Computing Fundamentals PC-DOS & MS-DOS®

Craig A. Wood

Stephen F. Austin State University,
Nacogdoches, Texas

Addison-Wesley Publishing Company

Reading, Massachusetts • Menlo Park, California • New York
Don Mills, Ontario • Wokingham, England • Amsterdam • Bonn
Sydney • Singapore • Tokyo • Madrid • San Juan

This book was produced by the Addison-Wesley Electronic Production Department on an Apple Macintosh II with PageMaker. The output was generated on an Apple LaserWriter II NTX.

This book is in the Addison-Wesley *Computing Fundamentals Series*.

Series Editor: William S. Davis

Library of Congress Cataloging-in-Publication Data
Wood, Craig A.
Computing Fundamentals : PC-DOS & MS-DOS / by Craig A. Wood.
p. cm.
Includes index.
ISBN 0-201-19822-3
1. PC DOS (Computer operating system) 2. MS-DOS (Computer operating system) I. Title.
QA76.76.O63W656 1990
005.4'46--dc20 89-34550
CIP

Reprinted with corrections June, 1990.

EFGHIJ-AL-943210

The Computing Fundamentals Series

The books in Addison-Wesley's *Computing Fundamentals Series* feature tutorials that teach the reader how to use specific software packages. Their low selling price makes them attractive as self-teaching aids and as supplements to a primary text. A unique feature entitled "What can go wrong?" anticipates problems, suggests a cause, and helps the reader recover. The modular nature of the series should prove attractive to the instuctor of a microcomputer applications course. Current and planned titles include:

AutoCAD®
Concepts, Second Edition
dBASE III PLUS™
dBASE IV™
Lotus® 1-2-3®
Microsoft® Word
PageMaker®
PC-DOS and MS-DOS®
UNIX®
WordPerfect 5.0®

Dedicated

To my family Ellen, Deron, and Ryan who make it all worthwhile.

To my parents Nelson and Louise Wood who have been a great inspiration to me.

To my brothers Britton and Loren who have shared a lifetime of friendship with me.

Preface

Computing Fundamentals: PC-DOS & MS-DOS is a part of Addison-Wesley's *Computing Fundamentals Series*. Its purpose is to teach you DOS for use on your personal computer and to provide you with a reference guide to DOS commands and procedures. Except for some familiarity with a keyboard, this book assumes no prior experience with a personal computer.

Computing Fundamentals: PC-DOS & MS-DOS will help you to become a more effective personal computer user. It provides you with a summary of commonly used DOS commands and procedures. It is designed to assist you in performing DOS tasks such as preparing disks, viewing information on disks, and manipulating files of information.

Like other books in this series, *Computing Fundamentals: PC-DOS & MS-DOS* is organized as a set of tutorials. Tutorials often assume that the reader will follow the instructions perfectly and that nothing will go wrong. Such assumptions are unrealistic; in the real world, something always seems to go wrong, and nothing is more frustrating to a beginner than hitting a dead end. Appearing throughout this book is a unique feature entitled "What Can Go Wrong?" that anticipates problems, suggests a cause, and shows how to recover. You should find this feature very useful. In addition, this book contains a summary, self-test questions, and exercises at the end of each chapter to help reinforce the concepts and procedures covered in the chapter.

The information contained in this book is based upon the contents of *PC-DOS & MS-DOS: A Ready Reference Manual*. However, the presentation of DOS commands and procedures and the items indicated above are unique to this book. Some material in the first two sections ("What Is a Computer System?" and "Communicating with a Computer System") of Chapter 1 is adapted (with permission) from Chapter 6 of *Computing Fundamentals: Concepts* (Second Edition) by William S. Davis.

Equipment Utilized

The material presented in this book covers all versions of IBM PC-DOS and MS-DOS. While the research for this book was done on a variety of personal computers, including IBM, Texas Instruments, Compaq, and Zenith personal computers, the procedures outlined in this book were performed on an IBM PC AT using IBM PC-DOS Version 3.30. The IBM PC AT contained one megabyte of memory, a 1.2MB diskette drive (drive A), a 360KB diskette drive (drive B), and a 30MB hard disk drive (drive C). PC-DOS and MS-DOS Versions 2.10, 2.13, 3.20, and 3.30 were used in the development of the material in Appendix B.

Acknowledgments

I prepared this manuscript using WordPerfect. The finished text, including most of the art, was produced with PageMaker. For the development of this book I owe much to the individuals who influenced the writing of *PC-DOS & MS-DOS: A Ready Reference Manual*. These individuals include: Warren J. Boe, University of Iowa; Orlynn R. Evans, Stephen F. Austin State University; Catherine Garrison, The Softwerks; Robert M. Goldhamer, National Education Corporation; Fred C. Homeyer, Angelo State University; Deborah R. Lafferty; Karen E. Leath; Richard L. Robertson, Stephen F. Austin State University; Arthur A. Strunk; and Sharon A. Williams.

I thank the following individuals, who reviewed this manuscript and provided helpful comments and suggestions: David Hata of Portland Community College and Deborah R. Lafferty of Addison-Wesley. I thank Orlynn R. Evans and Richard L. Robertson, computer science faculty members at Stephen F. Austin State University, for their comments and technical assistance. I thank Paul R. Blackwell at Stephen F. Austin State University for his assistance in drawing Fig. 3.2. I also thank Karen E. Leath for her assistance in the manuscript preparation and for her comments and suggestions resulting from her review of this manuscript.

Special thanks go to my editor Keith Wollman, who provided me with valuable support and direction during the writing of this book. Thanks Keith.

Craig A. Wood
Nacogdoches, Texas

Contents

1

Introduction to DOS

The material in this chapter:

- reviews basic computer concepts and terminology
- introduces DOS concepts and terminology

What Is a Computer System?

If you looked up the word "computer" in a dictionary, you might find a definition similar to the following: A **computer** is a device that is capable of solving problems or manipulating data by accepting data as input, performing prescribed operations both mathematical and logical on the data, and then supplying the results of these operations. Such a definition, however, is incomplete. The material in this section gives a more complete definition of a computer using the term "computer system."

Hardware and Software

The personal computer pictured in Fig. 1.1 consists of several components. The keyboard is the primary input device. A video display unit, or screen, and a printer are common output devices. Most personal computers contain

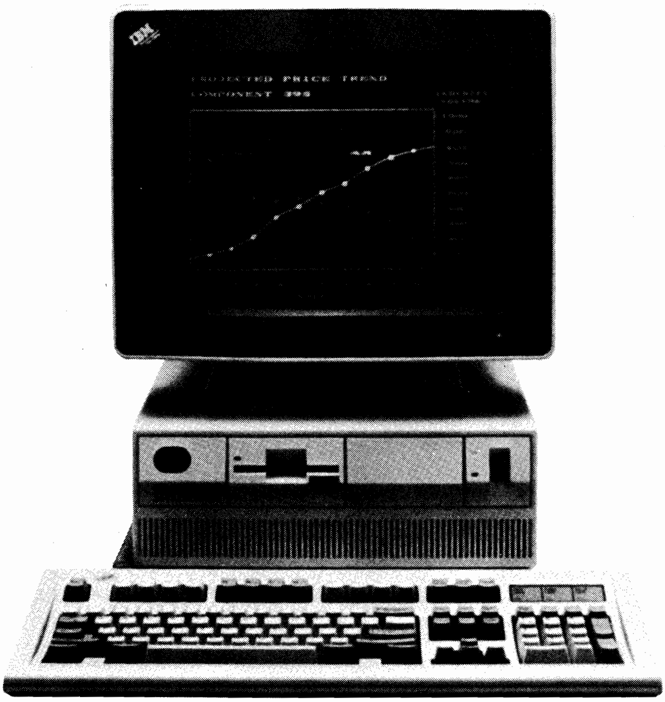


Figure 1.1 A typical personal computer. (Courtesy of International Business Machines Corporation.)

secondary storage devices such as one or more diskette drives and perhaps a hard disk drive. A **hard disk drive** is a disk storage device that reads and writes on rigid magnetic disks. Reading and writing information is faster on a hard disk than on a diskette. Also, a hard disk has a larger storage capacity than does a diskette, and it is usually permanently installed inside the computer cabinet.

The collection of these devices (equipment) along with all of the components that connect these devices together are referred to as computer **hardware**. But computer hardware, by itself, is useless. Hardware must be directed in the processing of information by programs. A computer **program** is a set of instructions that guides a computer through a process. Those instructions are stored in a computer's **memory**. All programs that govern the operation of a computer and make the hardware run are collectively referred to as computer **software**.

There are two general categories of software: application software and systems software. **Application software** is the general term for programs that perform specific user-oriented tasks, such as word processing and spreadsheet applications. **Systems software** is the general term for programs that direct hardware-related tasks. The most important subset of

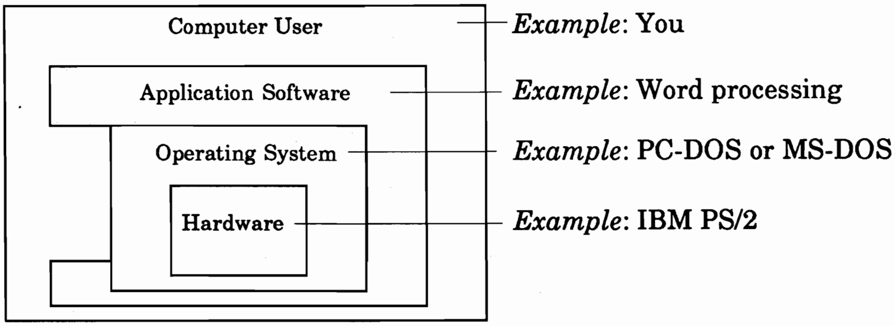


Figure 1.2 The operating system for a computer serves as an interface between the application software and/or computer user and the computer hardware.

systems software is the **operating system**. This collection of programs supervises all the activity that takes place within a computer. An operating system serves as an interface between an application program and computer hardware. It may also serve as an interface between a user and computer hardware. Figure 1.2 illustrates the relationship between computer hardware, an operating system, application software, and a user.

The term **computer system** (often shortened to just **computer**) is used to denote a set of computer hardware and software that is used as a single unit. When you first activate a computer system, there is no program in the computer's memory. Thus, the first step in using a computer is loading a program, usually the operating system, into the computer's memory. This process is commonly referred to as **booting the system**.

Once the operating system is loaded into a computer's memory, it works in the background to perform a number of support functions on a computer. For example, suppose that you want to run a word processing application such as WordPerfect and the application program is stored on a disk. Before this program can be executed, it must first be copied (loaded) into the computer's memory. The process of copying a program from a disk into memory involves considerable logic. The source of a computer's logic is software. Thus, if the word processing program is to be loaded into memory, there must be a special program in memory to control the loading process. That special loading program is just one of the operating system's many support programs. Basically, the operating system is a collection of programs that insulate the user from the computer's hardware, making the computer easier to use.

Communicating with a Computer System

The most common way of communicating with a computer system is through the computer's keyboard. When you press a key, a signal flows from the keyboard, through a cable, and into memory, where a code representing

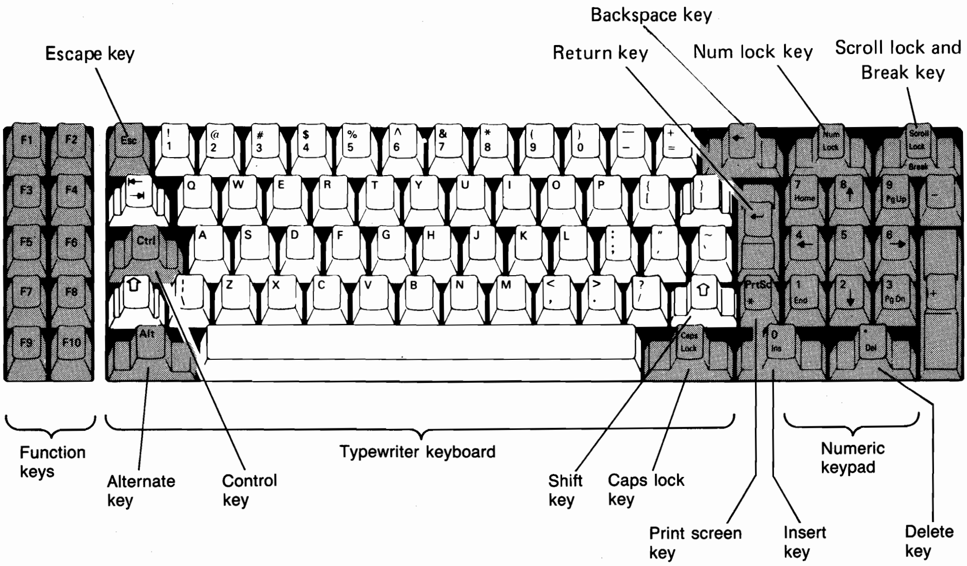


Figure 1.3 The standard IBM PC style keyboard.

the character is stored. Once it is in memory, the character is displayed on the screen. Those two steps happen so quickly that it appears as though you are typing directly to the screen.

The Keyboard for a Personal Computer

Since the keyboard will be the primary device that you will use to communicate with a personal computer system, you need to become familiar with it. Most of you will use a keyboard similar to the standard IBM PC style keyboard (Fig. 1.3) or the enhanced IBM style (101 style) keyboard (Fig. 1.4). Figures 1.3 and 1.4 identify some of the special keys on these keyboards. Figure 1.5 lists the special keys that will be used in this book along with the notation used for each key.

When the keyboard is used to enter information or commands into the computer, a minor distinction is made (in this book) between pressing a key and typing a key. You **type** any key that produces a character, and you **press** any special key. For example, you type A, B, 9, *, ?, and /, while you press ⇧, Alt, Esc, and ↵. When a key combination involves the use of special keys and character keys, the following wording will be used: hold the Ctrl key down and type Z (Ctrl *and* Z), or hold the ⇧ key down and press PrtSc (⇧ *and* PrtSc).

Issuing Commands to the Operating System

Computer systems are not intelligent. Before a computer's operating system can perform one of its support functions, the person using the computer must tell it what to do. The user, much like a military officer, issues orders. The operating system responds like a sergeant, gathering the necessary resources and carrying out each command. A computer user uses the keyboard to send commands to the operating system. Generally, an operating system command consists of a single word or code that is typed on the keyboard. For example,

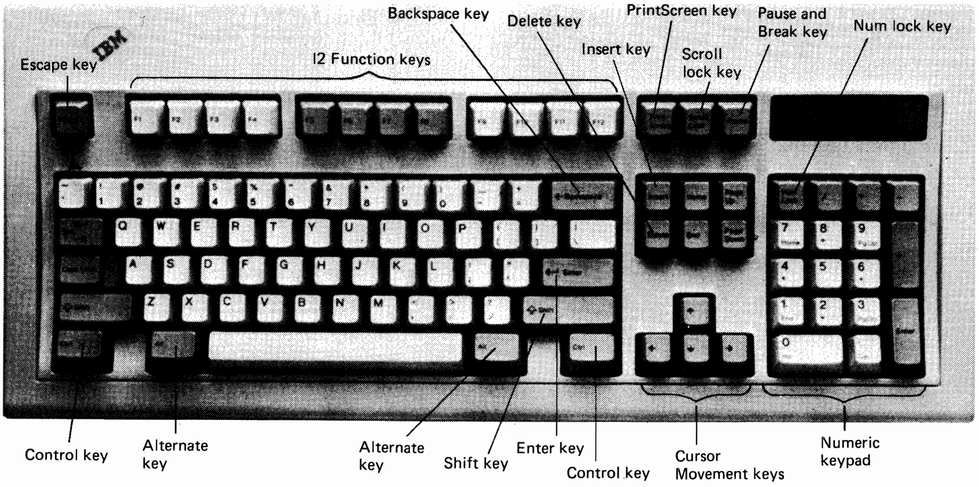


Figure 1.4 The enhanced IBM style (101 style) keyboard. (Courtesy of International Business Machines Corporation).

DISKCOPY is a command used to copy a diskette, and FORMAT is a command used to prepare a diskette for use by a computer system. After a command is typed on the keyboard, it is sent to the operating system by pressing the ↵ key. The operating system program module that accepts, interprets, and carries out commands is called the **command processor**. The command processor consists of a number of program modules, each of which performs a single task. If the command processor determines that the

Keyboard Key	Notation Used for Key
Alternate key	Alt
Break key	Break
Control key	Ctrl
Delete key	Del
Enter key	↵
Escape key	Esc
Insert key	Ins
Pause key	Pause
Print Screen key	Prt Sc
Shift key	⇧

Figure 1.5 Notation for special keyboard keys that is used in this book.

typed command is a valid command, it executes the appropriate program module to carry out the assigned task.

What Is DOS?

This book is concerned with the operating system commands that are used with an IBM type personal computer system. This operating system is commonly referred to as **DOS**. The term DOS is an acronym for **Disk Operating System**. DOS is a collection of special programs that supervise and control the operation of your personal computer. These programs allow you to create and manage files, execute programs, and use different devices, such as a printer, that may be attached to your personal computer. DOS for the IBM Personal Computer family is commonly referred to as **PC-DOS**. DOS was originally developed for IBM by Microsoft Corporation, a leading company in software for personal computers. Microsoft also provides versions of DOS for many other types of personal computers. These versions of DOS are commonly referred to as **MS-DOS (Microsoft DOS)**. There are only minor differences between comparable versions of PC-DOS and MS-DOS. Consequently, only the term DOS will be used in this book.

When you purchase DOS for or with a personal computer, you should get one or more DOS diskettes along with a comprehensive disk operating system manual. The names on the diskette labels may vary with different versions of DOS.

What Are DOS Commands?

Each of the special programs that is included in DOS has a unique name assigned to it. A **DOS command** is a character string specification that references the name of a DOS program. That is, DOS commands specify the execution of DOS programs. You may say that a certain DOS command performs a particular task, but remember that this statement has the following meaning: The DOS command dictates the execution of the appropriate DOS program that performs the assigned task.

Each version of DOS has a certain set of programs that are identified with DOS commands. Often the terms DOS command(s) and DOS program(s) are used interchangeably. IBM PC-DOS 3.30 is the version of DOS used in this book. Appendix B contains a complete list of the DOS commands through DOS Version 3.30.

Parts of DOS

DOS programs can be divided into the following four logical categories: DOS system files (programs), internal DOS commands (programs), external DOS commands (programs), and DOS utility programs. Depending upon the context of a statement, the term DOS can be used to include all or part of these categories. Normally, the term DOS is used to refer to the DOS system files.

DOS System Files

The term **DOS system files** is used to denote three files, two hidden system files and the visible system file COMMAND.COM. In PC-DOS the **hidden files** are called IBMBIO.COM and IBMDOS.COM; in MS-DOS they are usually called IO.SYS and MSDOS.SYS, respectively. All input information (such as input from a keyboard, a mouse, or a communications card) and all output information (such as output to a display screen, a disk, or a printer) are handled (processed) by the programs contained in the two hidden files.

The **IBMBIO.COM** (Basic Input/Output) program handles the input and output between the computer and its peripheral devices such as a disk or a printer.

The **IBMDOS.COM** program forms the heart of DOS. It receives all requests for DOS service functions and converts them into a form to send to IBMBIO.COM.

The **COMMAND.COM** program is called the **DOS command processor** and is the interface between you and DOS. COMMAND.COM displays the system prompt, accepts commands from the keyboard, and interprets the commands so that they can be acted upon by DOS.

Your personal computer is considered to be at **command level** if all three DOS system programs are resident in your computer's memory and a DOS system prompt is displayed on your screen.

Internal DOS Commands

DOS commands (programs) that reside in memory as part of DOS are called **internal DOS commands**. The command processor COMMAND.COM contains the programs that are associated with each of the internal DOS commands.

