of them, so you could lose some text permanently this way.

## Translating and Searching

DR EDIX allows you to search for text strings, and also search and replace strings with a new string. These functions work similarly to most of the popular word processors. Both the search and translate commands can be repeated with one keystroke. The translate command offers four options: local, global, skip, and abort. Local will cause just the current string (at the cursor) to be replaced with the new string. Global will replace all occurrences, skip will move on to the next occurrence, and abort will stop the command altogether.

DR EDIX lets you include wildcard characters in the string for which you are searching. For example, you could include a question mark to tell DR EDIX to accept any character in that position in the string. An asterisk will match any string of any length in a line. Some

```

Mr. Bob Thomas
279 Apple Wood Blvd.
Tarzana, CA 91356

Dear Mr. Thomas:

On Tuesday, March 31, our Tax Department received three shoe boxes
via Express Mail which they assumed contained your tax records.
Instead, the boxes contained the following:

    one pair athletic shoes, powder blue with white swoosh, size 12
    one pair wing tips, brown, size 12 EE
    one pair ladies dress sandals, red patent leather, size 7

We are returning the shoes via registered mail to your home address.
Mr. Thomas, you may want to check with your shoe repair shop to see
if they are in possession of the boxes containing your receipts. If
so, please expedite the records, as April 15 is fast approaching.
  
```

---

```

Read OK                                     LETTER.TXT
                                             B1      1      1/24
  
```

Figure III.2 DREDIX sample screen display.

wildcard examples are:

<i>Search string:</i>	<i>Matching strings:</i>
?og	dog, fog, bog, hog
??-DOS 2.?	MS-DOS 2.0, MS-DOS 2.1, PC-DOS 2.0, PC-DOS 2.1
help*	help, helping, helped, helper
*program*	program, preprogram, deprogram, programmer, programming

It is also possible to search for the actual asterisk and question mark by using an escape character to keep DR EDIX from interpreting them as wildcards. A backslash (\) character followed by the ? or \* character will accomplish this. If you need to search for the backslash character, type two backslashes in a row to override its special interpretation.

# IV

## Application Programs under Concurrent PC DOS

Concurrent PC DOS can be considered a compatible operating system in the same way that the Compaq is an IBM PC-compatible computer. Most programs will run, but once in a while you'll come across one that isn't well-behaved and just won't work. It should be noted, though, that many application programs that work with PC-DOS may not work with IBM's TopView either. In this respect, TopView is not fully PC-DOS compatible, and you may be no better off with it than with Concurrent. Concurrent has the advantage of being a better product than TopView in terms of functionality and performance.

### **PC-DOS/MS-DOS 3.0 Compatibility**

As of November 1985, Concurrent PC DOS Version 4.1 supports all PC-DOS/MS-DOS 3.0 functions, such as hierarchical directories and DOS media. Some of DOS 3.0 functions are not available, such as character device drivers. This makes Version 4.1 of Concurrent a hybrid in that it will run all DOS 3.0 programs and support subdirectories, but it won't handle certain device drivers, and can't cope with pop-up programs such as Sidekick.

### **Using Windows with Application Programs**

Taking into account what was said in the last section, Concurrent should run most programs written for DOS. However, certain pro-

grams may not allow their screen display to be windowed properly, even though windows are supposed to be completely transparent to the application program. This is because some programs (as discussed in Chapters 2 and 4) may write directly to the video display buffer. If this is the case, the program will work properly when used with a full screen window, but may write characters in the wrong places or entirely outside the window if the window is made smaller or moved around on the screen. This will not harm your program or the computer, but may look very strange. In these cases, you will just have to live with a full screen display, but you could set up other windows that overlap the full screen, which could run other applications or Concurrent utilities that do work properly with windows.

### Programs Tested on Concurrent

The following table lists all popular application programs that have been fully tested with Concurrent PC DOS. This list is being continually expanded, but it gives you some idea of the chances that your software will have of working properly under Concurrent.

- dBASE II Version 2.4
- DR Draw Version 1.0
- DR Graph Version 1.0
- General Ledger Version 1.1
- Home Accountant Plus
- MultiMate Version 3.31
- Multiplan Version 1.00
- Lotus 1-2-3
- PFS:File
- PFS:Graph
- PFS:Report
- PFS:Write
- Supercalc 2 Version 1.00
- Supercalc 3 Version 1.00
- Visicalc
- Wordstar Version 3.24

It should be noted that only one program that uses the BasicA interpreter can be run at one time. This is because BasicA uses some fixed memory locations to store control information and reprograms some devices such as the serial ports. Therefore, two copies of BasicA cannot coexist in the Concurrent PC DOS environment at the same time. For example, Home Accountant Plus and General Ledger both use BasicA and could not be run at the same time.

## Other Considerations

Most application programs don't properly release the printer device once they have allocated it. With the programs in the above list, the only way to release the printer device is to stop the program. This allows other concurrent applications to allocate the printer.

Applications that write directly to the screen buffer could cause strange characters to be displayed on the screen if they are run with the SUSPEND OFF option. This option allows programs in windows other than the currently selected one to continue to operate, thereby causing spurious characters to appear on the current window. You can simply select SUSPEND ON (the default) to solve this problem.

The number of programs that can be run concurrently is limited only by the amount of main memory you have installed in your personal computer. About 150K bytes of memory are used by Concurrent PC DOS. The remaining memory can be allocated to programs or to a memory disk. A minimum of 64K bytes are set aside for each of the four windows for programs that have a file extension of .COM. The COMSIZE command lets you change this allocation if necessary. The ADDMEM command lets you change the allocation for applications with the .EXE file type from an initial 16K bytes to whatever the program requires.

### **dBASE II Version 2.4**

dBASE II will operate with a COMSIZE setting of as little as 48K bytes. You may run two copies of dBASE II by setting the SYS (system) attribute on all dBASE program files, DBASE.COM, DBASEMSG.TXT, and DBASEOVR.COM. To run dBASE II under PC-DOS, you will have to use Concurrent to set the SYS attributes of these files back to the normal or DIR attribute. The dBASE II editor program (SED.EXE) requires a minimum setting of 64K bytes (using the ADDMEM command).

### **Lotus 1-2-3**

You can't run multiple copies of Lotus under Concurrent PC DOS at this time. One copy will work correctly though, provided you reinstall it for Concurrent PC DOS. You must keep the 1-2-3 system disk in drive A: for it to work properly, just as PC-DOS requires. To load large spreadsheets, use the COMSIZE command to increase the memory allocation for 1-2-3 and then load 1-2-3 by using the LOTUS.COM program.