External DOS Commands

DOS commands (programs) that are contained on disks as program files are called **external DOS commands**. These files have the extension .COM or .EXE as part of their name. When an external DOS command is used, the command processor COMMAND.COM loads the appropriate program file from disk into memory and then initiates its execution.

DOS Utility Programs

Several programs that come on the DOS diskettes are not considered to be DOS commands. These special programs are called **DOS utility programs**. They may be helpful in editing files, linking programs together, or performing other specialized tasks.

Batch Commands

DOS allows you to build a file of DOS commands and then execute these commands as a group. Such a file is called a **batch file** and has the extension .BAT as part of its name. When a batch file name is used as a command, it is referred to as a **batch command**.

DOS System Diskette

The term **DOS System diskette** or **DOS diskette** is used to denote a diskette that contains the three DOS system files and most, if not all, of the DOS external commands (programs). Appendix C outlines the steps to create a DOS System diskette. You always need the DOS System diskette and a couple of blank diskettes to perform the procedures outlined in each chapter of this book.

Disk Drives for a Personal Computer

A personal computer normally contains at least one and often two diskette drives. The specifications **A:** and **B:** are used to denote the diskette drives. If a personal computer contains a **hard disk (fixed disk) drive**, the specification **C:** is used to denote this drive. The term **disk (disk drive)** is used to specify both a diskette (diskette drive) and a hard disk (hard disk drive).

Diskette drives are distinguished by the number of characters that they can store on a diskette and the physical size (5.25 inch or 3.50 inch) of the diskette used in the diskette drive. Hard disks are characterized by their storage capacity and data access rate. The term **byte** is used to denote one character of information. The letter **K** is an abbreviation for kilo, which means one thousand, and the letter **M** is an abbreviation for mega, which means one million. Both K and M are used in specifying the capacities of disks and disk drives.

There are two basic types of 5.25 inch diskette drives, a 360KB diskette drive and a 1.2MB diskette drive. A **360KB diskette drive** can store approximately 360,000 bytes of information on a **360KB diskette** and is commonly referred to as a **double-sided double-density diskette drive**. A **1.2MB diskette drive** can store approximately 1,200,000 bytes of information on a **1.2MB diskette** and is commonly referred to as a **high-density diskette drive**.

There are also two basic types of 3.50 inch diskette drives, a 720KB diskette drive and a 1.44MB diskette drive. A **720KB diskette drive** can store approximately 720,000 bytes of information on a **720KB diskette** that is commonly referred to as a **1MB diskette**. A **1.44MB diskette drive** can store approximately 1,440,000 bytes of information on a **1.44MB diskette** that is commonly referred to as a **2MB (high-density) diskette**.

Care must be taken to use the proper diskettes in the appropriate diskette drives. A 5.25 inch diskette is considered to be **inserted in a diskette drive** if the diskette has been placed in the proper type of diskette drive and the drive latch is closed. A 3.50 inch diskette drive has no drive latch to close after you insert a 3.50 inch diskette in it.

Note that the letters K and M are regularly used in describing the storage capacity of a disk. Sometimes K is used to denote the value of 2^{10}, which is 1024. Sometimes M is used to denote the value of 2^{20}, which is 1,048,576.

If a personal computer system is at command level, the **normal DOS prompt** consists of a disk drive letter followed by the > (greater than) character. The **default disk drive** or **default drive** is specified by the drive letter appearing in the normal DOS prompt. For example, the DOS prompt A > implies that the system is at command level and diskette drive A is the default disk drive.

Entering DOS Commands

The instructions that you give DOS are called **DOS commands** or just **commands**. In order for you to **enter a DOS command** the following steps must be performed.

1. Your personal computer must be at command level.

2. You must type the keys on your keyboard that generate a command name and its parameters using proper syntax (punctuation and grammar). This sequence of characters is called a **command line**. For example, a typical command line may look like

   ```
   DISKCOPY A: B:
   ```

   where DISKCOPY is a command name and A: and B: are parameters.

3. You must press the ↵ key to end the command and send the command line to the DOS command processor for appropriate processing.

If you enter an invalid command, a command with improper syntax, or a command that DOS cannot locate in memory or on a disk, then DOS displays the following message:

```
Bad command or file name
```

Note that DOS makes no distinction between uppercase and lowercase alphabetical characters in a DOS command.

Using the correct syntax to enter a DOS command involves the proper use of delimiters. A command line **delimiter** is a special character, such as a space, comma (,), or slash (/), that is used to separate items in a list or sequence. In general, spaces are used as delimiters in DOS command lines; that is, spaces are used to separate a DOS command name from its parameter list and to separate individual parameter fields. A **parameter** is any item appearing on a command line after the DOS command name. Parameters reside in fields that are separated by command line delimiters. Spaces are required between the different parts of a DOS command line that do not already contain delimiters. If a delimiter is already present, a space may also be included before the delimiter to make the command line more readable. The formal syntax structure showing where spaces are required for each DOS command is given in Appendix B.

Disk Files and Naming

Information on a disk is organized into files. In particular, a **disk file** or **file** is a collection of related information that is treated as a basic unit of storage. In order for DOS to identify a file on a disk, files must be assigned names.

Naming a Disk File

The name of a disk file is referred to as a **file name** and consists of two parts. The first part of a file name is denoted by **filename** and consists of at least one and up to eight of the following characters:

A–Z a–z 0–9 $ & # @ ! % – _ ' ( ) { }

Some versions of DOS allow the use of additional special characters in a filename. The second part of a file name is called the **extension of the filename, filename extension**, **file extension**, or just **extension**. It consists of a period followed by one to three valid filename characters. The notation **.ext** is normally used to denote a filename extension. Filename extensions are an optional part of file names. The notation **filename[.ext]** is used to specify a file name where **[.ext]** denotes that the extension is optional.

A file name may be preceded by a disk drive designator (A:, B:, or C:) and/or information about where to locate the file on a disk (path name). The term **file specification** is used to denote a sequence of characters that completely identifies a file and where to find it. The abbreviation **filespec** is sometimes used to denote file specification.

File Naming Conventions

A file name should give you a general idea of what information is in the file. Proper naming of files can save you a great deal of time in searching for a particular file. You should note that DOS does not make a distinction between uppercase and lowercase alphabetical characters in a file name.

Wildcard (Global) File Name Characters

DOS allows you to specify groups of file names through the use of two special characters. The characters * and ? are called **wildcard (global) file name**

characters when they are used in a file name. The asterisk (*) can be used to represent zero or more characters at the end of a filename or filename extension, while the question mark (?) can be used to represent a single character in a filename or filename extension.

User (Visible) Files and Hidden Files

Disk files can be separated into the following two categories: user (visible) files and hidden files. A **user** or **visible file** is a file that you as a user can access through normal DOS commands. File names for user files are listed on the screen when DOS displays information about files on a disk. A **hidden file** is a file that you as a user cannot access through normal DOS commands. File names for hidden files are not listed when DOS displays information about files on a disk. In this book the term "files" is used to denote user files. The term "user files" may be used to emphasize that hidden files are not included in the discussion.

Types of User (Visible) Files

The type of a user file is normally dictated by the format of the information in the file. Three special file formats (text, COM, and EXE) and two general file formats (language and application) are presented in this section.

Text files are data files that contain **alphanumeric** (both alphabetic and numeric) and special symbol characters in an ASCII format. **ASCII** is an acronym for American Standard Code for Information Interchange and refers to a code scheme that computers use to represent characters of information. Text files are sometimes referred to as **ASCII files** or **ASCII format files**.

COM files contain executable (binary, machine language) programs that can be loaded directly into memory and executed without modifications being made to the programs. The name COM file comes from the .COM filename extension.

EXE files contain executable programs that need to be loaded into memory with some address modifications being made before they can be executed. The name EXE file comes from the .EXE filename extension.

Files containing programs written in a programming language such as BASIC are called **programming language files**. These files contain information that a corresponding language processor can read and translate into executable programs. BASIC language files have .BAS as a filename extension.

Application programs create **application data files** in a format that can be read and understood by that application. Application data files have filename extensions associated with the application. For example, a word processing application might use .DOC as a filename extension, and a spreadsheet (worksheet) application might use .WKS as a filename extension.

DOS Device Names

DOS has special names called **DOS device names** assigned to input/output **(I/O)** devices such as the keyboard, the display screen, and the printer. These device names have a special meaning when used in a DOS command and are reserved for use only with the assigned devices.

CON is an abbreviation for console. CON is assigned to the keyboard for input and the display screen for output.

PRN is an abbreviation for printer. It is an output device and refers to the parallel printer that DOS uses unless you instruct DOS otherwise. **LPT1** is another abbreviation for printer and is equivalent to PRN.

Summary

You will find a summary of new DOS commands and functions at the end of each chapter. No commands were introduced in Chapter 1.

Self-Test

The answers to the Self-Test questions at the end of each chapter are in Appendix D.

1. The ________________ is the primary input device for a personal computer.
2. A ________________ and a ________________ are two common output devices for a personal computer.
3. The collection of computer equipment is commonly referred to as computer ________________ .
4. A computer ________________ is a set of instructions that guides a computer through a process.
5. All programs that govern the operation of a computer and make the hardware run are collectively referred to as computer ________________ . Name the two general categories of computer programs. ________________
6. An ________________ is the collection of programs that supervises all the activity that takes place within a computer.
7. The term ________________ denotes a set of computer hardware and software that is used as a single unit.
8. The operating system program module that accepts, interprets, and carries out commands is called the ________________ .
9. The acronym DOS stands for ________________ .
10. The three DOS system files are ________________ . The two hidden system files are ________________ .
11. DOS commands that reside in memory as part of DOS are called ________________ .

12. DOS commands that are contained on disks as part of DOS are called ________________.

13. A command line ________________ is a special character that is used to separate items in a list or sequence.

14. A ________________ is any item appearing on a command line after the DOS command name.

Exercises

1. Briefly characterize the different types of disk drives used on a personal computer.
2. A personal computer is at command level. What does this mean?
3. What is a command line?
4. Briefly outline the steps needed to enter a DOS command.
5. State the rules for naming a disk file.
6. Briefly explain the difference between user files and hidden files.

2

Starting (Booting) DOS

The material in this chapter introduces the DOS features and commands that allow you to:

- start DOS running (boot the system) in your personal computer
- check and change the DOS system date using the DATE command
- check and change the DOS system time using the TIME command
- change the current (default) disk drive
- clear the screen using the CLS command
- display the active version of DOS using the VER command
- obtain a hard copy listing of the information displayed on the screen using the ⇧ *and* PrtSc key combination
- turn the printer echo mode on or off using the Ctrl *and* PrtSc or Ctrl *and* P key combination
- log off DOS

Booting the Operating System

Recall from Chapter 1 that the term DOS System diskette is used to denote a diskette that contains the DOS system files and most, if not all, of the disk operating system programs. This means that all of the internal DOS commands and most of the external DOS commands can be loaded and executed from this diskette.

Before you can use your computer to perform any useful tasks, the operating system (DOS) must be loaded (copied) from the DOS System diskette into the computer's memory; that is, you must start your computer. The process of starting a computer is referred to as the **boot process**, **booting the system**, **booting DOS**, or **starting DOS**.

Booting the System from a Diskette

To begin the boot (start) process on a personal computer containing only diskette drives, you need a working copy of the DOS System diskette. (If you do not have a working copy of the DOS System diskette, refer to Appendix C.) Insert the DOS System diskette in drive A and perform one of the following steps:

1. If the power to your computer is off, turn the power switch from OFF to ON (or from 0 to 1). For some computers the monitor's power may also need to be turned on. Initiating the boot process on your computer by turning the power on is called a **cold start** or **cold boot**.
2. If the power to your computer is already on, hold the Ctrl and Alt keys down and press Del (denoted by Ctrl *and* Alt *and* Del). Using the Ctrl *and* Alt *and* Del key combination to initiate the boot process on a personal computer is called a **warm start**, **warm boot**, **system reset**, or **system restart**.

Once the boot process has been initiated, several tasks are performed automatically by your computer system. Your computer contains two special programs that permanently reside in its read-only memory (**ROM**). If the boot process begins with a cold start, your computer's central processing unit (**CPU**) automatically executes both of these special programs. If the boot process begins with a warm start, only the second program is executed by your CPU. The first program performs a short self-test on some of the computer's component parts, checking for certain types of hardware (equipment) failures. The execution of this program is the reason your computer appears to be doing nothing for several seconds after the boot process is initiated. The second program is called the **bootstrap loader** because it "pulls DOS up by the bootstrap." It begins the process of loading DOS into the computer's memory by first loading and then executing a program called the **bootstrap program** from the initial part of the DOS disk. The execution of this program causes DOS's two hidden files (IBMBIO.COM and IBMDOS.COM) and the file COMMAND.COM to be loaded into your computer's memory.

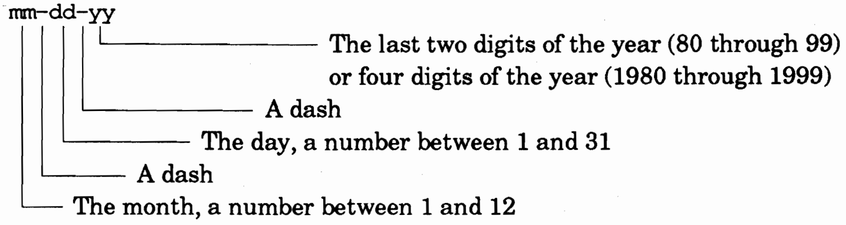

mm-dd-yy
The last two digits of the year (80 through 99) or four digits of the year (1980 through 1999)
A dash
The day, a number between 1 and 31
A dash
The month, a number between 1 and 12

Figure 2.1 The current date should be entered in the above format.

Once the DOS system files have been loaded into memory, DOS displays its current date followed by a message to enter a new date. Respond to the message:

```
Enter new date (mm-dd-yy):
```

by pressing ↵ to accept the displayed date as the current system date, or by typing the current date in the displayed format (Fig. 2.1) and then pressing ↵. For example, if today's date is August 8, 1989, type `8-8-89` and press ↵. Next DOS displays its current time followed by a message to enter a new time. Respond to the message:

```
Enter new time:
```

by pressing ↵ to accept the displayed time as the current system time, or by typing the current time in the displayed format using a twenty-four hour clock (Fig. 2.2) and then pressing ↵. Since DOS uses a twenty-four hour clock, 10:15 A.M. is entered as 10:15, and 3:45 P.M. is entered as 15:45. Thus, if the present time of day is 4:25 P.M., type `16:25` and press ↵.

After the above preliminaries have been completed, DOS sends a multiline message (Fig. 2.3) to the screen identifying the version of DOS and copyright information followed by the normal DOS prompt A> from the DOS command processor. This prompt indicates that the system is at command level, waiting for a command from you, and that the current (default) disk drive is drive A.

You should always be careful to enter the correct date and time during the boot process. In addition to DOS, many application programs, such as word processors and spreadsheet applications, use the DOS system date and time. If the date and time are not entered correctly when you boot the system, this

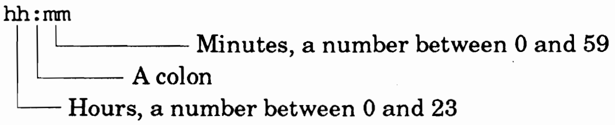
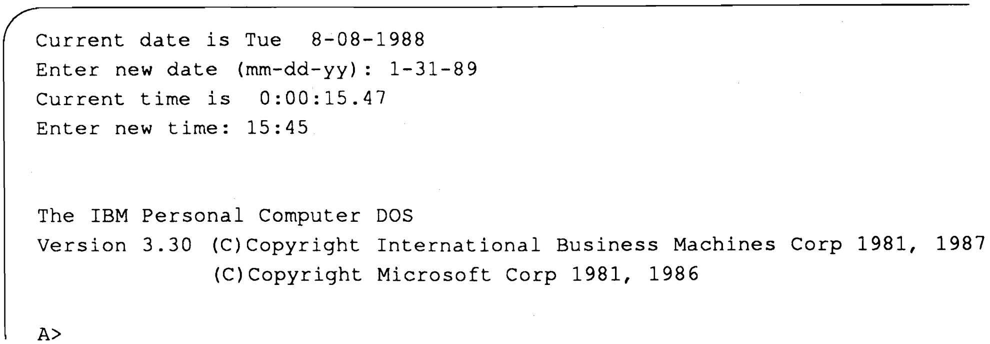

hh:mm
Minutes, a number between 0 and 59
A colon
Hours, a number between 0 and 23

Figure 2.2 The current time should be entered in the above format.



```
Current date is Tue  8-08-1988
Enter new date (mm-dd-yy): 1-31-89
Current time is  0:00:15.47
Enter new time: 15:45


The IBM Personal Computer DOS
Version 3.30 (C)Copyright International Business Machines Corp 1981, 1987
             (C)Copyright Microsoft Corp 1981, 1986


A>
```

Figure 2.3 After the system is booted, information similar to the above is displayed on the screen. The DOS prompt is A>, which indicates that the system is at command level and diskette drive A is the default disk drive.

information will not be correct when applications use functions that reference the DOS system date and time. Also, if the current date and time are not correctly entered into the computer, then the date and time associated with any file that is created or modified during this session will not be correct. This might cause problems in the future when you try to determine the creation date or the last modification date of a particular file.

You might notice that some personal computers never display the correct date and time when you boot the system, while other computers always appear to display the correct date and time. If a personal computer contains a **real-time clock/calendar (permanent clock)** with battery backup, then the system date and time should be correct when they are displayed during the boot process, since this device automatically keeps track of the date and time. If a personal computer does not contain a permanent clock, you will have to enter the correct date and time each time you boot the system.

Keystroke Summary: Booting the System from a Diskette

Insert: DOS System diskette in drive A

Perform: cold boot (turn computer's power switch on) or
warm boot (Ctrl *and* Alt *and* Del)

Check: displayed system date

Press: ↵ to accept the displayed system date or

Type: *mm-dd-yy* and Press: ↵

to change the system date where *mm* denotes the month (01 through 12), *dd* denotes the day (01 through 31), and *yy* denotes the year (80 through 99, or 1980 through 1999)

Check: displayed system time

Press: ↵ to accept the displayed system time or

Type: *hh:mm* and Press: ↵

to change the system time where *hh* denotes hours (0 through 23) and *mm* denotes minutes (0 through 59)

Booting the System from a Hard Disk

DOS must be installed on a hard disk before a hard disk can be used to boot the system on a personal computer. Unless you are using a new personal computer with a hard disk that has not been prepared for use by DOS, DOS is more than likely installed on the hard disk. If for some reason DOS is not installed on the hard disk, you should consult with your instructor. The material covered in this book assumes that your hard disk is prepared for use by DOS and that all DOS programs are on the hard disk.

Before you begin the boot (start) process on a personal computer containing a hard disk, you need to make sure that there is no diskette in drive A. Next perform one of the following steps:

1. If the power to your computer is off, perform a cold boot (cold start); that is, turn the power switch from OFF to ON (or from 0 to 1). For some computers the monitor's power may also need to be turned on.
2. If the power to your computer is on, perform a warm boot (warm start); that is, hold the Ctrl and Alt keys down and press Del (denoted by Ctrl *and* Alt *and* Del).

The steps performed when booting the system from a hard disk are essentially the same as those performed when booting the system from a diskette. Consequently, they will not be repeated here.

After the boot process has been completed, the DOS command processor displays the normal DOS prompt C> on the screen. This prompt indicates that the system is at command level waiting for a command from you and that the current (default) disk drive is drive C.

If your personal computer contains a hard disk, then, like most personal computers, it should automatically boot from the hard disk when drive A contains no formatted diskette. If a formatted diskette is contained in drive A, most personal computers will attempt to boot the system from that diskette. A comprehensive discussion on using a hard disk is given in Chapter 8.