### **SuperCalc 3**

Multiple copies of SuperCalc 3 can be run under Concurrent, but you must set a minimum comsize of 128K bytes and set the SYS attribute bits for the SuperCalc program file. If you plan to use both color and monochrome displays with SuperCalc, you must use the Window Manager to move the window running SuperCalc to the color display.

### **Wordstar Version 3.24**

Wordstar works as expected under Concurrent, and you can run multiple copies by setting the F1 attribute bit on the WS.COM program file. No other changes are necessary.

# V

## Summary of Features and Extensions

### PC-DOS Compatibility

- Version 3.2 of Concurrent PC DOS supports all functions available in PC-DOS Version 1.1.
- Version 4.1 of Concurrent PC DOS is fully compatible with PC-DOS Version 3.0.
- Concurrent PC DOS is an enhanced version of Concurrent CP/M-86 Version 3.1 and is able to read and write both CP/M-86 and PC-DOS media. It also supports fixed disk partitions of either media simultaneously.
- Full support is provided for hierarchical directories as used by PC-DOS 2.1 and 3.0. CP/M user numbers are also supported on CP/M media.

### Features

- Multitasking capability with up to four user programs in operation at one time plus a background Printer Manager.
- Up to four user windows can be used to execute programs. Window size and position can be adjusted by the user, and windows may be placed on either the color or monochrome display as desired.
- The File Manager simplifies the operation of your personal computer by placing all of the most frequently used commands in a series of

menus. It also displays the currently selected directory, and the user can select files to be operated on by pointing to various commands instead of typing in file names.

- DR Talk, a communications and file transfer program, is supplied with Concurrent PC DOS. It is functionally identical to the popular PC-TALK III communications program.
- DR EDIX is a full screen text editor/word processor that can be used to create and edit documents including letters, memos, batch files, and program source files. It has four text buffers that can be used to open and edit up to four files at a time. The screen can be split into two sections to see two of the buffers at once.
- The Printer Manager is similar to PC-DOS's PRINT command but supports up to five printers at once. Print jobs may be added to a print queue and will be directed to the desired printer. Options such as number of copies to print, formatting, page size, margins, and page numbering may be set.
- A menu creation and management tool is provided to create your own turnkey systems.
- A Help facility provides the user with online help for all commands and utilities supplied with Concurrent.
- The SYSDISK command lets you select a system disk (either a physical disk drive or a subdirectory within a disk) to search for files if they can't be found in the current disk or directory. This command is superior to the PC-DOS PATH command in that it works for all file references instead of just references to files of the type .COM or .EXE.
- The Window Manager can be loaded and made resident, which allows the user to position windows and change their colors on demand.

---

## A p p e n d i x

# VI

## Concurrent PC DOS Command Reference

**8087** Indicates to Concurrent PC DOS that a .COM or .EXE program uses the 8087 coprocessor. The form of the command is:

8087=ON  
or 8087=OFF

Concurrent checks to see if the 8087 coprocessor is present in your computer. You will need to issue the 8087=ON command in each window that will use the 8087. This command is only used for PC-DOS compatible programs. Programs with the .CMD file type can be set to indicate that the 8087 is used by running the CHSET command.

**ADDMEM** Sets the memory allocation requested by programs that have the .EXE file type. EXE programs often indicate only their minimum memory requirements, since all previous versions of PC-DOS and MS-DOS automatically allocate all available memory to the program. This command applies only to the next command loaded. The best way to use this command is to create a batch file that first invokes ADDMEM and then runs the desired program.

Form: ADDMEM  
ADDMEM=n

**BACK** Backs up the specified files from a fixed disk partition to floppy disks.

Form: BACK  
BACK FULL  
BACK CONTROL=n  
BACK REPORT

The backup utility automates the fixed disk backup process by allowing you to set up control files to indicate which files or directories to back up and which files to ignore. The backup utility will only back up files that have been modified since the last backup. Backup floppy disks are numbered sequentially for easy retrieval of the files.

**BACK** backs up only new files or files that have been modified since the last backup, while **BACK FULL** backs up all fixed disk files that satisfy the parameters set up in the **CONTROL.BR** file. **BACK CONTROL=n** takes control information from file **CONTROLn.BR** instead of **CONTROL.BR**. This allows you to set up multiple control files to back up different groups of files. **BACK REPORT** prints the **BACK REPORT** file.

**BATCH** Executes commands in a file that has the **.BAT** file type. This command is functionally identical to the PC-DOS batch facility.

**CARDFILE** An electronic address book utility that stores names, addresses, and phone numbers on "index cards." The user can search for or scroll through entries and may print one or more addresses or mailing labels.

Form: **CARDFILE**  
**CARDFILE COLOR**

**CHDIR** or **CD** Sets the current directory or shows the path to the currently selected directory. This command is functionally identical to the PC-DOS **CHDIR** command. In Concurrent, this command also lets you set the floating drives **N:** and **O:**.

Form: { } indicates optional parameters  
**CHDIR** {fd:=} {d:} {\  
**CHDIR** {fd:=} {d:} {dirpath}  
**CD** {fd:=} {d:} {\  
**CD** {fd:=} {d:} {dirpath}

The backslash character (\) by itself represents the root or main directory on PC-DOS media. It is also used as a separator between subdirectory names and the file name when used in the path name. For example:

**C:\** represents the root directory of drive C.  
**C:\dir1** represents the subdirectory dir1 on drive C.  
**C:\dir1\dir2** specifies the path to a sub-subdirectory.  
**C:\dir1\dir2\file** gives the path to the actual file.

The **CD ..** command can be used to cause Concurrent to make the previous directory the current one. If the current directory is **\dir1\dir2**, then **CD ..** would set the current directory to **\dir1**.

Floating drives are often useful (and sometimes necessary) when working with Concurrent. Drive letters **N:** and **O:** can be assigned to a disk or a directory in a way similar to the **SYSDISK** command. For example, suppose you want to copy a file from a subdirectory **\dir1\dir2** to another subdirectory **\dir3** using the **PIP** utility. Since **PIP** doesn't support subdirectories, it is necessary to set up floating drives to specify source and destination. One method of performing this copy would be as follows:

```
CD N: =\dir1\dir2
CD O: =\dir3
PIP O: =N: filename.ext
```

**CHSET** Changes the command header of programs with the **.CMD** file type. Program options can be changed to allow programs to run under Concurrent with greater efficiency.

Form: **CHSET [HELP]**  
**CHSET filespec**  
**CHSET filespec [field=setting[, field=setting...]]**

**Fields allowed:**

8087 ON or OFF (program uses 8087 coprocessor)  
 SHARED ON or OFF (program code may be shared)  
 SUSPEND ON or OFF (program may run in background mode)

**COMSIZE** Sets the amount of memory allocated to programs with the .COM file type extension.

Form: COMSIZE  
 COMSIZE=n

**COPY** Transfers files between character-oriented devices and disks. This command works just like COPY in PC-DOS.

Form: COPY source{/option} destination{/option}

Devices allowed: CON:, LPT1:, PRN:, and NUL:.

Options: /A treats file as ASCII text file.  
 /B copies entire file as binary data.  
 /V verifies copied file.  
 /n indicates CP/M-media user number of file.  
 + may be used between files in the source field to concatenate them and send them to the destination as one file.

**COPYMENU** Copies one or more menus from one .DAT menu file to another. This command is part of the menu creation/editing facility.

Form: COPYMENU  
 COPYMENU source-file destination-file -M  
 COPYMENU source-file destination-file menulist

**DATE** Sets and displays the current date. Operates identically to the DATE command in PC-DOS.

Form: DATE {mm-dd:yy}

**DEL** Erases files.

**DELQ** Erases files and file password space, but questions you before each erasure to prevent accidental loss of files you may not want to delete.

**DIR** Displays the files in a directory. This command also shows you what type of media (CP/M or DOS) the disk is formatted for and the remaining space on the disk.

Form: DIR {d:} {filename} {/L} {/S} {/P}  
 where d: is an optional drive specifier.  
 filename is a name to search for (may include wild card characters).  
 /L displays files, sizes in bytes, and time/date stamping if available.  
 /S displays files and directories with the SYS attribute.  
 /P causes display to pause with every full screen.

**DREDIX** Invokes the DR EDIX text editor, which can be used to create and edit any text files including batch files, letters and memos, and program source files.

**DRTALK** Invokes the DR Talk communications program, which may be used to communicate with other personal or mainframe computers. It is functionally identical with the popular PC-TALK III program.

**DSKMAINT** Allows you to format, verify, and copy floppy disks.

**EDITMENU** A menu creation utility that creates, modifies, and deletes menus.

**ERAQ** Identical to DELQ. Erases files and password space but questions you before each erasure to prevent accidental loss of files you may want to keep.

**ERASE** Erases files.

**FM** Loads Concurrent PC DOS's File Manager, which allows you to use most of the commands provided by Concurrent without having to remember the names or syntax of the commands. A series of menus plus a directory display allow you to easily use Concurrent without learning DOS commands.

**FUNCTION** Lets you assign function key text strings and set the window switching key combinations.

Form: FUNCTION

FUNCTION {d:} filename.PFK

You may assign character strings up to 20 characters in length to the keys F1 through F10. Strings up to four characters long may be assigned to any function key combination formed with the Ctrl, Alt, and Shift keys. Strings up to four characters long may also be assigned to any of the dual-purpose numeric keypad keys.

**HDMAINT** Performs fixed disk partition and verify services. You can create, change, or delete a CP/M- or PC-DOS-media partition. Both types of partitions can be tested for errors. The Concurrent loader program can be written to a fixed disk partition to make the fixed disk bootable. You can also display and change a partition's options for directory labels, number of directory entries allowed, and read-after-write verification.

**HELP** Provides an easy-to-use online reference guide to all Concurrent PC DOS commands and features.

Form: HELP

HELP {topic {subtopic}}

HELP [EXTRACT]

HELP [CREATE]

If you enter the name of a topic, HELP will take you directly to the information on that topic, if it exists. The extract option converts the HELP.HLP file into a text file that can be edited on a word processor or with DR EDIX. This allows you to delete or add help topics to the file. The CREATE option lets you create a new help file by reformatting your text file into a HELP.HLP file.

**INITDIR** Reformats CP/M-media disks to allow for time and date stamping of files (as on DOS-media disks).

**LOADCCPM** Loads Concurrent PC DOS from a PC-DOS fixed disk partition. This allows you to continue booting PC-DOS or MS-DOS from your fixed disk and then decide whether you want to invoke Concurrent.

Form: LOADCCPM

LOADCCPM ask

The "ask" option causes the LOADCCPM program to first ask you if you want to load Concurrent. This allows you to put the LOADCCPM command in

your PC-DOS or MS-DOS AUTOEXEC.BAT file so you can decide which operating system you want to run at load time.

**MKDIR or MD** Creates a new subdirectory on DOS-media disks.

Form: MKDIR \dir  
 MD \dir  
 MKDIR dir  
 MD dir

**ORDER** Changes the search sequence for command files, including .COM, .EXE, .CMD, and .BAT files.

Form: ORDER \
 ORDER ext1{,ext2{,ext3{,ext4}}

For example: ORDER COM,EXE,BAT,CMD causes Concurrent to search for a file with a .COM extension first, then .EXE, then .BAT and finally .CMD.

**PIP** The CP/M-86 file copy program. PIP offers many file reformatting options that COPY doesn't support.

Form: PIP destination=source[options]

**PRINTER** Sets the current printer number for a window.

Form: PRINTER  
 PRINTER n

where n is:

- 0 parallel printer (default assignment)
- 1 parallel printer (LPT2:)
- 2 parallel printer (LPT3:)
- 3 serial interface printer (COM1:)
- 4 serial interface printer (COM2:)

**REN or RENAME** Renames a file.

Form: RENAME oldfile newfile

**REST** Restores files backed up by the BACK utility to fixed disk partitions from floppy disks.

Form: REST  
 REST CONTROL=n  
 REST REPORT

**REST CONTROL=n** restores files based on control information taken from the file CONTROLn.BR instead of CONTROL.BR (both of which are created by the BACK command). REST REPORT generates a printed report of the restore operation.

**RMDIR or RD** Removes subdirectories on DOS media. A subdirectory is removed only after all files within it have been deleted.

Form: RMDIR {d:}dirpath  
 RD {d:}dirpath

**RUNMENU** Invokes the specified menu which was created by the EDITMENU utility.

Form: RUNMENU  
 RUNMENU filespec  
 RUNMENU filespec menuname

If no file name is specified, MENU.DAT is assumed. If a file name is given, the first menu in the file will be run, and if a menu name is also included, the specific menu will be run. The menu file must be in the current directory.