Keystroke Summary: Booting the System from a Hard Disk

Remove: any diskette from drive A

Perform: cold boot (turn computer's power switch on) or warm boot (Ctrl *and* Alt *and* Del)

Check: displayed system date

Press: ⏎ to accept the displayed system date or

Type: *mm-dd-yy* and Press: ⏎

to change the system date where *mm* denotes the month (01 through 12), *dd* denotes the day (01 through 31), and *yy* denotes the year (80 through 99, or 1980 through 1999)

Check: displayed system time

Press: ⏎ to accept the displayed system time or

Type: *hh:mm* and Press: ⏎

to change the system time where *hh* denotes hours (0 through 23) and *mm* denotes minutes (0 through 59)

What Can Go Wrong?

1. A message similar to the following is displayed on the screen when you attempt to boot the system:

```
Non-System disk or disk error
Replace and strike any key when ready
```

Cause: The diskette in drive A does not contain the three DOS system files that are necessary to boot the system.

Solution: Remove the diskette from drive A and check to be sure that it is the DOS System diskette. If it is the DOS System diskette, center the diskette in the diskette jacket, insert it back in drive A, and press any key to reinitiate the boot process. If the same message appears again, you have a bad DOS System diskette and it needs to be replaced.

2. One of the following messages is displayed on the screen after you enter the date or the time:

```
Invalid date                    or    Invalid time
Enter new date (mm-dd-yy):            Enter new time:
```

Cause: You probably typed the date or the time in the wrong format.

Solution: Invalid date and invalid time messages are displayed for a variety of reasons. For example, 7-49-89 generates an invalid date message, and 18:75 generates an invalid time message. Check the proper format for typing the date (Fig. 2.1) or the time (Fig. 2.2), and retype your entry.

3. You realize that you typed the wrong date or time but DOS accepted it.

Cause: You typed an incorrect date or time but it was accepted by DOS because it was typed in a valid format.

Solution: DOS cannot determine whether a date or time that you enter is correct. DOS can only determine whether the format of the date or time is valid. For example, 5-6-81 is a valid date to DOS, but it is not today's date. The next two sections show how to correct an invalid date and an invalid time in your computer system.

4. The boot process does not have steps that prompt you for the date and time, or the sequence and type of tasks are different from the steps described above.

Cause: The DOS System diskette contains a special file named AUTOEXEC.BAT, which may alter the tasks that DOS performs after the DOS system files have been loaded into memory.

Solution: You should follow any prompt messages that appear on the screen. If you are not prompted for the current date and time, you can use the DATE and TIME commands to verify and change (if necessary) the system date and time. The next two sections cover these commands. The creation and use of the AUTOEXEC.BAT file is covered in Chapter 9.

5. You initiate a cold boot, but no information is displayed on the screen.

Cause: The power to the monitor is off or the brightness/contrast setting for the screen is too low.

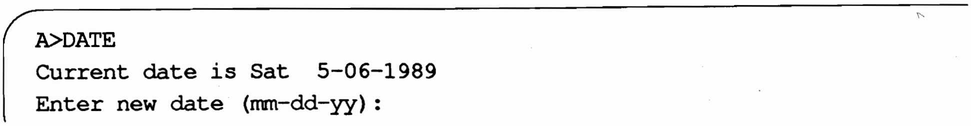
Solution: Check the power switch to the monitor and make sure that it is turned on. Check the brightness/contrast controls for the screen to see whether it needs to be adjusted to a brighter setting.

Checking and Changing the System Date (DATE)

There are times when you want to check and/or change the system date that DOS is using on your computer. The **DATE** command is an internal DOS command that allows you to check and change (if necessary) the DOS system date.

With the cursor positioned after the A> prompt, type

```
DATE
```



```
A>DATE
Current date is Sat  5-06-1989
Enter new date (mm-dd-yy):
```

Figure 2.4 When the DATE command is used to check and/or change the system date, information similar to the above is displayed on the screen. 5-06-1989 denotes the date of May 6, 1989.

and press ↵. A message followed by the current DOS system date is displayed on the screen (Fig. 2.4). Check the displayed date. If the date is correct, press ↵. If the date is not correct, type the correct date in the displayed format (Fig. 2.1) and press ↵.

Keystroke Summary: Checking and Changing the System Date

Type: `DATE`

Press: ↵

Check: displayed system date

Press: ↵ to accept the displayed system date or

Type: *mm-dd-yy* and Press: ↵

to change the system date where *mm* denotes the month (01 through 12), *dd* denotes the day (01 through 31), and *yy* denotes the year (80 through 99, or 1980 through 1999)

What Can Go Wrong?

1. DOS displays the following message on the screen:

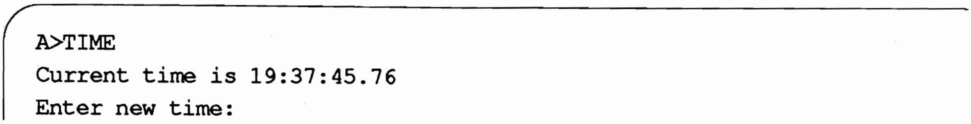
```
Bad command or file name
```

Cause: You probably made an error in typing the DOS command name.

Solution: Check the characters you typed for the DOS command name and retype your entry.

Checking and Changing the System Time (TIME)

There are times when you want to check and/or change the system time that DOS is using on your computer. The **TIME** command is an internal



```
A>TIME
Current time is 19:37:45.76
Enter new time:
```

Figure 2.5 When the TIME command is used to check and/or change the system time, information similar to the above is displayed on the screen. 19:37:45.76 denotes the approximate time of 7:37 P.M. The general form of the time display is *hh:mm:ss:xx* where *hh* specifies hours (0 through 23), *mm* specifies minutes (0 through 59), *ss* specifies seconds (0 through 59), and *xx* specifies hundredths of a second (0 through 99).

DOS command that allows you to check and change (if necessary) the DOS system time.

With the cursor positioned after the A> prompt, type

```
TIME
```

and press ↵. A message followed by the current DOS system time is displayed on the screen in a twenty-four hour clock format (Fig. 2.5). This means that the displayed time 16:25:15.46 denotes the approximate time of 4:25 P.M. Check the time displayed on the screen. If the displayed time is correct, press ↵. If the displayed time is not correct, type the correct time in the displayed format (Fig. 2.2) and press ↵.

Keystroke Summary: Checking and Changing the System Time

Type: `TIME`

Press: ↵

Check: displayed system time

Press: ↵ to accept the displayed system time or

Type: *hh:mm* and Press: ↵

to change the system time where *hh* denotes hours (0 through 23) and *mm* denotes minutes (0 through 59)

Changing the Current (Default) Disk Drive (d:)

When the system is at command level and drive A is the current (default) disk drive, DOS displays the A> prompt. If you want to change the default drive to a different drive, enter the letter identifying the drive followed by a colon (:).

With the cursor positioned after the A> prompt, type `B:` and press ↵. This entry instructs DOS to change the current drive to drive B and to display the B> prompt. If your computer contains a hard disk, type `C:` and press ↵. The current disk drive is changed to drive C (the hard disk drive), and DOS displays the C> prompt. To change the current disk drive back to drive A, type `A:` and press ⏎.

Keystroke Summary: Changing the Current (Default) Disk Drive

Type: `d:` where *d:* denotes the designated drive

Press: ⏎

What Can Go Wrong?

1. DOS displays the following message on the screen:

```
Invalid drive specification
```

Cause: You typed an alphabetic character followed by a colon that does not specify a valid disk drive.

Solution: If you entered C: and your computer does not contain a hard disk, the above message will be displayed on your screen. Check the drive specification that you typed and correct it by entering a valid drive specification for your computer.

Clearing the Screen (CLS)

The **CLS** (CLear Screen) command is an internal DOS command that allows you to clear all displayed information from the screen except the system prompt, which appears in the upper left-hand corner of the screen.

With the cursor positioned after the A> prompt, type

```
CLS
```

and press ⏎. All information that was displayed on your screen is cleared, and the A> prompt is displayed in the upper left-hand corner of the screen.

Keystroke Summary: Clearing the Screen

Type: `CLS`

Press: ⏎

Different Versions of DOS

DOS has been revised several times since the introduction of the first version of DOS, Version 1.00. The **DOS version number** is displayed in the format **X.xx** where X denotes a major revision number and xx denotes a minor revision number. DOS is revised to add more capability, to take advantage of advances in computer equipment, and to correct errors. A change in xx, the number following the decimal point, indicates a minor change that leaves DOS virtually the same. A change from DOS Version 3.10 to Version 3.20 is an example of such a change. A change in X, the number preceding the decimal point, indicates a major change in DOS. A change from Version 2.10 to Version 3.10 is an example of a major change in DOS.

***WARNING:* Since all parts of DOS work very closely together, you should be careful not to mix parts of DOS from two different versions. Unpredictable things can happen as a result of mixing different versions of DOS, such as losing an entire data disk.**

Determining the DOS Version (VER)

Since some applications run only under certain versions of DOS, you need a way to determine which version of DOS is active in your computer system. The **VER** (VERsion) command is an internal DOS command that allows you to display the version of DOS that is active in your computer system.

With the cursor positioned after the A> prompt, type

```
VER
```

and press ↵. A message similar to the following is displayed on the screen:

```
IBM Personal Computer DOS Version 3.30
```

Keystroke Summary: Determining the DOS Version

Type: `VER`

Press: ↵

Printing Information Displayed on the Screen

Many times you encounter a situation in which you want a hard copy (printed) listing of information that is displayed on the screen. DOS does not contain a command that allows you to print a screen image. However, most personal computers support a hardware (equipment) function, called the **print screen** function, that allows you to obtain a hard copy listing of the information that is displayed on the screen. The print screen function is initiated by holding the ⇧ key down and pressing PrtSc (denoted by ⇧ *and* PrtSc). Be sure your printer is turned on before you send output to the printer.

Insert the DOS system diskette in drive A and boot the system on your personal computer. After you respond to the date and time prompts, information similar to that given in Fig. 2.3 is displayed on the screen. To obtain a hard copy listing of this information, hold the ⇧ key down and press PrtSc.

The ⇧ *and* PrtSc key combination initiates the print screen function that is built into the hardware of your personal computer. After the cursor scans each row on the screen, sending the displayed information to the printer, it returns to its original position following the A> prompt.

Turning the Printer Echo Mode On and Off

When you are working on a personal computer, the information that you type on the keyboard and the information and messages generated from DOS commands are displayed on the screen. When the screen becomes full, the information displayed on the screen scrolls from the bottom toward the top and then off of the top of the screen. Sometimes you may want to have a hard copy (printed) listing of what is being displayed on the screen. DOS supports a printer echo mode that can be turned on and off with a special key combination. To turn the **printer echo mode** from off to on or from on to off, you need to hold the Ctrl key down and press PrtSc or hold the Ctrl key down and type P (denoted by Ctrl *and* PrtSc or Ctrl *and* P). When the printer echo mode is turned on, everything that DOS sends to be displayed on the screen is also sent (echoed) to the computer's printer. This feature allows you to automatically get a printed copy of what appears on the screen. Normally, the printer echo mode is turned off.

Insert the DOS System diskette in drive A and boot the system on your personal computer. After you respond to the date and time prompts, information similar to that given in Fig. 2.3 is displayed on the screen. At this point the printer echo mode is turned off. Ready the printer attached to your personal computer. Turn the printer echo mode on by holding the Ctrl key down and pressing PrtSc or by holding the Ctrl key down and typing P. With the cursor positioned after the A> prompt, type `DATE` and press ↵. Notice that once the printer echo mode is turned on, everything that is displayed on the screen is also printed on the printer. Respond to the date prompt. With the cursor positioned after the A> prompt, type `VER` and press ↵. Again notice that the DOS version information is displayed on the screen and echoed to the printer. Now turn the printer echo mode off using the same Ctrl *and* PrtSc or Ctrl *and* P key combination. With the cursor positioned after the A> prompt, type `VER` and press ↵. With the printer echo mode turned off, the DOS version information is displayed only on the screen.

Logging Off DOS

When you are finished using a personal computer, you need to log off DOS. This is a very simple but important task. The process of **logging off DOS** consists of first removing your diskette(s) from the appropriate diskette drive(s) and then turning the computer's power switch off.

Summary

Topic or Feature	Command or Reference	Page
Boot DOS from a diskette	Insert DOS System diskette in drive A; turn power switch on or press Ctrl *and* Alt *and* Del	15
Boot DOS from a hard disk	Remove any diskette from drive A; turn power switch on or press Ctrl *and* Alt *and* Del	18
Check and/or change system date	`DATE` Figs. 2.1, 2.4	20 16, 21
Check and/or change system time	`TIME` Figs. 2.2, 2.5	21 16, 22
Change the current (default) disk drive	`d:` where *d* denotes the designated drive	22
Clear the screen	`CLS`	23
Display active version of DOS	`VER`	24
Print screen information	⇧ *and* PrtSc	24
Printer echo mode on and off	Ctrl *and* PrtSc or Ctrl *and* P	25
Log off DOS	Remove diskettes and turn power switch off	25

Self-Test

1. The process of starting a personal computer is known as ____________________.
2. A cold boot or cold start is initiated by ____________________.
3. A warm boot or system restart is initiated by ____________________.
4. What program is executed when you do a cold boot of the system but is not executed when you do a system restart? ____________________
5. Name the three DOS system files that must be on a disk before it can be used to successfully boot the system. ____________________
6. Name two pieces of information that are normally entered during the boot process. ____________________
7. Most personal computers that contain a hard disk will boot the system from the hard disk unless ____________________.
8. The ________________ command is used to check and/or change the system date. Is this command an internal or external DOS command? ________________ State the general form of the response to this DOS command. ________________
9. The ________________ command is used to check and/or change the system time. Is this command an internal or external DOS command? ________________ State the general form of the response to this DOS command. ________________

10. What do you type to change the current disk drive from drive A to drive B? ____________ What system prompt is now displayed on the screen? __________

11. The ____________________ command is used to clear all displayed information from the screen. Is this command an internal or external DOS command? ____________________ After this command is executed, where is the DOS prompt displayed? ____________________

12. The ____________________ command is used to determine the version of DOS that is running in a personal computer. Is this command an internal or an external DOS command? ____________________

13. State the key combination that allows you to obtain a hard copy (printed) listing of the information displayed on the screen. ____________________

14. State the key combination that allows you to turn the printer echo mode on and off. ____________________

15. The process of logging off DOS consists of ____________________.

Exercises

1. Perform the appropriate function and/or use an appropriate DOS command to perform each of the assigned tasks.

   a. Boot the system.
   b. Make sure the correct date and time are entered.
   c. Display but do not change the system date.
   d. Display and change the system time to 7:53 P.M.
   e. Display the active version of DOS.
   f. Change the default drive to a different drive.
   g. Obtain a hard copy listing of the information displayed on the screen.
   h. Clear the screen.
   i. Log off DOS.

2. Perform the appropriate function and/or use an appropriate DOS command to perform each of the assigned tasks.

   a. Boot the system.
   b. Make sure the correct date and time are entered.
   c. Turn the printer echo mode on so that you will obtain a hard copy listing of parts (d) through (g).
   d. Display but do not change the system date.
   e. Display and change the system time to 3:29 P.M.
   f. Display the active version of DOS.
   g. Change the default drive to a different drive.
   h. Turn the printer echo mode off.
   i. Clear the screen.
   j. Log off DOS.

3

Preparing a New Disk for Use

The material in this chapter introduces the DOS commands that allow you to:

- prepare a data diskette using the FORMAT command
- prepare a bootable diskette using the FORMAT command with the /S option
- transfer the three DOS system files to a formatted diskette with an empty directory using the SYS command
- prepare a diskette with a volume label using the FORMAT command with the / V option

Formatting a Disk

Before you learn how to prepare (format, initialize) a disk to use with a personal computer, you need to understand why a disk must be formatted before it can be used and what the formatting process does to a disk. First, disks are used as a medium upon which information is electronically stored, similar to the way sound is stored on a cassette tape. This means that DOS must have a way to determine those parts of the disk that are free to receive information and a way to find and read information previously stored on the disk.

For example, suppose you are in charge of a new post office building that has 5000 post office boxes. How will you manage the use of these boxes? First, you realize that the boxes must be systematically numbered so that your employees can distribute the mail correctly and so that your customers can locate their mail. Second, you decide to rent (allocate) the boxes in sequential order to simplify record keeping. This method could cause a few problems. You might have a very short person assigned a top box that he or she cannot reach; you might have customers who want a large box or more than one box; you might have a box that was rented but is now free because the renter of the box moved out of town; you might have several boxes with broken locks that should not be rented (allocated) until they are repaired. You soon realize that if you are going to be successful at managing this facility (space), you must have ways

- for your employees to place (store) mail (information) in the correct boxes,
- for your customers to get (retrieve) their mail (information), and
- for you to determine what boxes are being used, what boxes are bad and should not be used, what boxes can be rented (allocated), and in what order you will rent (allocate) the free boxes.

DOS has some of the same problems in managing the finite storage space on the surface of a disk. Consequently, DOS must prepare (format) a disk so that it knows how the surface of the disk is subdivided and numbered (addressed), whether one or two sides of the disk are being used, where to find (read) information previously stored on the disk, where bad spots are on the disk, where to store (write) new information on the disk, and how to free space for information that you want to delete from the disk. In the next two sections you will see how the surface of a disk is subdivided and addressed for use by DOS and what tasks are actually performed during the formatting process.

Characteristics of Unformatted and Formatted Disks

Recall that diskettes used with personal computers are normally characterized by their physical size (5.25 inch and 3.50 inch) and the number of bytes (characters) that can be stored on the diskette. Figure 3.1 illustrates a 5.25 inch diskette that has not been formatted. Notice that the surface of the diskette is not subdivided and no addressing (numbering) scheme is present on the diskette. Figure 3.2 illustrates the top and bottom view of a 3.50 inch diskette that has not been formatted. Figure 3.3 illustrates a 5.25 inch 360KB diskette that has been formatted by DOS.