**SDIR** Displays detailed directory and file status information.

Form: SDIR  
 SDIR filespec{, filespec}  
 SDIR [option{, option ...}]  
 SDIR filespec{, filespec...} [option{, option}]

When no options are included, SDIR displays all files, including system files, in the current directory. File specifications may include wild card characters.

Options may include the following:

RW	displays only files with the Read/Write attribute.
XFCB	displays files that are password protected.
NONX{FCB}	displays files that don't have a password.
U{SER}=n	shows files in CP/M-media user number area n.
U{SER}=A{LL}	displays files for all user numbers on CP/M disks.
DR{IVE}=A{LL}	displays all files on all drives.
L{LENGTH}=n	shows n lines of filenames before inserting an SDIR table heading.
SI{ZE}	displays the number of kilobytes allocated to files.
FF	sends a formfeed to the printer at the start of the SDIR listing. If the above Length option is used, then a formfeed will be sent after every n lines.
M{ESSAGE}	displays the drive and user number of empty directories, along with the message "File not found".
NOS{ORT}	suppresses sorting on directory listings.
E{XCLUDE}	displays only files that don't match the file names entered on the command line.

**SET** Sets file and disk drive attributes such as time and date stamping and password protection on CP/M media.

Form: SET {options}  
 SET d: [options]  
 SET filespec [options]

**SETPORT** Configures the two serial ports.

Form: SETPORT  
 SETPORT parameter ...

Setport has a menu mode and a command line mode. Serial port options can be set, including data rate, handshaking, protocol, word length, parity, and stop bits.

**SETUP** Allows you to change Concurrent's default characteristics. The Setup menu includes the following items:

- Save System Parameters
- Diskette Head Step Time
- Verify After Diskette Write
- Select Serial Console
- Maximum Memory per Process
- Color Monitor Scroll Mode
- Update and Exit

**SHOW** Displays information about the storage allocation and size of disk drives.

Form: SHOW {d:}  
SHOW {d:} option {,option}

where options include:

SPACE displays remaining space on disk.  
 DRIVE shows drive-recording characteristics, including size of disk, number of directory entries, records per block, sectors per track, and number of reserved tracks.  
 USERS displays the current user number and all user numbers with files recorded in them.  
 LABEL displays the drive's label and password information, as well as the time and date when the label was created.  
 HELP Displays a summary of the SHOW commands.

**STOP** Displays memory allocation and can be used to terminate programs.

Form: STOP  
STOP program-name window-number

**SUSPEND** Turns on or off suspension of .COM and .EXE programs when their windows are not the currently selected one.

Form: SUSPEND=ON  
SUSPEND=OFF

**SYSDISK** Functionally similar to the PC-DOS PATH command, but much more useful. SYSDISK will set and display the path or disk drive letter of the current system disk. This causes Concurrent to check the system disk when a file can't be found on the currently selected disk or directory.

Form: SYSDISK  
SYSDISK {d:} { } {dirpath}  
SYSDISK [OFF]  
SYSDISK [ON]

**TIME** Sets the current time.

Form: TIME  
TIME hh:mm:ss

**TYPE** Displays the contents of a disk file.

Form: TYPE filespec

**USER** Sets the directory user number on CP/M-media disks.

Form: USER n

**WINDOW** Sets and displays window characteristics.

Form: WINDOW command NUMBER=n parameter {,parameter...}  
 Commands: V{IEW} displays each window's current parameters.  
 T{OP} switches to the specified window.  
 F{ULL} displays the complete screen for the specified window.  
 WR{ITE} records the window or entire screen contents in a file.  
 CH{ANGE} modifies a window's parameters.

The CHANGE command lets you set the following parameters:

PRow physical row of upper left corner of window  
 PCol physical column of upper left corner of window  
 NRows number of rows in window  
 NCols number of columns in window  
 Tracking controls cursor tracking in the window

Display	determines whether window is displayed on RGB or mono-chrome display adapter
FGcolor	selects foreground color
BGcolor	selects background color
Virtual row	row of window in upper left corner
Virtual col	column of window in upper left corner

**WMENU** Loads and makes resident the Window Manager. The Window Manager lets you modify window dimensions and colors without having to enter commands to the WINDOW utility. The parameters shown in the WINDOW command can be set and altered from a pop-up menu on line 25 of the display.

---

A p p e n d i x

VII

**Bibliography**

- Concurrent PC DOS Version 4.1 User's Manual*, Monterey, Calif., Digital Research, Inc., 1985.
- Dahmke, Mark: *The Byte Guide to CP/M-86*, New York, McGraw-Hill, 1983.
- : *Microcomputer Operating Systems*, New York, McGraw-Hill, 1982.
- IBM DOS User's Manual*, IBM Corporation, 1984.
- Norton, Peter: *Inside the IBM PC: Access to Advanced Features and Programming Techniques*, Bowie, Md., Brady, 1983.
- Waite, Mitchell, et al.: *DOS Primer for the IBM PC & XT*, New York, New American Library, 1984.
- Wolverton, Van: *Running MS-DOS*, Bellevue, Wash., Microsoft, 1984.

---

## A p p e n d i x

# VIII

## Glossary

**.BAT** The file type used for a batch file.

**.CMD** See command file.

**.COM** In the CP/M operating system, a .COM file is an object or binary file containing an executable program. .COM refers to a command file or to a file that can be executed by typing only the first part of the filename.

**.EXE** The file type used for DOS-compatible executable programs.

**.HLP** The file type used for the HELP file text.

**address bus** The portion of the system bus that carries the address of the memory or input/output port to be accessed.

**address space** The address space of a computer is a function of the addressing range. If a microcomputer has a 20-bit address range, the address space may be up to 1M byte in size.

**argument** A placeholder for a number, letter, or name that is being passed from one level of a system to another. For example, a command can have several arguments that are being passed symbolically from the command level to the command itself for option selection or decision making.

**ASCII (American Standard Code Version 2)** A standard code for representing alphabetic, numeric, and special characters on a computer.

**ASCII control character** A group of nonprintable ASCII characters, such as carriage return or line feed, used to control a peripheral device or send messages from one machine to another. For example, STX—Start of Text, ESC—Escape.

**assembler** A program that reads mnemonic instructions written for the machine instruction level of a computer and converts them to the operation codes that the computer can read directly.

**asynchronous data transmission** A form of data transmission where each byte may be sent at any time, with no special timing between bytes, but with strict timing within the byte. Used mostly for low data rate applications such as dial up terminals.