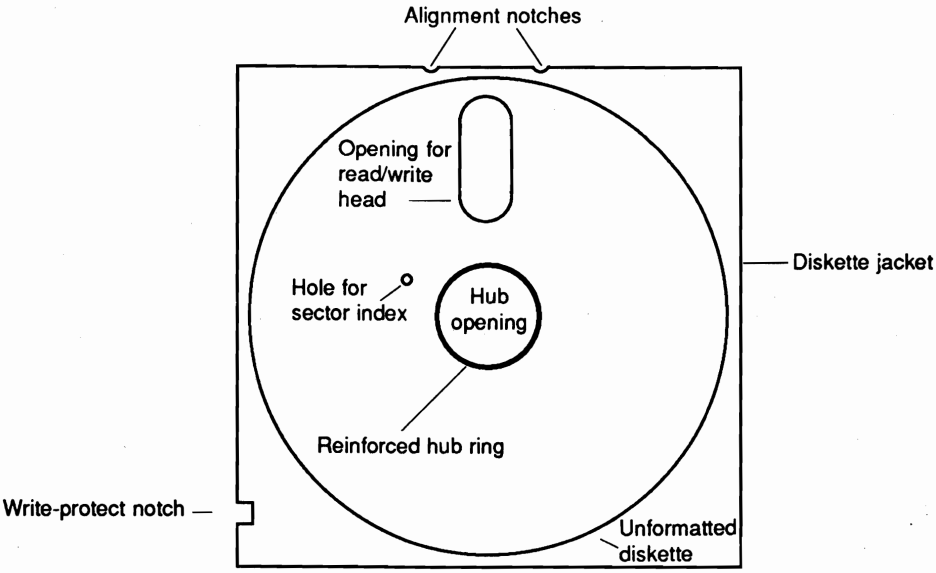


Figure 3.1 Unformatted 5.25 inch diskette

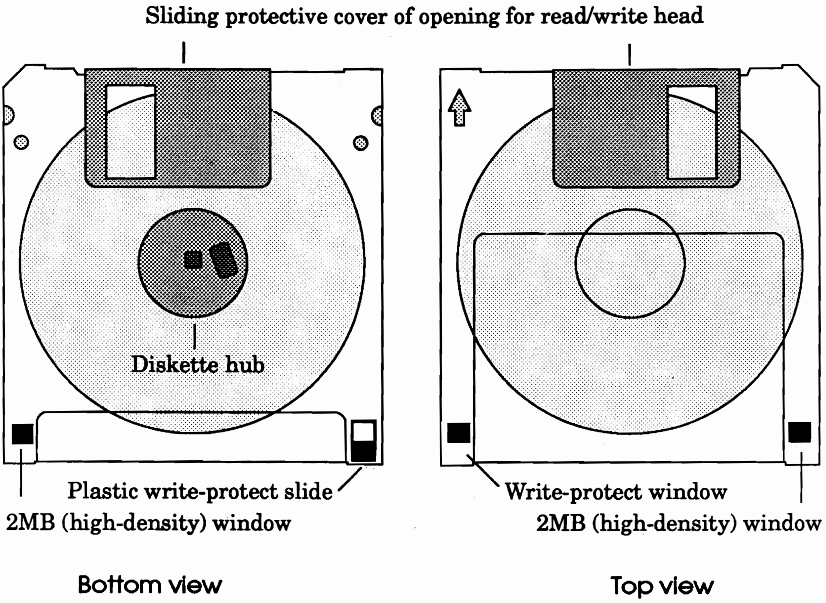

Sliding protective cover of opening for read/write head
Diskette hub
Plastic write-protect slide
2MB (high-density) window
Write-protect window
2MB (high-density) window
Bottom view
Top view

Figure 3.2 Unformatted 3.50 inch diskette

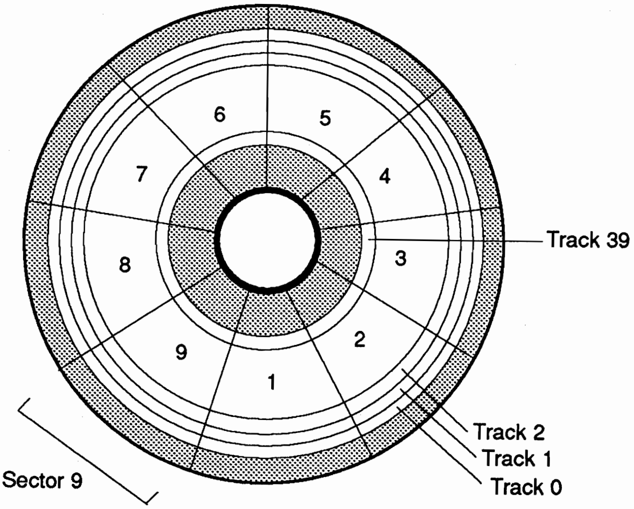


Figure 3.3 Formatted 5.25 inch 360KB diskette

Although Figs. 3.1 and 3.2 illustrate characteristics of unformatted 5.25 and 3.50 inch diskettes and Fig. 3.3 illustrates characteristics of a formatted 5.25 inch diskette, similar characteristics hold for all types of disks, including hard disks. For this reason the remarks in the remainder of this section are made about disks in general.

A formatted disk is divided into narrow concentric circles called **tracks**. Each track is subdivided into smaller areas called **sectors**; each sector can hold 512 bytes (characters) of information. DOS uses the disk side (bottom is side 0, top is side 1), track, and sector numbers to save and retrieve information; that is, usable areas of a disk's surface are addressed by using the side, track, and sector numbers. Even though a formatted disk is electronically divided into tracks and sectors as illustrated in Fig. 3.3, the format structure on the surface of a disk is not visible to an individual. The number of tracks and sectors that are created on a formatted disk is determined by the version of DOS that you use, the type of disk drive that is used in the formatting process, and the parameters that are specified in the formatting process. For complete information about capacity characteristics of different types of diskettes you should refer to Table B.2 under the FORMAT command in Appendix B.

Once a diskette has been formatted, a diskette drive can write information on the diskette and read information from the diskette. If a **write-protect tab** (sticker) is placed over the write-protect notch on the diskette jacket of a 5.25 inch diskette (Fig. 3.1), a diskette drive can read information from the diskette but cannot write information to the diskette. The diskette is said to be **write protected**. If the write-protect window on a 3.50 inch diskette (Fig. 3.2) is closed, a diskette drive can read and write information on the diskette. If the write-protect window on a 3.50 inch diskette is open, the diskette is write protected; that is, a diskette drive can read information from the diskette but cannot write information to the diskette. Normally, a diskette is write protected to protect the contents of the diskette, that is, to protect important information on the diskette from being changed or deleted and to prevent a diskette drive from writing additional information on the diskette.

Results of the FORMAT Command

The **FORMAT** command is an external DOS command that prepares (formats) a disk so that it can be used by DOS to save (store) and retrieve (read) information. The FORMAT command does five things to a disk to prepare it for use by DOS.

1. FORMAT addresses the usable area of the disk by writing electronic guidelines on the surface of the disk as outlined in Fig. 3.3. These guidelines allow DOS to locate a particular area of the disk by using the side, track, and sector numbers (addresses).
2. FORMAT writes on each sector of the disk to determine whether it is free of defects. Bad or defective sectors are marked as write protected so that DOS cannot use the bad parts of the disk to store data.
3. FORMAT creates a **directory** (main or root directory) for the disk, which contains information about files stored on the disk. This information consists of the filename, the filename extension, the file size in bytes, the date and time the file was created or last updated, and the location (address) of the beginning of the file data on the disk. The directory of a disk serves as a table of contents for the information stored on the disk.
4. FORMAT creates a **file allocation table (FAT)** for the disk. This table contains the information to keep track of bad tracks on the disk, to identify which sectors belong to which files, and to identify which sectors are not being used (free sectors). The file allocation table may be thought of as an index to the usable area of a disk.
5. FORMAT places a copy of the **boot record** on the first sector of the disk (side 0, track 0, sector 1). The boot record contains information about the format structure of the disk along with the bootstrap program.

The above items indicate that the FORMAT command identifies five types of sectors on a formatted disk: boot record sector, directory sectors, file allocation table sectors, bad or defective sectors, and data sectors (where file data are stored). The boot record sector, the directory sectors, and the file allocation table sectors are considered to be disk overhead on a formatted disk.

Preparing Data Diskettes (FORMAT)

Now that you know why a disk must be formatted (initialized) to be used by DOS and what the formatting process actually does to a disk, you are ready to learn how to format a disk using the FORMAT command. Note that most sections of this book illustrate the use of external DOS commands by referencing these commands from the DOS System diskette. The use of external DOS commands from a hard disk is covered in Chapter 8.

When you are formatting new diskettes, you should format several diskettes so that you always have extra formatted diskettes available for immediate use. The first time you lose data because your data diskette is full and will accept no more information, and you do not have a formatted diskette available for use, you will understand the true value of having formatted diskettes readily available. You will need at least two blank diskettes to perform the procedures outlined in the remaining sections of this chapter.

***WARNING:* The FORMAT command is one of the most dangerous DOS commands and should be used with great caution. Any data residing on a diskette will be destroyed by the formatting process. Before you format a diskette, you should be sure that there are no valuable data on the diskette.**

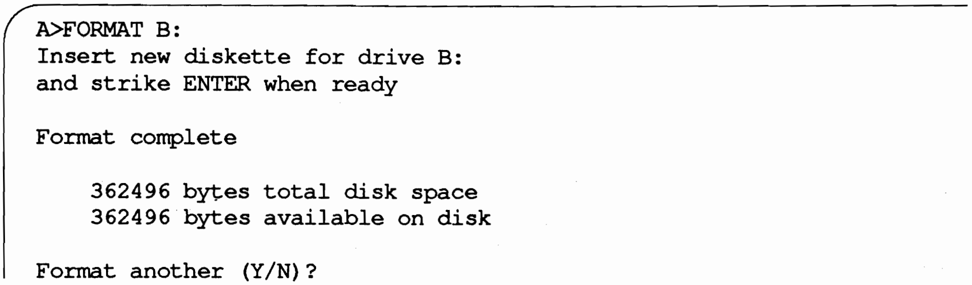
Insert the DOS System diskette in drive A and boot the system. With the cursor positioned after the A> prompt, type

```
FORMAT B:
```

and press ↵. Since FORMAT is an external DOS command, the FORMAT program must be loaded into memory from the DOS System diskette in drive A. Once the loading process is complete, DOS initiates the execution of the FORMAT program, which displays the following message on the screen:

```
Insert new diskette for drive B:
and strike ENTER when ready
```

After you insert a blank (target) diskette in drive B, press ↵. As the diskette in drive B is being formatted, the head number (indicating bottom [0] or top [1] of the diskette) and cylinder number (indicating the track number on the bottom or top of the diskette) are displayed. When the formatting process is complete, FORMAT displays summary information indicating the number of bytes available on the target diskette followed by a prompt message that asks if you want to format another diskette (Fig. 3.4). With the cursor positioned after the prompt message type `Y` and press ↵ to format another diskette. Remove your formatted diskette from drive B, insert another blank diskette in drive B, and press ↵ to begin the formatting process. After formatting several diskettes, type `N` and press ↵ to terminate the formatting process. System control returns to command level, and DOS displays the A > prompt.



```
A>FORMAT B:
Insert new diskette for drive B:
and strike ENTER when ready

Format complete

    362496 bytes total disk space
    362496 bytes available on disk

Format another (Y/N)?
```

Figure 3.4 When the FORMAT command is used to format a 360KB diskette, messages similar to the above are displayed on the screen. Note that once the formatting process is complete, the head number and cylinder number are no longer displayed.

When the FORMAT command is used without any special formatting parameters, the diskette drive type dictates the diskette format. That is, a 360KB diskette drive formats a 360KB diskette, a 1.2MB diskette drive formats a 1.2MB diskette, a 720KB diskette drive formats a 720KB diskette, and a 1.44MB diskette drive formats a 1.44MB diskette, unless the FORMAT command specifies an alternate diskette format. There are FORMAT parameters that allow you to specify different capacities of a formatted diskette (see the FORMAT command in Appendix B). For example, you can format a diskette on only one side (single-sided) or eight sectors per track.

If you are using a computer system with a 1.2MB (high-density) diskette drive and a 360KB (double-density) diskette drive, you need to be careful how you use these two types of drives. A 360KB diskette drive cannot read from or write to a formatted 1.2MB diskette. A 1.2MB diskette drive can read from and write to a formatted 360KB diskette. However, while 360KB diskettes written on by a 1.2MB drive can be reliably read in another 1.2MB drive, they may not be reliably read in a 360KB diskette drive. For this reason you should never format a 360KB diskette in a 1.2MB diskette drive. Similarly, a 720KB diskette drive cannot read from or write to a formatted 1.44MB diskette, but a 1.44MB diskette drive can read from and write to a formatted 720KB diskette. Also, 720KB diskettes written on by a 1.44MB drive can be reliably read by both 720KB and 1.44MB diskette drives.

Keystroke Summary: Formatting a Diskette

Insert:	DOS System diskette in drive A
Set:	default drive to drive A
Type:	`FORMAT B:`
Press:	↵
Insert:	blank (target) diskette in drive B

Press: ↵

Type: Y to format another diskette or

N to terminate the FORMAT command

Press: ↵

What Can Go Wrong?

1. The FORMAT command displays the following information that includes a message about bad sectors:

```
Format complete

362496 bytes total disk space
  5120 bytes in bad sectors
357376 bytes available on disk
```

Cause: When the FORMAT program finds parts (sectors) of a disk that cannot be written to and read from in a reliable manner, it marks these parts (sectors) as write protected so that they cannot be used by DOS to store information.

Solution: When the FORMAT command displays bad sector information for a diskette, you should format the diskette again to verify that the diskette does have bad sectors. As a general rule, do not use diskettes that contain bad or defective sectors. Not only do bad sectors decrease the space available to store information, they also raise a question about the quality and reliability of the diskette's surface. A diskette that contains bad sectors should be discarded.

2. The FORMAT command displays the following message after it attempts to format a diskette:

```
Attempted write-protect violation
Format failure
Format another (Y/N)?
```

Cause: The FORMAT command cannot format the target diskette because it is not inserted in the diskette drive with the drive latch closed or it is write protected.

Solution: Check to be sure that the target diskette is inserted in the target drive with the drive latch closed. If it is, remove the target diskette from the diskette drive and check to see whether the diskette is write protected. If you determine that the diskette is write protected and is an important diskette that should not be formatted, replace this diskette with a blank diskette to be formatted. Otherwise, change the write-protect status of the diskette before attempting to format it again. This is accomplished by removing the write-protect tab from a 5.25 inch diskette or by closing the write-protect window on a 3.50 inch diskette.

3. The FORMAT command displays the following message after it attempts to format a diskette:

```
Invalid media or Track 0 bad - disk unusable
Format failure
Format another (Y/N)?
```

Cause: The FORMAT command cannot format the target diskette because it has a bad track 0 on side 0 (bottom side).

Solution: Remember that one of the tasks performed by the FORMAT command is to place the boot record on the initial track of the diskette. Consequently, when the first track (track 0) of the bottom side (side 0) of the diskette is bad, the diskette cannot be formatted. A diskette that causes this message to be displayed should be discarded.

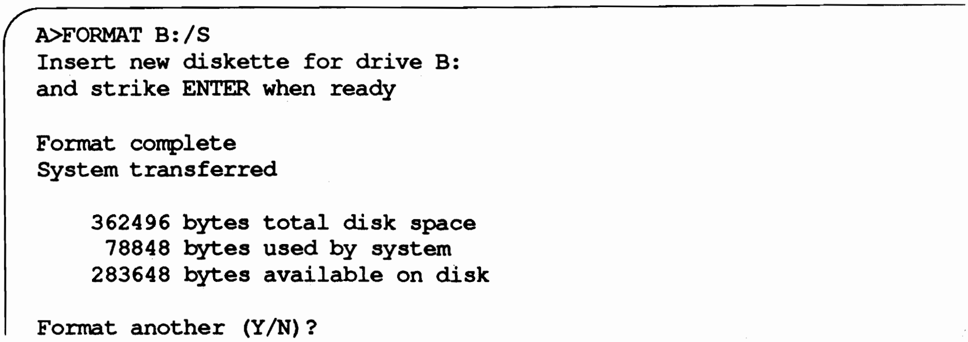
Preparing Bootable Diskettes (FORMAT /S)

The previous section showed you how to format a diskette. However, you cannot boot the system with a diskette that has only been formatted. In order for you to boot the system from a diskette, the diskette must contain the three DOS system files. That is, a bootable diskette must contain the two hidden system files IBMBIO.COM and IBMDOS.COM and the visible system file COMMAND.COM. The **/S** (system) parameter can be used with the FORMAT command to allow you to prepare a bootable diskette.

Insert the DOS System diskette in drive A. With the cursor positioned after the A> prompt, type

```
FORMAT B: /S
```

and press ↵. After the FORMAT program is loaded into memory from the DOS System diskette in drive A, it displays a message that asks you to insert a new diskette in drive B (Fig. 3.5). Insert a blank (target) diskette in drive B and press ↵. As soon as the target diskette is formatted, the /S parameter instructs the FORMAT program to transfer the three DOS system files from the diskette in drive A to the target diskette in drive B. When the transfer of the DOS system files is complete, FORMAT displays summary information indicating the number of bytes available on the target diskette followed by a prompt message that asks whether you want to format another diskette (Fig.



```
A>FORMAT B:/S
Insert new diskette for drive B:
and strike ENTER when ready

Format complete
System transferred

    362496 bytes total disk space
     78848 bytes used by system
    283648 bytes available on disk

Format another (Y/N)?
```

Figure 3.5 When the FORMAT command is used with the /S (system) parameter to format a 360KB diskette, messages similar to the above are displayed on the screen. Note that the three DOS system files occupy 78,848 bytes on the target diskette.

3.5). Type `Y` and press ↵ to repeat the above process, or type `N` and press ↵ to terminate the FORMAT command.

Note that a space does not have to be used before the slash (/) in the above FORMAT command. Since a slash is interpreted by DOS as a parameter field delimiter (separator), the space before the /S parameter is used only to help make the command more readable.

Any diskette that is formatted by the FORMAT command with the /S parameter can be used to boot the system, since it contains the three DOS system files. The disadvantage of having the three DOS system files on a diskette is that they occupy some of the diskette's usable space. As a general rule, you might want to put the DOS system files on your program diskettes but not on your data diskettes.

Keystroke Summary: Preparing a Bootable Diskette

Insert:	DOS System diskette in drive A
Set:	default drive to drive A
Type:	`FORMAT B: /S`
Press:	↵
Insert:	blank (target) diskette in drive B
Press:	↵
Type:	`Y` to prepare another bootable diskette or
	`N` to terminate the FORMAT command
Press:	↵

Transferring DOS to a Formatted Diskette (SYS)

The /S option on the FORMAT command is not the only method for creating a bootable diskette. The steps outlined in this section give you a process for installing the three DOS system files on a formatted diskette that contains no volume label and no files, that is, on a formatted diskette that contains no user information. The **SYS** (SYStem) command is an external DOS command that is used to copy the two DOS hidden files (IBMBIO.COM and IBMDOS.COM) to a formatted diskette. For this process to work, the formatted diskette must have an empty directory or the diskette must have been originally formatted with the /S or /B parameter. (See the FORMAT command in Appendix B for information about the / B parameter.)

Insert the DOS System diskette in drive A and insert the formatted diskette with an empty directory in drive B. With the cursor positioned after the A > prompt, type

```
SYS B:
```

and press ↵. After the SYS program is loaded into memory from the DOS System diskette in drive A, it checks the directory of the diskette in drive B. If the directory of the diskette in drive B is empty, SYS instructs DOS to copy the two DOS hidden files IBMBIO.COM and IBMDOS.COM from the DOS System diskette in drive A to the formatted diskette in drive B. When this process is complete, SYS displays the message

```
System transferred
```

before returning to command level with the A> prompt.

At this point, two-thirds of the DOS system files have been transferred to the formatted diskette. The third DOS system file (COMMAND.COM) must now be copied to the formatted diskette. With the cursor positioned after the A > prompt, type

```
COPY COMMAND.COM B:
```

and press ↵. This command copies the DOS file COMMAND.COM from the DOS System diskette in drive A to the diskette in drive B. When this process is complete, the COPY command displays the message

```
1 File(s) copied
```

before returning to command level with the A > prompt. (A comprehensive discussion of the COPY command is given in Chapter 5.) Since the diskette in drive B now contains the three DOS system files, it can be used to boot the system; that is, it is a bootable diskette.

Keystroke Summary: Transferring DOS to a Formatted Diskette

Insert:	DOS System diskette in drive A
Set:	default drive to drive A

Insert: formatted diskette with an empty directory in drive B

Type: `SYS B:`

Press: ↵

Type: `COPY COMMAND.COM B:`

Press: ↵

What Can Go Wrong?

1. The SYS command displays the following message and returns to command level with the A > prompt after it attempts to transfer the two DOS hidden files to a formatted diskette:

```
No room for system on destination disk
```

Cause: The SYS command cannot copy the two DOS hidden files to the formatted diskette because the directory of the formatted diskette is not empty.

Solution: In Chapters 4 and 5 you will learn about DOS commands that might be used to help correct this situation. At this point, however, the DOS hidden files cannot be transferred to this diskette. Consequently, you cannot make this diskette bootable.

2. The SYS command displays the following message after it attempts to transfer the two DOS hidden files to a diskette:

```
Insert destination disk in drive B
and strike any key when ready
```

Cause: The SYS command cannot copy the two DOS hidden files to the target diskette because it is not a formatted diskette.