**attribute** A file characteristic that can be either on or off. File attributes in DOS and CP/M include: DIR, SYS, RO, and RW (userfile, systemfile, read-only, read-write).

**backup** A copy of a disk or a file made for safe keeping.

**background activity** Another name for low-priority programs running on a multiprogramming system. Since these programs have low priorities, they are said to execute in the "background."

**Basic Disk Operating System (BDOS)** The heart or nucleus of Concurrent. This portion of the operating system performs all DOS services, such as task switching and scheduling, file input and output, and console input and output.

**batch programming** A method of executing programs where each program is read into the system in order and executed sequentially, as opposed to a timesharing or multiprogramming system where multiple users have concurrent access to the system and can run different programs at one time.

**bit** A single binary digit, which may be either one or zero.

**boot-load** The action of loading the operating system or monitor into the computer after a cold start power up. The boot-load program is stored in programmable read-only memory so that it is present in the system on power up.

**bootstrap** See boot-load.

**buffer** A block of memory set aside to hold incoming or outgoing data.

**byte** A unit of data, consisting of 8 bits.

**change directory (CHDIR)** A command that changes the current directory path.

**character string** A sequence of data bytes that represents characters in ASCII, EBCDIC, or some other code.

**cold start** The act of starting up a computer after power on. The computer has no programs in memory and must be loaded from scratch. See boot-load.

**command** A sequence of characters entered either at the DOS console command processor level or to a utility program, such as DR EDIX, or a user program. The command usually consists of a keyword that tells the program what you want to do and optional arguments.

**command file** A file that contains executable machine instructions.

**compiler** A program that accepts as input a series of high-level commands and converts them to machine-level instructions that the computer can read.

**CON:** The logical console device.

**concatenate** An operation that combines two or more disk files and writes the result to a third file. COPY can be used to concatenate files.

**console** The primary input/output device. The logical console device may be connected through the BIOS to a physical device, such as a video display terminal (VDT) with a keyboard and text display, or to another device, such as a serial port.

**controller** A hardware circuit that controls access to a peripheral device, such as a disk drive.

**COPY** A command used to make copies of files and move data between logical devices.

**control characters** ASCII characters that provide control information to a device or the computer. On input, control characters can be entered by holding down the Ctrl key and pressing the desired alphanumeric key. The Carriage Return and Backspace keys are examples of control characters.

**CP/M (Control Program for Microcomputers)** An operating system from Digital Research that runs on 8080- or Z80-compatible microprocessors. CP/M-86 will operate on the 8086 and 8088 16-bit microcomputers.

**CPU (Central Processor Unit)** The portion of the computer that controls all operations of the arithmetic logic unit and the interface between the computer and the user.

**cursor** A one-character symbol that appears on the screen of the console where the next character to be sent from the keyboard or computer will be displayed.

**data bus** The portion of the system bus that carries the data that is to be read or written to the main memory or input/output ports.

**debugger** A program that assists the user in testing and tracing errors in an application program.

**default drive** The disk drive that has been selected as the current drive. All disk file operations are performed (or attempted) on the default drive if no disk drive specifier is included in the file name.

**delimiter** A special character, such as a comma, that separates different items in a command line; for example, a period separates the file name from the file type.

**device assignments** Logical to physical assignments that allow peripheral devices to be traded as needed for an application without having to be physically disconnected and reconnected.

**directory** The portion of a floppy disk that contains entries for each file on the disk. The DIR command will display all the entries on a particular disk.

**direct memory access** An action performed by a peripheral device that allows it to gain direct access to the main memory of the microcomputer. When the peripheral initiates direct memory access, the microprocessor is forced to stop all bus activity while the peripheral occupies the bus. This method allows extremely high data transfer rates at the expense of the throughput of the microprocessor.

**disk file** A logical grouping of information on a disk. A file may contain text, a program, or any other information in any form.

**disk formatting** A procedure that writes blank sectors of data to a disk, initializing it for later use.

**disk initialization** See formatting.

**DIR attribute** This file attribute specifies that a particular directory entry will be listed as a regular file, not as a system or invisible file.

**downloading** A way of transferring programs or data from a large computer system to a microcomputer. This allows program development or data acquisition to take place on the large computer while program execution may take place on the microcomputer.

**DOS (Disk Operating System)** An operating system based primarily on disk storage capabilities.

**EBCDIC (Extended Binary Coded Decimal Interchange Code)** Another common code for representing alphanumeric and special characters on a computer.

**editor** A utility program that creates and modifies text files. The DR EDIX editor can be used to enter programs or data.

**EPROM (Erasable Programmable Read-Only Memory)** Similar to a PROM, but erasable when held under a short wave ultraviolet light.

**FCB (File Control Block)** A block of data used to represent the name and physical attributes of a disk file. Usually stored on disk in a directory of files, but copied to memory when the file is in use.

**file name** The name assigned to a file. The file name portion of the file specification usually refers to only the first eight characters, while the file type refers to the three-character extension of the file name.

**file specification** A unique file identifier. Sometimes referred to as the "filespec" or even just "filename," it includes optional file name, file type, and disk drive specifiers.

**file type** The extension of a file specification. The file type can be from one to three characters and must be separated from the file name by a period. The file type usually tells something about the contents of the file; for example, CMD stands for "command file," EXE refers to a DOS-compatible executable program.

**fixed disk** See hard disk.

**floppy disk controller** The hardware or circuit board used to interface the floppy disk drive to the microcomputer bus.

**foreground activity** Activity taking place at a higher priority in a multiprogramming system.

**formatting** The process of clearing off a floppy or fixed disk and writing fresh identification information onto the disk.

**function keys** On the IBM PC and compatibles, the function keys are provided to allow a program to have unlabeled keys that are context-dependent.

**hardware** The physical circuitry that makes up the computer. This can include the central processor, the arithmetic unit, the main memory, disk drives, and so on.

**hard disk** A type of disk that is made from a rigid piece of metal, can store from 5M to over 200M bytes, and has a much faster access time than floppy disks. Also called a fixed disk.

**hexadecimal** A number system based on 16. The digits are represented as: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F.

**I/O** Input/output.

**input** Data going into the computer, usually from the console, or serial interface, although it can refer to any input operation on the computer.

**instruction pointer** The internal CPU register that holds the address of the current machine instruction.

**instruction register** The internal CPU register that holds the current machine instruction to be executed.

**interface** The hardware that connects one piece of equipment to another. "Interface" can also refer to the logical connection of two programs or data files.