Solution: Remove the unformatted target diskette from the diskette drive, insert a formatted diskette with an empty directory in the diskette drive, and press ↵. If a formatted diskette with an empty directory is not available for your use, you need to terminate the SYS command. This can be accomplished by holding the Ctrl key down and typing C (denoted by Ctrl *and* C). This process is called **breaking out of a command** and may also be accomplished by holding the Ctrl key down and pressing Break (denoted by Ctrl *and* Break). Notice that DOS displays the characters ^C on the screen and returns you to command level with the A > prompt. Actually, the Ctrl *and* C and Ctrl *and* Break key combinations give you a way to abort (perform a nonstandard termination of) a DOS command.

Formatting a Diskette with a Volume Label (FORMAT /V)

When you are working with several diskettes, you might want to assign a name to each of your diskettes to assist you with diskette identification. There are two ways that you can label a diskette. One is by writing information on a paper label and attaching (sticking) it to the diskette jacket. This is an external label that can be read by you when the diskette is not in a diskette drive. The second way is to have DOS place an identifying name (label) in the directory of the diskette. When this is done, you can direct DOS to display the diskette name on the screen (Chapter 4). This allows you to identify a diskette in a particular diskette drive without having to remove the diskette from that drive to read the paper label. This also allows you to identify a diskette that has lost its paper label. A name assigned to a disk by DOS is commonly referred to as a **volume label**. A volume label consists of one to eleven filename characters. You can also include one or more space characters in a volume label. The / V (volume label) parameter can be used with the FORMAT command to allow you to format a diskette and assign a volume label to the diskette. Additional DOS commands will be introduced in Chapter 4 that allow you to display, assign, change, and remove a volume label of a disk.

Insert the DOS System diskette in drive A. With the cursor positioned after the A > prompt, type

```
FORMAT B: /V
```

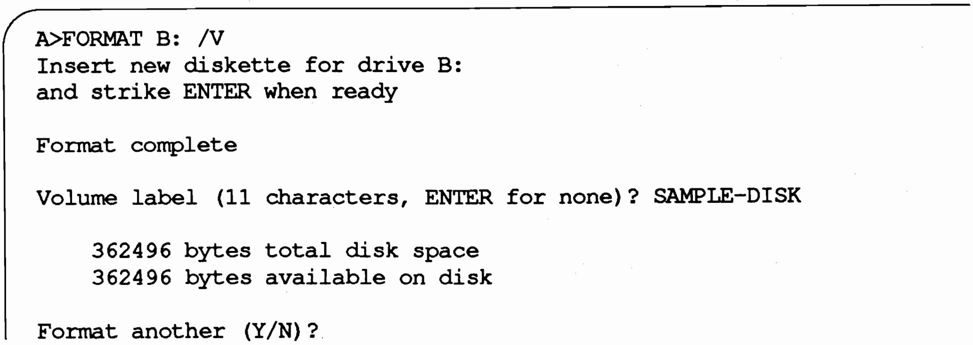
and press ↵. After the FORMAT program is loaded into memory from the DOS System diskette in drive A, it displays a message that asks you to insert a new diskette in drive B. Insert a blank (target) diskette in drive B and press ↵. As soon as the target diskette has been formatted, the / V parameter instructs the FORMAT program to display the following prompt message for a volume label:

```
Volume label (11 characters, ENTER for none)?
```

With the cursor positioned after the question mark (?), type the volume label `SAMPLE-DISK` (or a name of your choice) and press ↵. If you press ↵ without typing any characters, no volume label will be placed on the diskette. When a volume label is entered, the FORMAT command places the volume label in the diskette's directory and then displays summary information indicating the number of bytes available on the target diskette followed by a prompt message that asks if you want to format another diskette (Fig. 3.6). Type `Y` and press ↵ to repeat the above process, or type `N` and press ↵ to terminate the FORMAT command.

Again recall that a space is not necessary before the slash (/) in the above FORMAT command. Since a slash is interpreted by DOS as a parameter field delimiter (separator), the space before the / V parameter is used only to help make the command more readable.

If you want to prepare a bootable diskette with a volume label, you need to



```
A>FORMAT B: /V
Insert new diskette for drive B:
and strike ENTER when ready

Format complete

Volume label (11 characters, ENTER for none)? SAMPLE-DISK

    362496 bytes total disk space
    362496 bytes available on disk

Format another (Y/N)?
```

Figure 3.6 When the FORMAT command is used with the /V (volume label) parameter to format a 360KB diskette, messages similar to the above are displayed on the screen.

insert the DOS System diskette in drive A. With the cursor positioned after the A > prompt, type

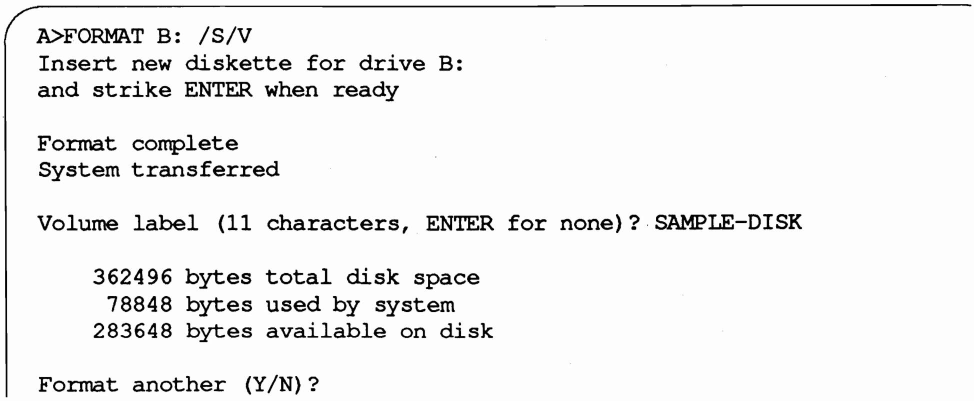
```
FORMAT B: /S /V
```

and press ↵. The prompts and messages generated by this FORMAT command are the same as those described above and in a previous section (Fig. 3.7). Consequently, they are not repeated here.

Keystroke Summary: Formatting a Diskette with a Volume Label

Insert: DOS System diskette in drive A

Set: default drive to drive A



```
A>FORMAT B: /S/V
Insert new diskette for drive B:
and strike ENTER when ready

Format complete
System transferred

Volume label (11 characters, ENTER for none)? SAMPLE-DISK

    362496 bytes total disk space
     78848 bytes used by system
    283648 bytes available on disk

Format another (Y/N)?
```

Figure 3.7 When the FORMAT command is used with the /S (system) and /V (volume label) parameters to format a 360KB diskette, messages similar to the above are displayed on the screen.

Type:	`FORMAT B: /V` or `FORMAT B: /S /V`
Press:	↵
Insert:	blank (target) diskette in drive B
Press:	↵
Type:	*volume-label*
Press:	↵
Type:	`Y` to prepare another diskette with a volume label or
	`N` to terminate the FORMAT command
Press:	↵

Summary

Topic or Feature	Command or Reference	Page
Format a diskette	`FORMAT d:` where *d:* denotes the specified drive	33
	Fig. 3.4	34
Format a bootable diskette	`FORMAT d: /S` where *d:* denotes the specified drive	36
	Figs. 3.5, 3.7	37, 41
Format a diskette with a volume label	`FORMAT d: /V` where *d:* denotes the specified drive	40
	Figs. 3.6, 3.7	41
Format a bootable diskette with a volume label	`FORMAT d: /S /V` where *d:* denotes the specified drive	41
	Fig. 3.7	41
Transfer DOS to a formatted diskette	`SYS d:` and `COPY COMMAND.COM d:` where *d:* denotes the specified drive	38

Self-Test

1. Before a disk can be used by DOS, it must be ______________. This process is accomplished by using the ______________ command. Is this command an internal or external DOS command? ______________
2. Addressing a disk for use by DOS is accomplished by subdividing the disk into concentric circles called ______________ and into pie-shaped areas called ______________.

3. List five things that are done to a disk to prepare it for use by DOS.
   ____
4. What does it mean to say that a diskette is write protected? ____________
   How do you write protect a diskette? ____________________
5. The system can be booted from a disk that contains the ____________ files.
   Name these files. ______________________________
6. The /S parameter is used with the FORMAT command to ____________ .
7. The /V parameter is used with the FORMAT command to ____________ .
8. Can the /S and /V parameters be used together with the FORMAT command?
   ____
9. The ____________ command is used to transfer the two DOS hidden files to a formatted diskette with an ____________ directory. Is this command an internal or external DOS command? ____________

Exercises

1. Use an appropriate DOS command to perform each of the assigned tasks.
   a. Format a data diskette with volume label *namef*#3-1a, where *name* denotes the first five characters of your last name or your entire last name (whichever is shorter) and *f* denotes the first character of your first name.
   b. Format a data diskette with volume label *namef*#3-1b, where *namef* is defined above.
2. Use the FORMAT command to format a bootable diskette with volume label BOOT-DISK. Use this diskette to boot the system.
3. Use an appropriate DOS command to perform each of the assigned tasks.
   a. Format a diskette without a volume label.
   b. Use the diskette prepared in part (a) to attempt to boot the system. (Remember that this diskette must be inserted in drive A.) Note the message displayed on the screen.
   c. Use the DOS System diskette to boot the system.
   d. Transfer the two DOS hidden files from the DOS System diskette to the formatted diskette.
   e. Use this modified diskette to attempt to boot the system. Note the message displayed on the screen.
   f. Use the DOS System diskette to boot the system.
   g. Copy the file COMMAND.COM from the DOS System diskette to the modified diskette.
   h. Use this modified diskette to boot the system.

   (This exercise shows that the three DOS System files must be on a diskette before it can be used to boot the system.)

4

Displaying Information about a Diskette or Disk

The material in this chapter introduces the DOS commands that allow you to:

- display the contents of a disk's directory using the DIR command
- display directory information for selected files using the wildcard file name characters * and ? with the DIR command
- send directory information (output of the DIR command) to the printer using the output redirection operator >
- check a disk and memory for important information using the CHKDSK command
- display and modify the volume label of a disk using the VOL and LABEL commands

Displaying the Table of Contents (Directory Information) of a Disk

In Chapter 3 you learned that a disk directory is created by the FORMAT command during the formatting process and serves as a table of contents for the information stored on a disk. Since information stored on a disk is stored as a file, DOS keeps track of this information through file entries in a directory. Each file entry contains a name for the file, an address (side, track, and sector numbers) specifying where the file is placed on the disk, the size of the file in bytes, and a date and time indicating when the file was created or last modified. The **DIR** (DIRectory) command is an internal DOS command that allows you to display the contents of a disk's directory on the screen.

Displaying Complete Directory Information (DIR)

Insert the DOS System diskette in drive A and boot the system. Then insert any usable diskette in drive A. This diskette does not have to be the DOS System diskette. With the cursor positioned after the A > prompt, type

```
        DIR A:
```

and press ↵. Since the DIR command is an internal DOS command, the DOS System diskette does not have to be available to execute this command. The DIR command instructs DOS to display the complete directory information from the diskette in drive A, the specified drive. When the DIR command completes its task, control returns to command level with the A > prompt. The information displayed on the screen will be similar to the example given in Fig. 4.1. Figure 4.1 illustrates several items that the DIR command displays from the directory of a disk. First, a heading message is displayed that contains the volume label for the disk. In Fig. 4.1, SAMPLE-DISK is the volume label of the diskette. If a disk contains no volume label, heading information similar to the following will be displayed by the DIR command:

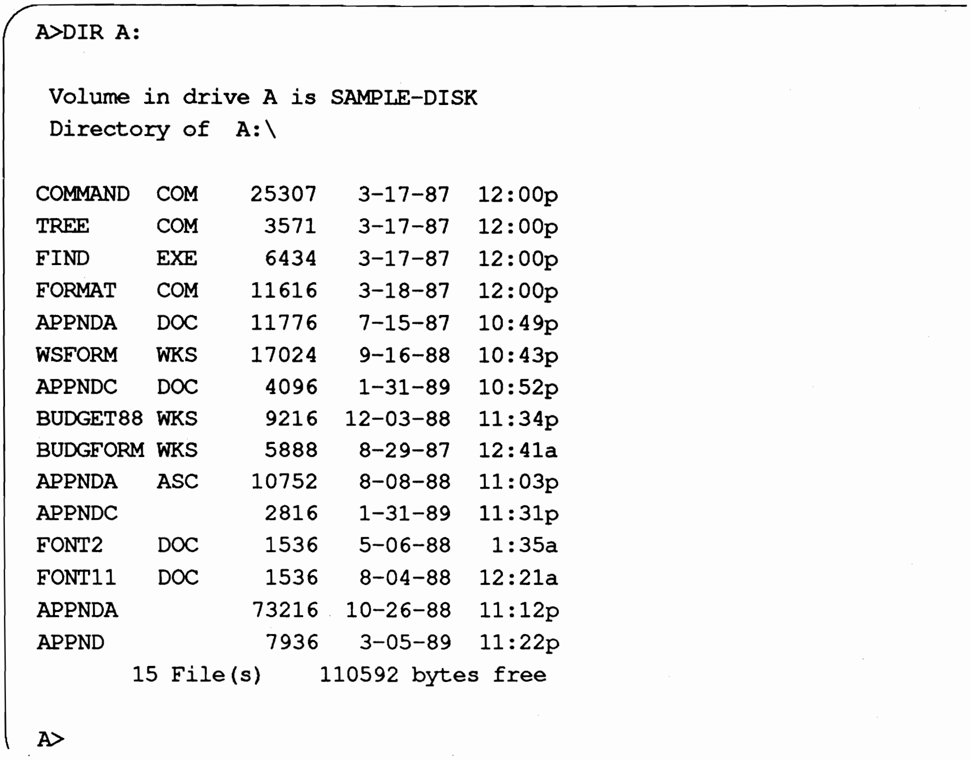
```
        Volume in drive A has no label
        Directory of  A:\
```

Notice from this and from Fig. 4.1 that the heading information displayed by the DIR command also includes a message identifying the path to the directory. In this case the path includes the disk drive specification A: followed by a backslash (\). A:\ specifies the root (main) directory of the diskette in drive A. You will learn in Chapter 8 that a disk can contain subdirectories of the root (main) directory and that directory paths are used to access these subdirectories.

Immediately following the heading information from the DIR command is a list of file names for all visible (user) files that are contained in the disk's directory. Notice from the example in Fig. 4.1 that each line contains a filename followed by the filename extension (if any), the number of bytes in the file, and a date and time indicating when the file was created or last modified. At the conclusion of the list of files, the DIR command displays a summary line similar to the following:

```
        27 File(s)     79872 bytes free
```

This line contains the number of files listed from the directory and the number of free bytes available for use on the disk. When the DIR command is



```
A>DIR A:

 Volume in drive A is SAMPLE-DISK
 Directory of  A:\

COMMAND  COM     25307   3-17-87  12:00p
TREE     COM      3571   3-17-87  12:00p
FIND     EXE      6434   3-17-87  12:00p
FORMAT   COM     11616   3-18-87  12:00p
APPNDA   DOC     11776   7-15-87  10:49p
WSFORM   WKS     17024   9-16-88  10:43p
APPNDC   DOC      4096   1-31-89  10:52p
BUDGET88 WKS      9216  12-03-88  11:34p
BUDGFORM WKS      5888   8-29-87  12:41a
APPNDA   ASC     10752   8-08-88  11:03p
APPNDC            2816   1-31-89  11:31p
FONT2    DOC      1536   5-06-88   1:35a
FONT11   DOC      1536   8-04-88  12:21a
APPNDA           73216  10-26-88  11:12p
APPND             7936   3-05-89  11:22p
       15 File(s)    110592 bytes free

A>
```

Figure 4.1 When the DIR command is used to display the directory contents of a 360KB diskette in drive A, information similar to the above is displayed on the screen.

finished listing the directory information, control returns to command level, and the A > prompt is displayed on the screen.

Remove the diskette from drive A and insert it in drive B. With the cursor positioned after the A > prompt, type

```
DIR B:
```

and press ⏎. The DIR command instructs DOS to display the complete directory information from the diskette in drive B, the specified drive.

Remove the diskette from drive B and insert it into drive A. With the cursor positioned after the A > prompt, type

```
DIR
```

and press ⏎. Notice that the DIR command instructs DOS to display the complete directory information from the disk in the default drive—in this case, drive A. If you want to display the directory contents of a disk in the default drive, you do not need to specify a disk drive after the DIR command.

Keystroke Summary: Displaying Complete Directory Information

Insert:	any usable diskette in drive A
Set:	default drive to drive A
Type:	`DIR A:` or `DIR`
Press:	↵

Pausing after Each Screen Image (DIR /P)

Insert the DOS System diskette in drive A. With the cursor positioned after the A > prompt, type

```
DIR
```

and press ↵. Notice that the entries from the directory of the DOS System diskette cannot all be displayed in one screen image. Some of the directory information scrolls off of the top of the screen before it can be read. There are two ways to pause the output to the screen so that you have an opportunity to read it before it moves off of the top of the screen. One method involves sending an interrupt signal to DOS instructing it to suspend output to the screen until it receives a second command telling it to continue sending information to display on the screen. An **interrupt signal** is a signal that causes a break in the normal sequence of executing instructions. Interrupting the flow of information to the screen is accomplished by holding the Ctrl key down and typing S (denoted by Ctrl *and* S).

With the DOS System diskette inserted in drive A and with the cursor positioned after the A > prompt, type

```
DIR
```

and press ↵. As the directory information begins to be displayed on the screen, hold the Ctrl key down and type S. Notice that the output to the screen immediately stops but will continue when you type S again. Actually, you instruct DOS to suspend output to the screen by holding the Ctrl key down and typing S; you instruct DOS to resume sending output to the screen by typing any character key on the keyboard. Note that holding the Ctrl key down and then pressing the Num Lock key (denoted by Ctrl *and* Num Lock) has a similar effect to the Ctrl *and* S key combination. If a personal computer has an enhanced IBM PC AT keyboard (a 101 style keyboard), then pressing the Pause key on this keyboard has a similar effect to the Ctrl *and* S key combination.

The second way to pause output from the DIR command to the screen is to use the /P (pause) parameter in the DIR command. With the DOS System diskette inserted in drive A and with the cursor positioned after the A >

prompt, type

```
DIR /P
```

and press ⏎. Note that you do not need to specify A: in the DIR command (DIR A: /P), since drive A is the default drive. The /P parameter instructs the DIR command to pause the directory output listing after a full screen of directory entries is displayed. After a full screen of directory information is displayed, the directory output pauses, and the DIR command displays a message similar to the following:

```
Strike a key when ready . . .
```

Type any character key or press ⏎, and the directory entries continue to be displayed on the screen, pausing each time the screen is filled with new information. This process may need to be repeated several times for directories with a large number of entries.

Note that a space does not have to be used before the slash (/) in the above DIR command. Since a slash is interpreted by DOS as a parameter field delimiter (separator), the space before the /P parameter is used only to help make the command more readable.

Keystroke Summary: Displaying Directory Information One Screen at a Time

Insert:	a diskette in drive A that contains a directory with a large number of entries (for example, the DOS System diskette)
Set:	the default drive to drive A
Type:	`DIR /P`
Press:	⏎

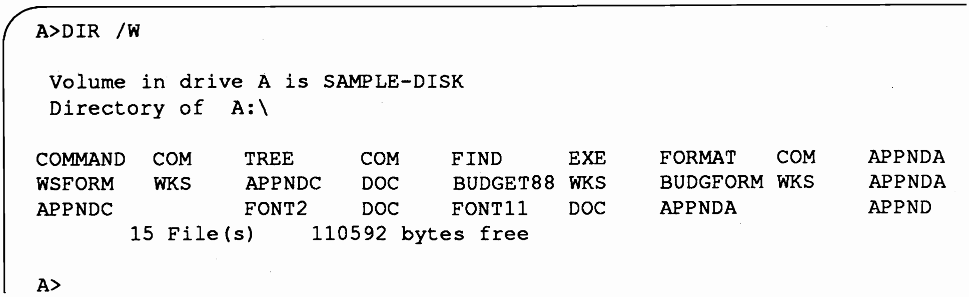
Displaying Abbreviated Directory Information (DIR /W)

The DIR command can also be used to display the directory entries of a disk in an abbreviated format by using the / **W** (wide-line) parameter in the DIR command. The / W parameter instructs the DIR command to display only the filenames and filename extensions in a wide-line format on the screen.