**interleaving** A method of improving disk access time by skewing or shifting sector addresses around the disk. If the operating system reads a sector and then attempts to read the next sequential sector, the disk will generally have rotated out of position and will have to complete one revolution before it will be in position again. Interleaving involves a logical to physical mapping of sectors so they are offset around the disk, allowing the operating system time to process each sector before reading the next one. This saves a great deal of time because the computer does not need to wait through a whole revolution for each sector access.

**interpreter** A program that accepts as input a series of high-level commands and executes them as they are received rather than saving them as machine instructions.

**interrupt** An external signal that causes the microprocessor to interrupt what it was doing and execute a series of instructions called the interrupt

handler. An interrupt may be generated by a disk drive when an action is completed, or by an incoming character at a serial port. The use of interrupts allows the microprocessor to handle events as they come, rather than having to repeatedly “poll” or search for incoming data.

**IPL (Initial Program Load)** See boot-load.

**jump vector** A jump or branch instruction that transfers control to a specific subroutine or function.

**kilobyte** 1024 bytes or 1K byte. Since all computer addressing is done in powers of two, it is convenient to define “K” as 1024 rather than 1000, since 1024 is a power of two.

**list device** An output-only device, such as a line printer, to which encoded characters (usually ASCII) can be sent in order to get hard copy.

**logical device** A device that is known by name in the operating system, such as CON (the console) or PRN (the printer).

**machine independence** A software design philosophy allowing programs written on one type of computer to be run without changes on another type of computer.

**mass storage** Storage external to the microcomputer, such as a disk drive or tape drive. Generally this term is used in reference to fixed disk technology.

**megabyte** 1024 kilobytes or  $1024 \times 1024$  (1,048,576) bytes. Rather than defining one megabyte as an even million, it is defined as the nearest power of 2, or 1,048,576, to make division by two easy. See kilobyte.

**memory** Any of several forms of storage, including RAM, PROM, disk, tape, or any magnetic media.

**monochrome display** A video display that accepts only black and white signals from a display adapter board. The screen display may actually appear in green or amber, but no specific color information is sent to the display.

**multiprogramming operating system** An operating system that executes multiple programs concurrently.

**nucleus** The part of an operating system that is the center of all control and resource management activities. Whenever the application program requires service from the operating system, it must communicate its desires to the nucleus. The nucleus of Concurrent PC DOS is called the BDOS, or basic disk operating system.

**operating system** Any program or group of related programs whose purpose is to act as intermediary between the hardware and the user. The operating system’s main job is to manage resources such as disk drives, printers, and other peripherals, thereby freeing the programmer from having to rewrite

certain commonly used functions for each application. In this way, the operating system provides a uniform, consistent means for all user-written software to access the same machine resources.

**output** Data that the processor sends to a disk or some logical device, such as a console or list device.

**parallel interface** A peripheral interface to the CPU via the system bus that sends or receives data in parallel from another computer or device.

**parameter** The value in a command string that provides additional information to the program being called. A required element of a command.

**partition** A memory segment in a multiprogramming system into which an application program can be loaded.

**path** A pointer to a unique directory or subdirectory on a disk.

**peripheral** Any device external to the CPU, such as input or output devices, disk drives, or tape drives.

**peripheral devices** Devices that are external to the computer. Printers, terminals, and disks are peripherals, while main memory is not.

**physical devices** Physical hardware attached to the computer through serial or parallel input/output ports, such as printers or terminals.

**pointer** A storage location or register that contains a memory address that may point to a specific location in memory.

**priority** The active status given a program in a multiprogramming system. The priority of a program determines how much CPU time (how many time slices) it will get.

**priority scheduling** A method of task dispatching that is based on a priority scheme. Each program is given a priority and executed on the basis of what other programs have higher priorities.

**PROM (Programmable Read-Only Memory)** An integrated circuit that stores data when programmed with high-voltage pulses and cannot be erased.

**RAM (Random-Access Memory)** Memory that can be randomly accessed as necessary to retrieve data. This is the type used for the main memory of most computers.

**read-only** An attribute commonly used to refer to a file or disk that has been marked by the operating system so that it cannot be overwritten. The read-only attribute bit in the file control block may be set for each file on a disk. A whole disk can be set to read-only as well.

**read/write** A file or disk that can be read or written without any special action. The read-write bit of the file control block can be set to indicate that a file can be read or written by anyone. This is the normal state of a file control block.

**record** A collection of data on a disk. A logical record in CP/M is 128 bytes

long. It is mapped into the physical sectors of a disk even if sectors are longer than 128 bytes.

**root directory** The primary directory on a DOS-media disk. All other directories are attached to this directory.

**ROM (Read-Only Memory)** A kind of memory circuit that can only be programmed when the circuit is manufactured.

**RO** See read-only.

**RW** See read/write.

**scrolling** The act of moving the text lines displayed on a video monitor up one line, so as to simulate the action of a teletypewriter.

**sector** A portion of a track on disk, usually 128 bytes long.

**serial interface** A peripheral interface to the CPU via the system bus that sends and receives data serially from another computer or terminal.

**software** The program or machine instructions that the central processor interprets to perform arithmetic and data movement functions on the hardware.

**source file** An ASCII text file that contains source statements for a high-level language, such as Basic or Pascal.

**subdirectories** Directories that are attached to the root directory or to another directory closer to the root.

**subroutine** A program that can be called from another program. When the subroutine is completed, control is returned to the calling program.

**subroutine library** A collection of frequently referenced subroutines that can be loaded with an application program.

**syntax** The format for entering a command.

**syntax error** An error in the format of a command line, either in the command processor or other programs or utilities.

**system attribute** A file attribute that can be given to any file control block on disk. This attribute tells Concurrent PC DOS that the file is a system file and it shouldn't be displayed in a directory listing. However, the file itself is no different than any other file.

**system bus** The group of parallel data, address, and control lines that connect the CPU to the rest of the computer system, including memory and input/output ports.

**system utilities** Programs that assist the user in performing certain house-keeping functions on the system.

**task** Any program that runs on a computer may be considered a task, but usually the term is reserved for programs running on multiprogramming systems.

**terminal** See console.

**terminal emulation** A program that, when run on a microcomputer system with a video display, will emulate a nonintelligent terminal, allowing the user to dial up another computer system. The purpose of this feature is to allow access to other computers without having to reconnect cables or rearrange a system.

**time slice** The period of time allocated to each program or task in a multiprogramming system.

**timesharing** A form of multiprogramming oriented to multiple users at console devices.

**transient program** A program that is used only briefly or only once in a great while; it is not resident in the operating system.