Insert any usable diskette in drive A. This diskette does not have to be the DOS System diskette. With the cursor positioned after the A > prompt, type

```
DIR /W
```

and press ⏎. Again note that you do not need to specify A: in the DIR command (DIR A: /W), since drive A is the default drive. The / W parameter instructs the DIR command to display only the filenames and extensions, from the directory of the diskette in drive A, on the screen in a wide-line format. The format of the displayed information is similar to the example given in Fig. 4.2.



```
A>DIR /W

 Volume in drive A is SAMPLE-DISK
 Directory of  A:\

COMMAND  COM     TREE     COM     FIND     EXE     FORMAT   COM     APPNDA   DOC
WSFORM   WKS     APPNDC   DOC     BUDGET88 WKS     BUDGFORM WKS     APPNDA   ASC
APPNDC           FONT2    DOC     FONT11   DOC     APPNDA           APPND
       15 File(s)    110592 bytes free

A>
```

Figure 4.2 When the DIR command is used with the /W (wide-line) parameter to display the file names from the directory of a 360KB diskette in drive A, information similar to the above is displayed on the screen.

Keystroke Summary: Displaying Directory Information in a Wide-Line Format

Insert:	any usable diskette in drive A
Set:	default drive to drive A
Type:	`DIR /W`
Press:	↵

Displaying Directory Information for Selected Files

The DIR command can be used to display directory information about a single file or about groups of files having similar filenames and/or extensions. The wildcard file name characters * and ?, introduced in Chapter 1, are used extensively to specify groups of files in the DIR command. Several examples are given in this section to illustrate the use of the DIR command to display directory information for selected files.

Insert the DOS System diskette in drive A. With the cursor positioned after the A > prompt, type

```
DIR FORMAT.COM
```

and press ↵. This DIR command displays one file name, namely, FORMAT.COM, from the directory of the diskette in the default drive (drive A). Directory information similar to the following is displayed on the screen before the system returns to command level with the A > prompt:

```
 Volume in drive A has no label
 Directory of  A:\

FORMAT   COM    11616   3-18-87  12:00p
        1 File(s)     79872 bytes free
```

This DIR command illustrates that you can request directory information about a single specified file—in this case, FORMAT.COM in the directory of the DOS System diskette.

With the cursor positioned after the A > prompt, type

```
DIR C*
```

and press ↵. The C* specification instructs the DIR command to display all file names in the diskette's directory of the default drive (drive A) that begin with the character C and have any extension. Directory information similar to the following is displayed on the screen:

```
 Volume in drive A has no label
 Directory of  A:\

COMMAND   COM     25307   3-17-87  12:00p
CHKDSK    COM      9850   3-18-87  12:00p
COMP      COM      4214   3-17-87  12:00p
        3 File(s)      79872 bytes free
```

This DIR command illustrates that you can request directory information about all file names that begin with the same first character and have any extension—in this case, all file names in the directory of the DOS System diskette that begin with the character C.

With the cursor positioned after the A > prompt, type

```
DIR *.EXE
```

and press ↵. The *.EXE specification instructs the DIR command to display all file names in the diskette's directory of the default drive (drive A) that have .EXE as an extension. Directory information similar to the following is displayed on the screen:

```
 Volume in drive A has no label
 Directory of  A:\

FASTOPEN EXE       3919   3-17-87  12:00p
REPLACE   EXE     11775   3-17-87  12:00p
APPEND    EXE      5825   3-17-87  12:00p
ATTRIB    EXE      9529   3-17-87  12:00p
FIND      EXE      6434   3-17-87  12:00p
JOIN      EXE      8969   3-17-87  12:00p
SHARE     EXE      8608   3-17-87  12:00p
SORT      EXE      1977   3-17-87  12:00p
SUBST     EXE      9909   3-17-87  12:00p
XCOPY     EXE     11247   3-17-87  12:00p
       10 File(s)      79872 bytes free
```

This DIR command illustrates that you can request directory information about all file names that have the same extension—in this case, all file names in the directory of the DOS System diskette that have .EXE as an extension.

Remove the DOS System diskette from drive A and insert it in drive B. With the cursor positioned after the A > prompt, type

```
DIR B:F*.COM
```

and press ⏎. The B:F*.COM specification instructs the DIR command to display all file names in the diskette's directory of the specified drive (drive B) that begin with the character F and have .COM as an extension. Directory information similar to the following is displayed on the screen:

```
 Volume in drive B has no label
 Directory of  B:\

FORMAT   COM    11616   3-18-87  12:00p
        1 File(s)     79872 bytes free
```

This DIR command illustrates that you can request directory information about all file names that begin with the same first character and have the same extension—in this case, all file names in the directory of the DOS System diskette that begin with the character F and have .COM as an extension.

Assume that SAMPLE-DISK denotes the diskette used to generate the information in Fig. 4.1 and that SAMPLE-DISK has been inserted in drive A. The next few examples illustrate the effect of the wildcard file name characters * and ? when they are used in the DIR command to access the directory of SAMPLE-DISK.

With the cursor positioned after the A > prompt, type

```
DIR BUDG*.*
```

and press ⏎. The BUDG*.* specification instructs the DIR command to display all file names in the directory of SAMPLE-DISK that begin with the same first few characters—in this case, BUDG—and have any extension. This DIR command would display the following directory information from SAMPLE-DISK:

```
 Volume in drive A is SAMPLE-DISK
 Directory of  A:\

BUDGET88 WKS     9216  12-03-88  11:34p
BUDGFORM WKS     5888   8-29-87  12:41a
        2 File(s)    110592 bytes free
```

With the cursor positioned after the A > prompt, type

```
DIR *.
```

and press ⏎. The *. specification instructs the DIR command to display all file names in the directory of SAMPLE-DISK that have no extension. This DIR command would display the following directory information from SAMPLE-DISK:

```
 Volume in drive A is SAMPLE-DISK
 Directory of  A:\

APPNDC           2816   1-31-89  11:13p
APPNDA          73216  10-26-88  11:12p
APPND            7936   3-05-89  11:22p
        3 File(s)    110592 bytes free
```

With the cursor positioned after the A > prompt, type

```
DIR APPNDA.*
```

and press ⏎. The APPNDA.* specification instructs the DIR command to display all file names in the directory of SAMPLE-DISK that have the same filename—in this case, APPNDA—and any extension. This DIR command would display the following directory information from SAMPLE-DISK:

```
 Volume in drive A is SAMPLE-DISK
 Directory of  A:\

APPNDA   DOC    11776   7-15-87  10:49p
APPNDA   ASC    10752   8-08-88  11:03p
APPNDA          73216  10-26-88  11:12p
        3 File(s)    110592 bytes free
```

With the cursor positioned after the A > prompt, type

```
DIR APPND?
```

and press ⏎. The APPND? specification instructs the DIR command to display all file names in the directory of SAMPLE-DISK that begin with the same first five characters—in this case, APPND—, have at most six characters in the filename, and have any extension. This DIR command would display the following directory information from SAMPLE-DISK:

```
 Volume in drive A is SAMPLE-DISK
 Directory of  A:\

APPNDA   DOC    11776   7-15-87  10:49p
APPNDC   DOC     4096   1-31-89  10:52p
APPNDA   ASC    10752   8-08-88  11:03p
APPNDC           2816   1-31-89  11:13p
APPNDA          73216  10-26-88  11:12p
APPND            7936   3-05-89  11:22p
        6 File(s)    110592 bytes free
```

What Can Go Wrong?

1. The DIR command displays the following message when it reads the directory of a disk:

```
File not found
```

Cause: The directory of the disk contains no user files or it contains no file names as specified in the DIR command.

Solution: Check the DIR command line to see whether it contains any errors in the file specification. For example, you might have typed V* instead of C* in the command. If the DIR command that you entered is correct, then either the disk's directory contains no user files or it contains no file names that satisfy the file specification in the DIR command.

2. The DIR command displays the following message when it attempts to read the directory of a diskette:

```
General Failure error reading drive A
Abort, Retry, Fail?
```

Cause: The diskette is not formatted, or it is formatted in a form that is incompatible with the diskette drive that is attempting to read it.

Solution: If the diskette is not formatted, then the diskette cannot be accessed by any DOS command. Remember that a diskette must be formatted before it can be used by DOS. If the diskette is formatted, it might have a format that is not compatible with the diskette drive trying to read it. For example, a 1.2MB (high-density) diskette cannot be read by a 360KB diskette drive. If you want the DIR command to attempt to read the directory of the diskette again, type R. If you want to abort the DIR command, type A, and the system returns to command level with the DOS prompt.

Sending Directory Information to the Printer (DIR > PRN)

In the previous sections you noticed that the output of the DIR command is displayed on the screen. DOS gives you the capability to **redirect** (send) the output from the DIR command to the printer to get a **hard copy** (printed copy) of the directory information for a disk. This is accomplished by using the **output redirection operator** > in the DOS command sequence.

Insert the DOS System diskette in drive A of a personal computer that has a printer attached to it. Make sure that your printer is turned on and ready to print. With the cursor positioned after the A> prompt, type

```
DIR > PRN
```

and press ↵. The output redirection operator > instructs DOS to send the output from the DIR command to the printer that is specified by PRN. Notice that the directory information is not displayed on the screen but is printed on the printer. When the printing is completed, DOS returns to command level with the A > prompt.

Keystroke Summary: Sending Directory Information to the Printer

Insert:	any usable diskette in drive A
Ready:	printer
Set:	default drive to drive A
Type:	`DIR > PRN`
Press:	↵

What Can Go Wrong?

1. After you enter the command, `DIR > PRN` , your computer system appears to stop and will not respond to anything that you type on the keyboard. After approximately one minute, the following message is displayed on the screen:

```
Not ready error writing device PRN
Abort, Retry, Ignore, Fail?
```

Cause: Either your personal computer has no printer attached to it or the printer is not ready to print.

Solution: If your personal computer has no printer attached to it, type A after the question mark (?) to abort the DOS command, and the system will return to command level. If your personal computer has a printer attached to it, check the status of the printer. Make sure that the printer is turned on and that it is ready to print. After you have readied the printer, type R to instruct DOS to retry sending the information to the printer.

Checking a Diskette/Disk (CHKDSK)

Even though the DIR command displays a great deal of information about the contents of a disk, it does not give you a complete picture of what is stored on a disk. The **CHKDSK** (CHecK DiSK) command is an external DOS command that can be used to display certain important characteristics of a disk and memory that are not displayed by the DIR command. In fact, the CHKDSK command does several things for you.

1. CHKDSK analyzes the directory of the specified disk, comparing the directory entries with the locations and lengths of the files. It reports any errors that it finds.
2. CHKDSK displays information about how the disk space is being utilized. Information about bad sectors (if they exist) is also reported to you.
3. CHKDSK displays a report on the amount of memory available in your personal computer. The memory report displays the amount of memory in your personal computer that DOS recognizes and the amount of memory that is free for use. The maximum amount of memory that DOS Version 3.30 and previous versions of DOS recognize in an IBM Personal Computer is 640KB of memory. Your personal computer may have additional memory installed that is not recognized by DOS.
4. CHKDSK can be used to display the file names from the directory of a disk for both user and hidden files.

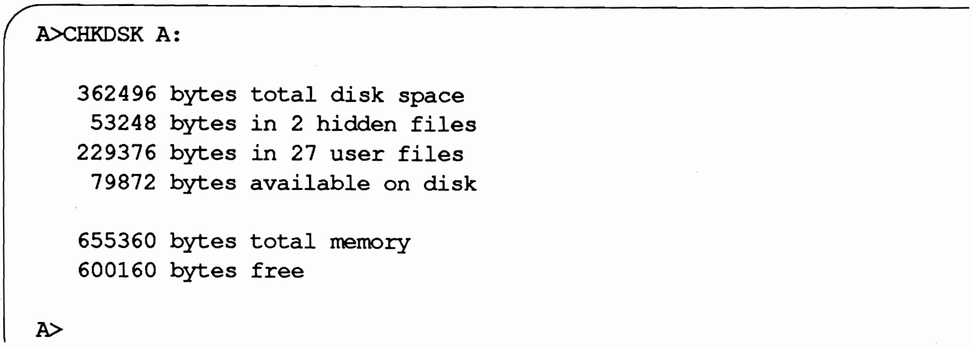
Using CHKDSK to Display Summary Information about a Disk and Memory

Insert the DOS System diskette in drive A. With the cursor positioned after the A > prompt, type

```
        CHKDSK A:
```

and press ↵. Since CHKDSK is an external DOS command, it must be loaded into memory from the DOS System diskette in drive A. Once the loading process is complete, DOS initiates the execution of the CHKDSK program, which analyzes the disk in the specified or default drive—in this case, drive A. When the CHKDSK program completes the check disk process, it displays a summary of the total disk space and a summary of total memory in·your personal computer (Fig. 4.3) before the system returns to command level with the A > prompt. Note that the drive specifier A: in the above command is not necessary, since drive A is the default drive.

The CHKDSK status report in Fig. 4.3 contains two parts; the first part refers to the disk, and the second part refers to the computer's memory. The first part of the status report tells you that the DOS System diskette contains 27 user files occupying 278,528 bytes of disk space, two hidden files occupying 53,248 bytes of disk space, and 79,872 bytes of free space on the disk. If the checked disk had contained any bad sectors, the first part of the status report would also have included a statement specifying the number of bytes in bad sectors.



```
A>CHKDSK A:

   362496 bytes total disk space
    53248 bytes in 2 hidden files
   229376 bytes in 27 user files
    79872 bytes available on disk

   655360 bytes total memory
   600160 bytes free

A>
```

Figure 4.3 When the CHKDSK command is used to check the contents of a disk—in this case, the DOS System diskette—information similar to the above is displayed on the screen.

The second part of the CHKDSK status report states the number of bytes of total memory recognized by DOS and the number of bytes of free memory in your personal computer. The memory information in Fig. 4.3 tells you that the personal computer contains 655,360 bytes of total memory recognized by DOS and that 600,160 bytes of this memory are free. The difference between the bytes of total memory and the bytes of free memory is the amount of memory being used by DOS to house the three DOS system files and certain DOS support programs.

If you want to check the contents of a diskette other than the DOS System diskette, you should perform the following steps. Insert the DOS System diskette in drive A and the diskette you want to check in drive B. With the cursor positioned after the A > prompt, type

```
CHKDSK B:
```

and press ↵. After the CHKDSK program is loaded into memory from the DOS System diskette in drive A, it does a disk check of the disk in the specified drive—in this case, drive B—and displays a summary similar to the one given in Fig. 4.3.

Keystroke Summary: Checking a Diskette and Memory for Summary Information

Insert:	DOS System diskette in drive A
Insert:	diskette to be checked in drive B
Set:	default drive to drive A
Type:	`CHKDSK B:`
Press:	↵

Using CHKDSK to Display Hidden and User File Names

Recall that the DIR command listed all of the user files in a disk's directory but gave you no information about any hidden files. The CHKDSK commands given above gave you the number of hidden files and the number of user files in a disk's directory, but they did not give you the names of these files. To obtain a complete list of the file names for both user and hidden files on a disk, in addition to the information outlined above, you need to use the **/ V** (view) parameter with the CHKDSK command.

Insert the DOS System diskette in drive A. With the cursor positioned after the A > prompt, type

```
CHKDSK /V
```

and press ↵. After the CHKDSK program is loaded into memory from the DOS System diskette in drive A, it does a disk check of the DOS System diskette in drive A and displays the names of all files, hidden and user, from the diskette's directory before it displays the CHKDSK status report. Recall that holding the Ctrl key down while typing S allows you to control the flow of output to the screen. When you do this, you note that the two DOS hidden files, IBMBIO.COM and IBMDOS.COM, are the first two entries in the displayed directory list.

If you want to display the names of all files in the directory of a diskette other than the DOS System diskette, you should perform the following steps. Insert the DOS System diskette in drive A and the diskette you want to check in drive B. With the cursor positioned after the A > prompt, type

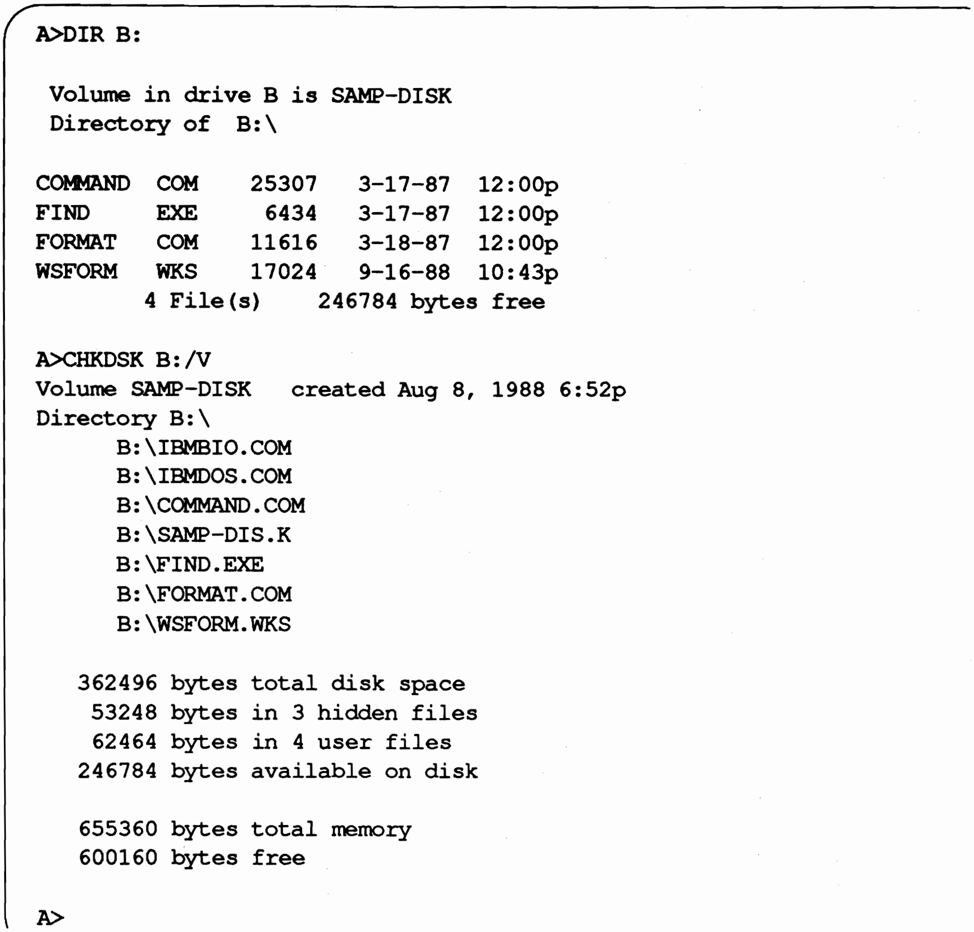
```
CHKDSK B: /V
```

and press ↵. After the CHKDSK program is loaded into memory from the DOS System diskette in drive A, it does a disk check of the disk in the specified drive—in this case, drive B—and displays a summary similar to the one given in the second half of Fig. 4.4. Notice the difference between the information generated by the DIR command and the CHKDSK command in Fig. 4.4. The DIR command displays only directory information about user files. The CHKDSK command with the / V parameter displays the names of all files, both hidden and user, in a disk's directory. Note that the two DOS hidden files, IBMBIO.COM and IBMDOS.COM, are listed and that the volume label of the diskette, SAMP-DISK, is displayed as a hidden file name.

Note that a space does not have to be used before the slash (/) in the above CHKDSK commands. Since a slash is interpreted by DOS as a parameter field delimiter (separator), the space before the / V parameter is used only to help make the command more readable.

Keystroke Summary: Displaying Hidden and User File Names

Insert:	DOS System diskette in drive A
Insert:	diskette to be checked in drive B



```
A>DIR B:

 Volume in drive B is SAMP-DISK
 Directory of  B:\

COMMAND  COM    25307   3-17-87  12:00p
FIND     EXE     6434   3-17-87  12:00p
FORMAT   COM    11616   3-18-87  12:00p
WSFORM   WKS    17024   9-16-88  10:43p
        4 File(s)    246784 bytes free

A>CHKDSK B:/V
Volume SAMP-DISK   created Aug 8, 1988 6:52p
Directory B:\
      B:\IBMBIO.COM
      B:\IBMDOS.COM
      B:\COMMAND.COM
      B:\SAMP-DIS.K
      B:\FIND.EXE
      B:\FORMAT.COM
      B:\WSFORM.WKS

   362496 bytes total disk space
    53248 bytes in 3 hidden files
    62464 bytes in 4 user files
   246784 bytes available on disk

   655360 bytes total memory
   600160 bytes free

A>
```

Figure 4.4 When the DIR command and the CHKDSK command with the /V (view) option are used to display directory information about the same diskette, information similar to the above is displayed on the screen.

Set: default drive to drive A

Type: `CHKDSK B: /V`

Press: ↵

What Can Go Wrong?

1. After you enter the CHKDSK command, the following message is displayed on the screen:

```
Bad command or file name
```

Cause: Either you made an error in typing CHKDSK or the CHKDSK program is not available on the disk in the default or specified drive to load into memory.

Solution: Check the characters you typed for CHKDSK and retype the command if you made an error. If there is no error in what you typed, then the CHKDSK program is not on the disk in the default or specified drive to load into memory. Recall that the CHKDSK command is an external DOS command. Therefore, it must be loaded into memory from a disk before it can be executed. If the CHKDSK program is not on the diskette being accessed, replace that diskette with the DOS System diskette and retype the command.

Diskette/Disk Volume Label

In Chapter 3 you learned that names can be assigned to diskettes to assist you with diskette identification. The / V option was used with the FORMAT command to assign a **volume label** (name) to a diskette at the completion of the formatting process. DOS supports a separate command that allows you to assign a volume label to any diskette or hard disk. In the next two sections you will learn how to display, assign, change, and remove a volume label of a disk.

Displaying a Volume Label (VOL)

The **VOL** (VOLume) command is an internal DOS command that allows you to display the volume identification label of the disk in the specified or default drive.

Insert the DOS System diskette in drive A. With the cursor positioned after the A > prompt, type

```
VOL A:
```

and press ↵. The VOL command looks for a volume label in the directory of the diskette in the specified drive—in this case, drive A. If the diskette contains no volume label, the VOL command displays a message similar to the following:

```
Volume in drive A has no label
```

If the diskette in drive A had contained the volume label SYSTEM-DISK, then the VOL command would have displayed the following message:

```
Volume in drive A is SYSTEM-DISK
```

Note that the drive specifier A: in the above command is not necessary, since drive A is the default drive. To display the volume label of a disk in a different drive, you should perform the following steps. Insert the diskette to be checked for a volume label in drive B. With the cursor positioned after the A > prompt, type

```
VOL B:
```

and press ↵. The VOL command displays one of the above two messages, depending upon whether or not the diskette contains a volume label.

Keystroke Summary: Displaying the Volume Label of a Disk

Insert:	diskette to be checked for a volume label in drive A
Set:	default drive to drive A
Type:	VOL
Press:	↵

Assigning, Changing, and Removing a Volume Label (LABEL)

This section shows you how to assign a volume label to a disk that has been formatted without one, change a volume label that was previously assigned to a disk, and remove a volume label from a disk. The **LABEL** command is an external DOS command that allows you to perform each of these functions to a disk.

Insert the DOS System diskette in drive A. With the cursor positioned after the A > prompt, type

```
LABEL A:
```

and press ↵. After the LABEL program is loaded into memory from the DOS System diskette in drive A, it displays a message similar to the following about the disk in the specified or default drive—in this case, drive A:

```
Volume in drive A has no label
Volume label (11 characters, ENTER for none)?
```

Recall from Chapter 3 that the volume label can be up to eleven characters long, using the same characters that are valid for filenames plus a space. With the cursor positioned after the question mark (?), type `SYSTEM-DISK` (or a label of your choice) and press ↵. The DOS System diskette is assigned the volume label SYSTEM-DISK (or the label of your choice) before the system returns to command level with the A > prompt. To verify that SYSTEM-DISK is the volume label for the diskette in drive A, type

```
VOL
```

and press ↵. The VOL command displays a message similar to the following on the screen:

```
Volume in drive A is SYSTEM-DISK
```

The LABEL command can also be used to change or delete the volume label on a disk. To change or delete the volume label on the DOS System diskette in drive A, perform the following steps. With the cursor positioned after the A> prompt, type

```
LABEL
```

and press ↵. Notice that the drive specification A: does not have to be included in the above command, since drive A is the default drive. After the LABEL program is loaded into memory from the DOS System diskette in drive A, it displays a message similar to the following on the screen:

```
Volume in drive A is SYSTEM-DISK
Volume Label (11 characters, ENTER for none)?
```

To change the volume label, simply enter a different volume label after the question mark (?); for example, type `DOS-SYS-DSK` and press ↵ to change the volume label of the DOS System diskette in drive A to DOS-SYS-DSK. If you wish to delete the SYSTEM-DISK volume label, just press ↵, and the LABEL command displays a message similar to the following on the screen:

```
Delete current volume label (Y/N)?
```

With the cursor positioned after this prompt message, type `Y` and press ↵, and the volume label will be removed from the DOS System diskette in drive A.

Similar steps can be used to assign a volume label to a diskette other than the DOS System diskette. Insert the DOS System diskette in drive A and insert the diskette to which you want to assign a volume label in drive B. With the cursor positioned after the A > prompt, type

```
LABEL B:
```

and press ↵. The remainder of the steps are the same as those already described above.

When you assign a volume label to a disk, you can use the CHKDSK command with the / V option to observe that the volume label has been entered into the directory of the disk as a hidden file entry.

Keystroke Summary: Assigning a Volume Label to a Diskette

Insert:	DOS System diskette in drive A
Set:	default drive to drive A
Insert:	diskette to be assigned a volume label in drive B
Type:	`LABEL B:`
Press:	↵
Type:	*volume-label*
Press:	↵

Summary

Topic or Feature	Command or Reference	Page
Display complete directory information	DIR Fig. 4.1	45 46
Control the flow of directory information to the screen	DIR /P Ctrl *and* S	47 47
Display directory entries in wide-line format	DIR /W Fig. 4.2	48 49
Display directory information for selected files	use of the wildcard file name characters * and ? in the file specification of the DIR command	49
Send directory information to the printer	DIR > PRN	53
Check a disk and memory	CHKDSK Fig. 4.3	55 56
Check a disk and display hidden and user file names	CHKDSK /V Fig. 4.4	57 58
Display a disk volume label	VOL	59
Assign, change, and delete a disk volume label	LABEL	60

Self-Test

1. The ________________ command is used to display information from a disk's directory on the screen. Is this command an internal or external DOS command? ________________ Identify four types of information that this command displays on the screen. ________________
2. State two ways in which the flow of directory information to the screen can be controlled. ________________
3. The /W parameter is used with the DIR command to ________________.
4. Given the DOS prompts below, write the complete DOS command that displays directory information about the following groups of files from the disk in the specified disk drive.
   a. Display all file names in the directory of the diskette in drive A that begin with T and have any extension. A> ________________
   b. Display all file names in the directory of the diskette in drive B that have an extension that begins with WK. A> ________________
   c. Display all file names in the directory of the diskette in drive A that have no extension. B> ________________

d. Display all file names in the directory of the diskette in drive B that have .DOC as an extension. B> ________________

e. Display all file names in the directory of the diskette in drive A that have filename CHAP and any extension. B> ________________

5. The ________________ operator (symbol) is used to redirect output from the DIR command to the printer. Write the DOS command line necessary to send the directory information from the diskette in drive B to the printer.
A> ________________

6. The ________________ command is used to display utilization information about a disk and memory. Is this command an internal or external DOS command? ________________ Name three tasks that this command performs each time it is executed. ________________ This command can be used with the / V parameter to display the file names of both ________________ and ________________files while the DIR command only displays file names for ________________files.

7. The ________________command can be used to display the volume label of a disk on the screen. Is this command an internal or external DOS command?

____

8. The ________________ command can be used to assign, change, and delete a disk volume label. Is this command an internal or external DOS command?

____

Exercises

1. Use an appropriate DOS command to display the following groups of file names from the directory of the DOS System diskette.

    a. File names for all user files.

    b. File names for all user files that have .COM as an extension.

    c. File names for all user files that have .EXE as an extension.

    d. File names for all user files that begin with the character F and have .COM as an extension.

    e. File names for all user files that begin with the character S.

    f. File names for all files both user and hidden.

2. Use the output redirection operator to obtain a hard copy listing for each of the sets of file names generated in Exercise 1.

3. Turn the printer echo mode on so that you can obtain a hard copy listing for each of the sets of file names generated in Exercise 1. Then turn the printer echo mode off.

4. a. Use a DOS command to display the amount of memory that DOS recognizes in your personal computer and the amount of memory that is free for use by an application program.

    b. Use the output redirection operator to obtain a hard copy listing of the information generated from the command in part (a).

5. Use an appropriate DOS command to perform each of the assigned tasks.

   a. Format a bootable diskette with volume label BOOT-DISK.

   b. Display all file names for both user and hidden files on the diskette prepared in part (a).

   c. Use the output redirection operator to obtain a hard copy listing of the output from the command in part (b).

   d. Delete the volume label from the diskette.

   e. Show that the diskette has no volume label.

   f. Assign the volume label *namef*#4-5 to the diskette, where *name* denotes the first six characters of your last name or your entire last name (whichever is shorter) and *f* denotes the first character of your first name.

   g. Display the volume label on the screen.

6. The CHKDSK command can also be used to check the way in which files are stored on a disk. It allows you to determine whether a file is stored on a disk in a contiguous manner (contiguous or adjacent sectors) or stored in disjoint (separated) blocks (sectors). A file is called **fragmented** if it is stored on a disk in nonadjacent sectors. Disks that have a lot of file creation and deletion activity become fragmented, since disk space is not allocated sequentially. The first free sector that DOS finds is the next sector allocated to a file, regardless of its location on the disk. A computer user is interested in file fragmentation because a disk drive uses more time to read and write fragmented files than it does to access files stored on contiguous sectors. Assume that the DOS System diskette is in drive A and that drive A is the default drive.

   a. Use the command `CHKDSK *.*` to determine whether the DOS System diskette has any fragmented files.

   b. Use the output redirection operator to obtain a hard copy listing of the output from the command in part (a).

   c. Insert a diskette in drive B and use the command `CHKDSK B:*.*` to determine whether this diskette has any fragmented files.

7. Directory entries can be displayed in alphabetical order by **piping** (sending) the output from the DIR command to the SORT command as input. The **SORT** command is an external DOS command that sorts each line of output from the DIR command into alphabetical order. The symbol | is used to specify piping in the DOS command sequence. (*Note*: When the piping feature is utilized, DOS generates temporary files on the disk in the default drive to store the input and output data being piped. Consequently, the disk in the default drive cannot be write protected when you use piping in a DOS command sequence.) Insert the DOS System diskette (or a diskette supplied by your instructor) in drive A. Assume that drive A is the default drive.

   a. Use the DOS command sequence `DIR | SORT` to display the directory information for the DOS System diskette in alphabetical order. (The output information generated by the DIR command is piped as input to the SORT program, which is loaded from the DOS System diskette in drive A. The directory information is sorted and then displayed in alphabetical order on the screen.)

b. If the DOS System diskette is write protected, insert a usable diskette in drive B, set the default drive to drive B, and use the DOS command sequence `DIR A: ¦ A:SORT` to display the directory information for the DOS System diskette in alphabetical order.

c. Use the output redirection operator to obtain a hard copy listing of the output from the commands in parts (a) and (b).

5

Manipulating Files and File Characteristics

The material in this chapter introduces the DOS commands that allow you to:

- copy a specified file to the same disk or to another disk using the COPY command
- copy groups of files from one disk to another disk using the COPY command and the wildcard file name character *
- remove file names from the directory of a disk using the DEL or ERASE command
- change a file name using the RENAME command
- protect files from accidental deletion using the ATTRIB command

Copying One User File (COPY)

To make a copy of a single user file, you need to know the name of the file to be copied, the disk drive that contains the disk on which the file is stored, and the location of the file on that disk. This information is referred to as the **file specification**. Normally, a file specification consists of a filename and an optional filename extension. Sometimes a file specification includes a disk drive specifier and/or information about where to locate the file on a disk (path name). The **COPY** command is an internal DOS command that allows you to copy a specified file to the same disk using a different filename and/or extension, or to a second disk using the same filename and extension or a different filename and/or extension.

Since the COPY command is an internal DOS command, the DOS System diskette does not have to be available to use the COPY command. However, the DOS System diskette contains numerous user files. For this reason it will be used in the next two sections as a source of user files to help illustrate the use of the COPY command. Refer to Fig. 5.1 for a directory listing of the DOS System diskette.

Copying One User File to the Same Disk, Different Filename and/or Extension

Insert the DOS System diskette in drive A and boot the system. With the cursor positioned after the A > prompt, type:

```
        COPY A:FORMAT.COM A:FORMAT.BAK
```

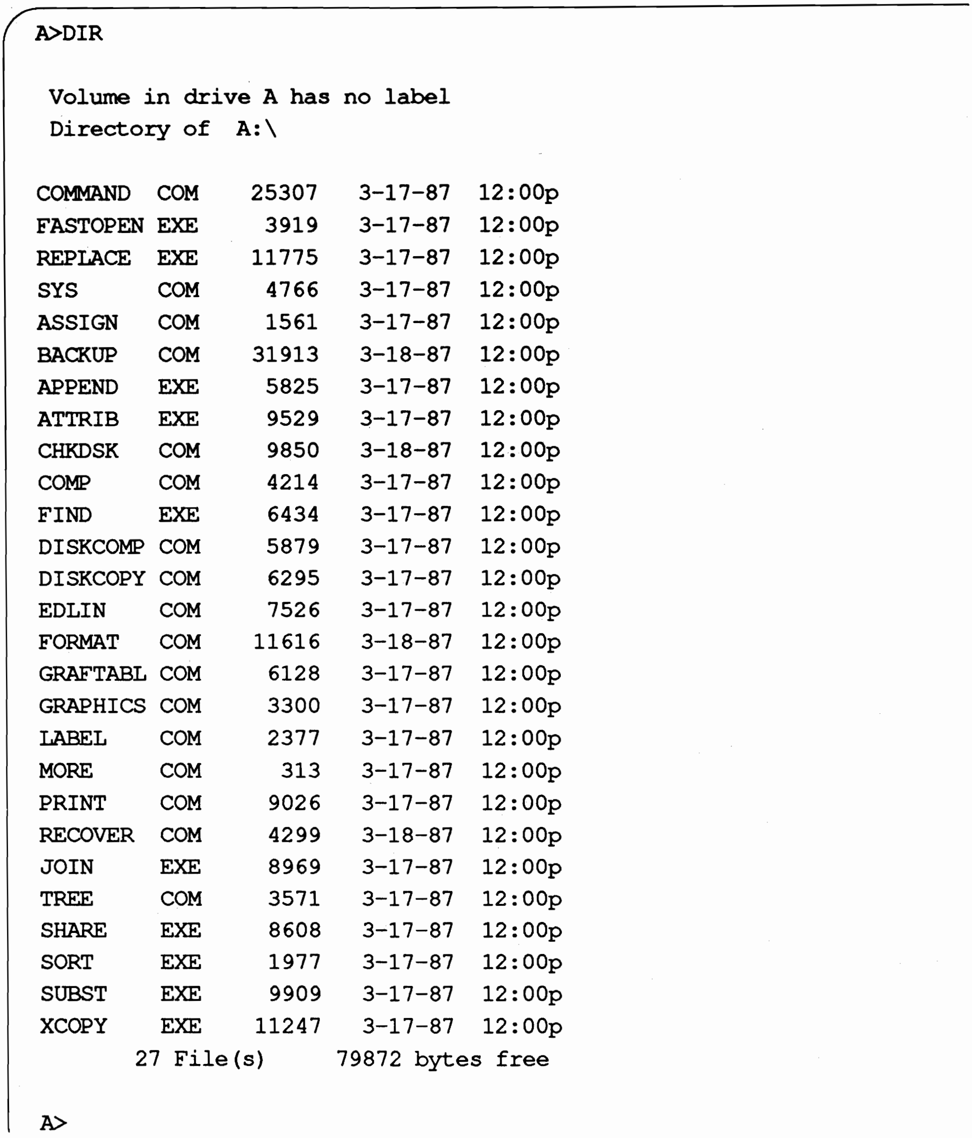
and press ⏎. This COPY command instructs DOS to copy a single file—in this case, FORMAT.COM—from the diskette in drive A to the diskette in drive A, giving it the same filename but a different extension. The first parameter in the COPY command identifies the file to be copied and is called the **source file specification**. The second parameter in the COPY command identifies the destination of the copied file and is called the **target file specification**. The term file specification is sometimes denoted by the abbreviation **filespec**. In this example, A:FORMAT.COM is called the source file specification, and A:FORMAT.BAK is called the target file specification (Fig. 5.2). When the COPY command finishes copying the specified file, a message similar to the following is displayed on the screen before the system returns to command level with the A > prompt:

```
        1 File(s) copied
```

The DOS System diskette in drive A now contains two copies of the FORMAT program, the original copy and the newly created backup copy FORMAT.BAK. The DIR command can be used to verify that there are two copies of the FORMAT program on the DOS System diskette in drive A. With the cursor positioned after the A > prompt, type

```
        DIR FORMAT.*
```

and press ⏎. Directory information similar to the following is displayed on



```
A>DIR

 Volume in drive A has no label
 Directory of  A:\

COMMAND  COM    25307   3-17-87  12:00p
FASTOPEN EXE     3919   3-17-87  12:00p
REPLACE  EXE    11775   3-17-87  12:00p
SYS      COM     4766   3-17-87  12:00p
ASSIGN   COM     1561   3-17-87  12:00p
BACKUP   COM    31913   3-18-87  12:00p
APPEND   EXE     5825   3-17-87  12:00p
ATTRIB   EXE     9529   3-17-87  12:00p
CHKDSK   COM     9850   3-18-87  12:00p
COMP     COM     4214   3-17-87  12:00p
FIND     EXE     6434   3-17-87  12:00p
DISKCOMP COM     5879   3-17-87  12:00p
DISKCOPY COM     6295   3-17-87  12:00p
EDLIN    COM     7526   3-17-87  12:00p
FORMAT   COM    11616   3-18-87  12:00p
GRAFTABL COM     6128   3-17-87  12:00p
GRAPHICS COM     3300   3-17-87  12:00p
LABEL    COM     2377   3-17-87  12:00p
MORE     COM      313   3-17-87  12:00p
PRINT    COM     9026   3-17-87  12:00p
RECOVER  COM     4299   3-18-87  12:00p
JOIN     EXE     8969   3-17-87  12:00p
TREE     COM     3571   3-17-87  12:00p
SHARE    EXE     8608   3-17-87  12:00p
SORT     EXE     1977   3-17-87  12:00p
SUBST    EXE     9909   3-17-87  12:00p
XCOPY    EXE    11247   3-17-87  12:00p
       27 File(s)     79872 bytes free

A>
```

Figure 5.1 When the DIR command is used to display the directory contents of a 360KB DOS System diskette for Version 3.30, information similar to the above is displayed on the screen. The DOS System diskette is used repeatedly in this chapter as a source of user files to help illustrate the use of DOS commands introduced in this chapter.