**uploading** A way of transferring programs or data from a microcomputer system to a host or timesharing system. The data being transmitted is sent via serial data link (usually a telephone) to another computer system.

**user number** The number assigned to a file in a CP/M-media disk directory that tells Concurrent which user area the file belongs to. Since up to 15 users are allowed, each user can have a separate directory display that doesn't contain the files of other users. Actually, the directory entries for all users are intermixed, but the user number lets Concurrent sort them out for each user.

**utility** A program that enables the user to perform commonly used operations such as copying, deleting, or editing files, and so on. Utilities are a convenience for the user, and are provided so the user doesn't have to write them all from scratch.

**VDT (Video Display Terminal)** A terminal that uses a video display rather than a printer to display text.

**warm start** The act of restarting a computer after it has been running. This may be necessary because of program or hardware failure.

**wild card characters** Special characters that match certain fields in a file name. In Concurrent PC DOS there are two wild card characters: \* and ?. The question mark (?) can be substituted for any single character in a file name or file type, and the asterisk (\*) can be substituted for the file name, file type, or both. By placing wild card characters in file specifications, the user can create an ambiguous file name that can quickly reference one or more files (when using utilities that support wild cards, such as COPY and DIR).

**XON-XOFF protocol** A standard protocol used when information is being transferred from one computer to another. This protocol usually requires a full duplex data link. When the receiving end can no longer accept data (if it has something else to do and can't wait for data to come to it), it can send an XOFF ASCII control character that will tell the sender to wait until an XON is sent.

**window** A virtual “screen” used by a program to display information. Windows are mapped to the real screen depending on many factors, such as which program is currently in the foreground and how the user has set up the window dimensions and locations.



# Index

- ADDMEM command, 117, 121
- Assign Function Keys, 58
- Asterisk (wild card), 47, 114
  
- BACK command, 84, 121
- BACK CONTROL=n command, 85, 121
- BACK REPORT command, 86, 121
- Background colors, 91
- Background task, 16
- Backslash key, 44, 122
- Backup file, 83
- BASICA, 97
- BAT, 10, 89, 122
- BATCH command, 90, 122
- Batch files, 89, 122
- Baud rate, 103, 105
- BG, 91
- Binary data transmission, 106
- Built-in command, 48
  
- CARDFILE command, 92, 98, 122
- Cardfile commands, 99, 100
- CCPM.SYS, 90
- CD command, 122
- Change Directory, 44, 122
- Change Drive command, 47
- CHDIR command, 122
- CHSET command, 122
- CMD, 122
- CMD file extension, 122
- CMD programs, 122
- COM, 117
- COM file extension, 117
- Command files, 117, 122
- Command line prompt, 37
- Command panel, 40
  
- Command summary, 121
- Communications, 103
- Communications protocol, 106
- COMSIZE command, 117, 123
- Concurrency, 10
- Concurrent command summary, 121
- Control character commands, 23
- CONTROL.BR file, 84, 85
- COPIES=n, 80
- COPY command, 51, 53, 123
- COPYMENU command, 123
- CP/M media, 34, 45, 85
- Ctrl commands, 23
- Current directory, 45
- Current drive, 37
- Current printer, 79
- Cursor, 42
- Cursor control keys, 42
- Cursor movement keys, 42
- Cursor position, 42
- Cut and paste, 28
  
- Date and time stamping, 66, 126
- DATE command, 123
- dBASE II, 117
- Default colors, 91
- Default printer, 125
- DEL command, 123
- Delete files command, 123
- DELQ command, 123
- Device ownership, 13
- Dialing commands, 107
- Dialing directory, 107
- DIR, 123
- DIR attributes, 57
- DIR command, 123
- DIR options, 123

## Directory:

- current, 37, 122
- root, 44
- Directory path, 45
- Dirname, 45, 122
- Dirpath, 45, 122
- Disk format, 35, 45, 85, 123
- Diskette, types, 35, 45, 85
- DOS media, 34, 45, 85
- DR Draw, 116
- DR EDIX, 96, 110, 120, 123
- DR Graph, 116
- DR Talk, 96, 103, 120, 123
- DR Talk dialing directory, 107
- DREDIX command, 123
- Drive:
  - current, 37, 122
  - floating, 38
  - select, 37
  - speed, 38
- Drive attributes, 126
- Drive designation, 35
- Drive selection command, 47
- Drive specification, 35
- DRTALK command, 103
- DSKMAINT command, 96, 123
- DSKMAINT.CMD, 123

- Edit a file command, 110
- EDITMENU, 67, 96, 124
- EDITMENU command, 67, 96, 124
- EDITMENU.CMD, 123
- Editor, 110
- 8087 command, 121, 123
- ERAQ command, 124
- ERAQ.CMD, 124
- ERASE command, 124
- EXE, 117
- EXE file extension, 117

- FG, 91
- File attributes, 122, 123
- File Directory menu, 44
- File extension, 47
- File Manager, 40
- File Manager Main menu, 41
- File specifications, 47
- Filename, 47
- Filespec, 35

## Files:

- backup, 84, 121
- batch, 10, 89, 122
- copy, 51, 53, 122, 123, 125
- delete, 41
- display, 41, 52, 127
- edit, 96, 110, 120, 123
- erase, 41
- password protected, 35, 93
- print, 11, 49, 120
- program, 117, 122
- recover, 125
- rename, 125
- restore to hard disk, 125
- subset, 46
- type, 49, 52, 127
- Fixed disk, 35
- Floating drives, 38, 54
- Floppy disks, 35
- FM command, 40, 124
- FM.CFG, 91
- Foreground, 91
- Foreground color, 91
- Format diskette, 34, 45, 85, 123
- Format Diskette command, 123
- Format hard disk, 123
- Formfeeds, 81
- FORMSIZE=n, 81
- Free memory, 55
- Free Memory command, 55
- Full screen, 19
- FUNCTION command, 96, 124
- Function key assignments, 42, 57, 92
- Function keys, 40, 41, 111, 112
- FUNCTION.CMD, 124
- General Ledger, 116
- Global replacement, 113
- Hard disk, restore files to, 84, 86, 125
- HDMAINT command, 96, 124
- HDMAINT.CMD, 124
- HELP, 48, 93, 124
- HELP command, 48, 93, 124
- Help facility, 48, 120
- HELP.CMD, 48, 93, 124
- HELP.EDX, 110
- HELP.HLP, 93
- Home Accountant Plus, 116
- HOWTO?, 49, 51

- INITDIR command, 124
- Job queue, display, 79
- Keyboard commands, DR Talk, 105
- KEYS.PFK, 58
- Lines per page, 81
- LOADCCPM command, 90, 124
- LOADCCPM.COM, 90, 124
- Local Attributes command, 57
- Logical device, 35
- Lotus 1-2-3, 116, 118
- Make Directory command, 125
- MARGIN=n, 81
- Memory disk, 38, 93
- Menu(s), 67, 68, 93, 120
  - copy, 75
  - create, 67
  - display, 67, 96, 125
- Menu editing, 67
- Menu files, 67, 68, 93
- Menu names, 68
- MENU.DAT, 126
- MKDIR command, 125
- Multimate, 116
- Multiplan, 116
- Multiprogramming, 10
- Multitasking, 119
- Multiuser support, 10, 94
- NOFEED, 81
- Object panel, 40
- ORDER command, 125
- PAGESIZE=n, 81
- Paper length, 81
- Password protection, 35, 93
- Passwords, 35, 93
- Path, 35, 122
  - CHDIR, 122
  - directory, 35, 122
  - display current, 122
- PC DOS media, 34, 45, 85
- PC Talk, 96
- PFS: series, 116, 117
- PIP, 122, 125
  - intersubdirectory transfers, 122, 125
- PIP command, 122, 125
- PIP options, 125
- Port configuration, 15, 95, 126
- Port numbers, 96
- PRINT command, 11, 49, 120
- Print Files, 76, 119, 120
- PRINTER command, 125
- Printer defaults, 125
- Printer Manager, 76, 119, 120
  - options, 81
  - start, 76, 79
  - stop, 82
  - suspend, 81, 82
- Printer numbers, 79
- PRINTER=n, 79
- Printing commands, 76, 119, 120
- PRINTMGR command, 76, 79, 92, 96
- Program files, 117, 122
- Prompt panel, 40
- Prompts, 40
- RD command, 125
- Read/Write, 126
- Remove Directory command, 125
- REN command, 125
- RENAME command, 41, 48, 125
- Rename files, 125
- REST command, 84, 86, 125
- REST CONTROL=n command, 86, 125
- REST REPORT, 86, 125
- Restore command, 125
- RMDIR command, 125
- Root directory, 122
- RUNMENU, 67, 96, 125
- RUNMENU command, 67, 96, 125
- Screen areas:
  - message, 40
  - prompt, 40
  - status, 40
  - text, 40
- SDIR, 126
- SDIR command, 126
- SDIR options, 126
- Search command, 113

- Search path, 90, 120, 127
- Serial device, 15, 94, 96, 97
- Serial port, 15, 63, 94, 97
- Serial terminal, 15, 96, 97, 104
- SET command, 126
- Set Directory 1 command, 54
- Set Directory 2 command, 54
- SETPORT command, 96, 126
- SETPORT parameters, 126
- SETPORT.CMD, 126
- Set Up System command, 56
- SETUP command, 56, 96, 126
- SETUP.CMD, 126
- SHOW command, 126
- Size/Date ON, 61
- Special characters, EDITMENU, 72
- Special effects, EDITMENU, 71
- Startup files, 92
- Startup Menu, 94
- STARTUP1.BAT, 92
- STARTUP2.BAT, 92
- STARTUP3.BAT, 92
- STARTUP4.BAT, 92
- Status area, 40
- Status display, 40, 76
- Status line, 40
- STOP command, 55, 127
- Subdirectories, 35, 122
  - as floating drives, 122
  - remove, 125
- Subset of Files command, 46
- SuperCalc, 117, 118
- SUSPEND command, 104, 117, 127
- SYS attribute, 57
- SYSDISK command, 90, 120, 127
- System attributes, 57
- System Attributes command, 57
- System defaults, 60
- System disk, 38
- System drive, 90, 120, 127
- System Setup command, 56
  
- Test files, 110
- Text pattern, 113, 114
- TIME command, 127
  
- Timesharing, 11
- Translate command, 113
- TYPE command, 52, 127
- Type Files command, 49
  
- USER command, 127
- User number, 35
  - change, 44, 127
  - current, 44, 127
  - display, 44, 127
- USER.TXT, 46
  
- Virtual Scroll point, 24, 25
- VISICALC, 117
  
- Wild card characters, 47
- Wild cards, 47, 114
- WINDOW, 24, 127
- WINDOW CHANGE command, 25, 127
- WINDOW command summary, 127
- Window Manager command, 26, 120
- Window numbers, 24
- Window parameters, 24
- WINDOW VIEW command, 24
- Windows:
  - change size, 27
  - configure, 25
  - display parameters, 24
  - full screen, 116
  - move, 25, 27
  - occluded, 18
  - save contents, 25, 28
  - switch, 22
  - switching keys, 22, 60
  - tiled, 19
- WMENU, 26, 128
- WMENU command, 26, 128
- WMENU command summary, 127
- WordStar, 117, 118
- WordStar files, 28, 117, 118
- WSETUP.BAT, 25
  
- XMODEM protocol, 106

*Picture it...*

*Window 1* — you're running a 45-minute database sort

*Window 2* — you're pulling a set of numbers from a spreadsheet file

*Window 3* — you're "pasting" these numbers into a word processing document

*Window 4* — you're in constant touch with incoming electronic mail

## USING CONCURRENT PC DOS

...OR — how to turn your PC into a powerful and friendly genie that does windows — 4 at a time!

Everything you need to know to operate this new operating system from Digital Research is right here in this plain English manual.

Everything from the whats, hows, and whys of the system to its 5 applications programs and how to put them to work in your everyday computing... file manager ★ cardfile ★ word processor (DR Edix) ★ printer manager ★ and communications routine (DR Talk).

Everything including creating new menus for accessing programs and issuing DOS or CP/M commands from a menu tree or command line, as you choose!

*Picture it — your productivity will soar!*



The author, Mark Dahmke, is a contributing editor for BYTE magazine. He is also owner of MCD Consulting and has acted as consultant to the Nebraska Telecommunications and Information Center, the University of Nebraska, IBM, and the Nebraska Videodisc Group. He authored two previous BYTE books: MICRO-COMPUTER OPERATING SYSTEMS and THE BYTE GUIDE TO CP/M-86.

Cover Design: Abner Graboff



**McGraw-Hill Book Company**  
Serving the Need for Knowledge  
1221 Avenue of the Americas  
New York, NY 10020

ISBN 0-07-015073-7