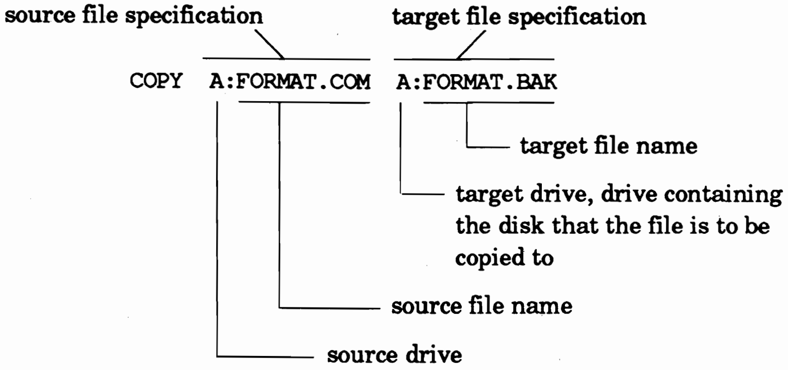

source file specification
target file specification
COPY A:FORMAT.COM A:FORMAT.BAK
target file name
target drive, drive containing the disk that the file is to be copied to
source file name
source drive

Figure 5.2 The above example illustrates the parts of the source and target file specifications in a COPY command.

the screen:

```
         Volume in drive A has no label
         Directory of  A:\

        FORMAT   COM     11616    3-18-87  12:00p
        FORMAT   BAK     11616    3-18-87  12:00p
                2 File(s)      67584 bytes free
```

With the DOS System diskette in drive A and with the cursor positioned after the A > prompt, type

```
        COPY FORMAT.COM FORMT.*
```

and press ⏎. This COPY command instructs DOS to copy a single file—in this case, FORMAT.COM—from the diskette in the default drive (drive A) to the diskette in the default drive (drive A), giving the copied file a different filename (FORMT) but the same extension (.COM). This example illustrates two things. First, when the default drive is the disk drive for the source file specification (or the target file specification), the drive letter does not have to be specified in that COPY command parameter. Consequently, the command COPY FORMAT.COM FORMAT.BAK could have been typed in place of the first COPY command in this section. Second, when the wildcard file name character * is used in place of a filename (or an extension) in the target file specification of a COPY command, the corresponding filename (or file extension) in the source file specification is substituted for the * character. In the above example the wildcard character * was used to declare the same extension in the target file name that was specified in the source file name—in this case, .COM. This also means that the command COPY FORMAT.COM *.BAK could have been typed in place of the first COPY command in this section.

With the cursor positioned after the A > prompt, type

```
DIR FOR*
```

and press ↵. Directory information similar to the following is displayed on the screen:

```
 Volume in drive A has no label
 Directory of  A:\

FORMAT   COM    11616   3-18-87  12:00p
FORMAT   BAK    11616   3-18-87  12:00p
FORMT    COM    11616   3-18-87  12:00p
      3 File(s)     55296 bytes free
```

This directory information indicates that the DOS System diskette now contains three copies of the FORMAT program. The three files have names FORMAT.COM, FORMAT.BAK, and FORMT.COM. Do not worry about the two extra files that you just put on your DOS System diskette. You will learn how to remove these files from the directory of the DOS System diskette before the end of this chapter.

Keystroke Summary: Copying One User File to the Same Disk

Insert:	diskette in drive A that contains a file to be copied
Set:	default drive to drive A
Type:	`COPY` *`source-file-name target-file-name`*
Press:	↵

Copying One User File to a Second Disk

The COPY command is normally used to copy named files from a disk in one drive (source disk) to a disk in a second drive (target disk) using either the same or different filenames and extensions. The following examples illustrate the use of the COPY command to copy a single specified file from a disk in one drive to a disk in another drive.

Insert the DOS System diskette in drive A and a formatted diskette with an empty directory in drive B. With the cursor positioned after the A > prompt, type

```
COPY FORMAT.COM B:FORMAT.COM
```

and press ↵. This COPY command instructs DOS to copy the specified file—in this case, FORMAT.COM—from the diskette in the default (source) drive (drive A) to the diskette in the specified (target) drive (drive B) using the same file name. Each of the following commands produces the same result as the above command:

```
COPY FORMAT.COM B:FORMAT.*
COPY FORMAT.COM B:*.COM
COPY FORMAT.COM B:*.*
COPY FORMAT.COM B:
```

The last COPY command illustrates that when you want the target filename and extension to be the same as the source filename and extension, you need to specify only the target drive in the COPY command.

With the cursor positioned after the A > prompt, type `B:` and press ↵ to change the default drive to drive B. With the cursor positioned after the B> prompt, type

```
COPY A:CHKDSK.COM
```

and press ↵. Since drive B is the default drive, this COPY command copies the specified file—in this case, CHKDSK.COM—from the diskette in the specified drive (drive A) to the diskette in the default drive (drive B). When no target file specification is given in the COPY command, the default drive—in this case, drive B—is assumed, and the source filename and extension—in this case, CHKDSK.COM—are used for the target file.

With the cursor positioned after the B> prompt, type

```
DIR
```

and press ↵. Directory information similar to the following is displayed on the screen:

```
 Volume in drive B has no label
 Directory of  B:\

FORMAT   COM    11616   3-18-87  12:00p
CHKDSK   COM     9850   3-18-87  12:00p
        2 File(s)    339968 bytes free
```

This directory information indicates that the diskette in drive B contains the two files copied to it by the above COPY commands.

With the cursor positioned after the B> prompt, type

```
COPY A:FORMAT.COM FRMT.BAK
```

and press ↵. Since drive B is the default drive, this COPY command copies the specified file—in this case, FORMAT.COM—from the diskette in the specified drive (drive A) to the diskette in the default drive (drive B) using a different filename and extension—in this case, FRMT.BAK.

With the cursor positioned after the B> prompt, type

```
DIR
```

and press ↵. Directory information similar to the following is displayed on the screen:

```
  Volume in drive B has no label
  Directory of  B:\

 FORMAT   COM     11616   3-18-87  12:00p
 CHKDSK   COM      9850   3-18-87  12:00p
 FRMT     BAK     11616   3-18-87  12:00p
         3 File(s)    327680 bytes free
```

This directory information indicates that the diskette in drive B now contains three files.

Keystroke Summary: Copying One User File to a Second Disk

Insert: source diskette in drive A

Insert: target diskette in drive B

Set: default drive to drive A

Type: COPY *source-file-name* B: or

COPY *source-file-name* B:*target-file-name*

Press: ↵

What Can Go Wrong?

1. After you enter a COPY command, information similar to the following is displayed on the screen:

```
        File cannot be copied onto itself
                0 File(s) copied
```

Cause: This message means that the source and target file specifications in a COPY command cannot specify the same file. For example, a COPY command like COPY FORMAT.COM attempts to copy the specified file—in this case, FORMAT.COM—from the default drive to the default drive using the same file name. Such a command causes the above message to be displayed on the screen.

Solution: There are two possible solutions to this problem. If you want to copy a file onto the same disk, you must specify a target file name that is different from the source file name. The COPY command, COPY FORMAT.COM FRMT.COM, copies the source file (FORMAT.COM) to the same diskette using a different file name (FRMT.COM). If you want to copy a file to a different disk using the same file name, then the source file drive and the target file drive must be different. If the default drive is drive A, then the COPY command, COPY FORMAT.COM B:, copies the source file (FORMAT.COM) from the default drive (drive A) to the specified drive (drive B) using the same file name.

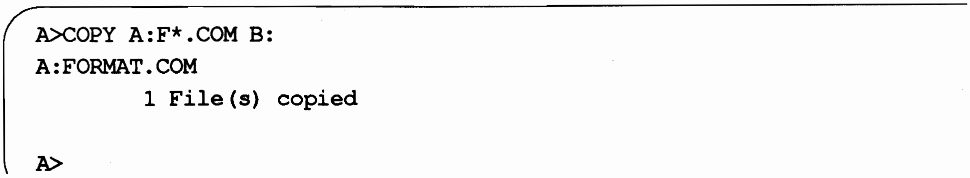
Copying Selected User Files (COPY)

Recall from Chapter 4 that the wildcard file name characters * and ? are used to identify groups of files having similar filenames and/or extensions. The following examples illustrate the use of the wildcard character * with the COPY command to copy groups of files having similar file names from a disk in one drive to a disk in a second drive. Again note that the DOS System diskette will be used in this section as a source of user files to help illustrate the use of the COPY command. See Fig. 5.1 for a directory listing of the DOS System diskette.

Insert the DOS System diskette in drive A and a formatted diskette with an empty directory in drive B. Set the default drive to drive A. With the cursor positioned after the A > prompt, type

```
COPY A:F*.COM B:
```

and press ↵. This COPY command instructs DOS to copy from the diskette in the specified drive (drive A) each file having the same first character in the filename—in this case, F—and the same extension—in this case, .COM—to the diskette in the specified drive (drive B) using the same filename and extension. Also note that DOS displays the specification of each file being copied when a wildcard character is used in the source file specification of the COPY command (Fig. 5.3).



```
A>COPY A:F*.COM B:
A:FORMAT.COM
        1 File(s) copied


A>
```

Figure 5.3 When a wildcard file name character is used in the source file specification of the COPY command, the specification of each file being copied is displayed on the screen.

With the cursor positioned after the A > prompt, type

```
DIR B:
```

and press ↵. Directory information similar to the following is displayed on the screen:

```
  Volume in drive B has no label
  Directory of  B:\

FORMAT   COM    11616   3-18-87  12:00p
        1 File(s)     350208 bytes free
```

This directory information indicates that the diskette in drive B contains the file copied to it by the above COPY command.

With the cursor positioned after the A > prompt, type

```
COPY F*.* B:
```

and press ↵. This COPY command copies from the diskette in the default drive (drive A) each file having the same first character in the filename—in this case, F—and any extension to the diskette in the specified drive (drive B) using the same filename and extension. Again, note that DOS displays the specification of each file being copied by the above COPY command (Fig. 5.4).

With the cursor positioned after the A > prompt, type

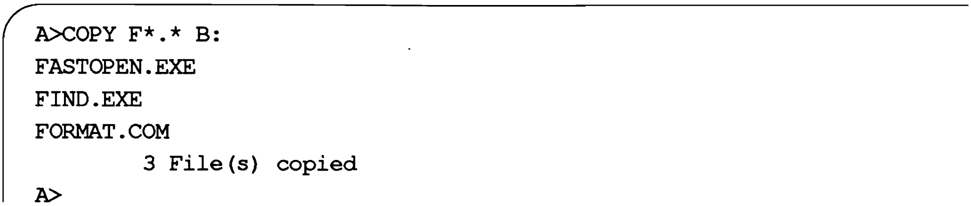
```
DIR B:
```

and press ↵. Directory information similar to the following is displayed on the screen:

```
  Volume in drive B has no label
  Directory of  B:\

FORMAT   COM    11616   3-18-87  12:00p
FASTOPEN EXE     3919   3-17-87  12:00p
FIND     EXE     6434   3-17-87  12:00p
        3 File(s)     338944 bytes free
```

This directory information indicates that the diskette in drive B now contains three files. Note that the information in Fig. 5.4 shows that the command, COPY F*.* B:, copied three files from the DOS System diskette in drive A to the diskette in drive B. The above COPY command copied the file FORMAT.COM from the diskette in drive A over the file with the same name in drive B; it also copied two new files, FASTOPEN.EXE and FIND.EXE, to the diskette in drive B. Even though DOS will not copy a file over itself on the same disk, this example illustrates that DOS will copy a file from one disk over a file on another disk having exactly the same file name. For this reason you should use the DIR command to carefully check file dates before copying



```
A>COPY F*.* B:
FASTOPEN.EXE
FIND.EXE
FORMAT.COM
        3 File(s) copied
A>
```

Figure 5.4 When the wildcard file name characters are used in the source file specification of the COPY command, the specification of each file being copied is displayed on the screen. Since the default drive (drive A) is being assumed in the source file specification of the COPY command, it is not displayed with each file name as it was in Figure 5.3.

files between disks. This procedure may help you avoid copying an old file over a new file with the same file name.

With the cursor positioned after the A > prompt, type `B:` and press ↵ to change the default drive to drive B. With the cursor positioned after the B> prompt, type

```
COPY A:F*.*
```

and press ↵. This COPY command copies from the diskette in the specified drive (drive A) each file having the same first character in the filename—in this case, F—and any extension to the diskette in the default drive (drive B) using the same filename and extension. In other words, this COPY command duplicates the copying process of the previous COPY command. The only difference is that the default drive for this command is drive B, while the default drive for the previous COPY command was drive A. This example illustrates that when the target file specification parameter in a COPY command is omitted, the default drive is assumed to be the target drive, and the target file names are assumed to be the same as the corresponding source file names.

With the cursor positioned after the B> prompt, type

```
COPY A:*.COM
```

and press ↵. This COPY command copies from the diskette in the specified drive (drive A) each file having the same extension—in this case, .COM—to the default drive (drive B) using the same filename and extension.

With the cursor positioned after the B> prompt, type

```
DIR *.COM
```

and press ↵. This DIR command displays from the diskette's directory in the default drive (drive B) all file names having the same extension—in this case, .COM. Each displayed file was copied to the diskette in drive B by the above

COPY command. Directory information similar to the following is displayed on the screen:

```
      Volume in drive B has no label
      Directory of  B:\

     FORMAT    COM     11616   3-18-87  12:00p
     COMMAND   COM     25307   3-17-87  12:00p
     SYS       COM      4766   3-17-87  12:00p
     ASSIGN    COM      1561   3-17-87  12:00p
     BACKUP    COM     31913   3-18-87  12:00p
     CHKDSK    COM      9850   3-18-87  12:00p
     COMP      COM      4214   3-17-87  12:00p
     DISKCOMP  COM      5879   3-17-87  12:00p
     DISKCOPY  COM      6295   3-17-87  12:00p
     EDLIN     COM      7526   3-17-87  12:00p
     GRAFTABL  COM      6128   3-17-87  12:00p
     GRAPHICS  COM      3300   3-17-87  12:00p
     LABEL     COM      2377   3-17-87  12:00p
     MORE      COM       313   3-17-87  12:00p
     PRINT     COM      9026   3-17-87  12:00p
     RECOVER   COM      4299   3-18-87  12:00p
     TREE      COM      3571   3-17-87  12:00p
            17 File(s)    203776 bytes free
```

With the cursor positioned after the B> prompt, type

```
COPY A:MO*.COM *.BAK /V
```

and press ↵. This COPY command copies from the diskette in the specified drive (drive A) each file having the same first two characters in the filename—in this case, MO—and the same extension—in this case, .COM—to the diskette in the default drive (drive B) using the same filename but the new specified extension—in this case, .BAK. The **/ V** (verify) parameter at the end of this COPY command instructs DOS to verify that the source and target files are identical after copying. When you want to be certain that the copied file(s) is identical to the source file(s), you should use the / V parameter at the end of the COPY command. The disadvantage of using the / V parameter is that file verification slows down the file copying process.

Note that a space does not have to be used before the slash (/) in the above COPY command. Since a slash is interpreted by DOS as a parameter field delimiter (separator), the space before the / V parameter is used only to help make the command more readable.

With the cursor positioned after the B> prompt, type `A:` and press ↵ to change the default drive back to drive A. With the cursor positioned after the A > prompt, type

```
COPY *.* B:
```

and press ⏎. This COPY command copies each user file from the diskette in the default drive (drive A) to the diskette in the specified drive (drive B) using the same file name. Note that this COPY command copied all user files from the DOS System diskette in drive A to the diskette in drive B. However, the two hidden system files IBMBIO.COM and IBMDOS.COM are not copied by this COPY command. The above COPY command is a good command to use when you want to make duplicate copies of all user files on a diskette. The topic of backing up disks will be covered in Chapter 6.

Keystroke Summary: Copying Selected User Files

Insert:	source diskette in drive A
Insert:	target diskette in drive B
Set:	default drive to drive A
Type:	`COPY` *`source-filespec`* `B:` or
	`COPY` *`source-filespec`* `B:`*`target-filespec`*
Press:	⏎

Removing (Deleting, Erasing) File Names from a Disk's Directory (DEL, ERASE)

Now that you know how to make copies of files, you need to know how to remove unwanted file names (specifications) from a disk's directory. The **DEL** (DELete) and **ERASE** commands are internal DOS commands that allow you to remove specified files from the directory of a disk. Only the DEL command will be used in this section, since both the DEL and the ERASE commands perform exactly the same operations.

Since the DEL command is an internal DOS command, the DOS System diskette does not have to be available to use the DEL command. However, the DOS System diskette contains numerous user files. For this reason a scratch copy of the DOS System diskette will be used in some of the following sections as a source of user files to help illustrate the use of the DEL command. See Fig. 5.1 for a directory listing of the DOS System diskette.

Removing One User File Name from a Disk's Directory

The DEL command can be used to delete a single file name from a disk's directory by specifying the file name in the DEL command. Recall from the first part of this chapter that you created two new files on the DOS System diskette. Insert the DOS System diskette in drive A. Set the default drive to drive A. With the cursor positioned after the A > prompt, type

```
DIR FOR*
```

and press ↵. Directory information similar to the following is displayed on the screen:

```
         Volume in drive A has no label
         Directory of  A:\

        FORMAT   COM     11616   3-18-87  12:00p
        FORMAT   BAK     11616   3-18-87  12:00p
        FORMT    COM     11616   3-18-87  12:00p
                3 File(s)     55296 bytes free
```

The two extra files have the names FORMAT.BAK and FORMT.COM. You will now learn how to remove these two file names from the diskette's directory.

With the DOS System diskette in drive A and the cursor positioned after the A> prompt, type

```
DEL A:FORMAT.BAK
```

and press ↵. This DEL command instructs DOS to remove a single user file name—in this case, FORMAT.BAK—from the directory of the diskette in the specified drive—in this case, the DOS System diskette in drive A. After the DEL command is finished, control returns to command level with the A> prompt.

With the cursor positioned after the A > prompt, type

```
DEL FORMT.COM
```

and press ↵. This DEL command instructs DOS to remove a single user file name—in this case, FORMT.COM—from the directory of the diskette in the default drive—in this case, the DOS System diskette in drive A. With the cursor positioned after the A > prompt, type

```
DIR FOR*
```

and press ↵. Directory information similar to the following is displayed on the screen:

```
         Volume in drive A has no label
         Directory of  A:\

        FORMAT   COM     11616   3-18-88  12:00p
                1 File(s)     79872 bytes free
```

Notice that the two extra file names have been deleted from the directory of the DOS System diskette. This diskette is now back to its original form.

Keystroke Summary: Removing One User File Name from a Disk's Directory

Insert: diskette in drive A that contains a file to be deleted from the disk's directory

Set: default drive to drive A

Type: `DEL` *`file-name`*

Press: ↵

Removing Selected User File Names from a Disk's Directory

The examples in this section illustrate the use of the wildcard file name character * with the DEL command to delete groups of files having similar file names from the directory of a disk. A scratch copy of the DOS System diskette will be used in this section as a source of user files to help illustrate the use of the DEL command. To prepare a scratch copy of the DOS System diskette, you need to perform the following three steps. First, insert the DOS System diskette in drive A and a scratch diskette in drive B, and set the default drive to drive A. Second, use the command FORMAT B: /S to prepare a bootable diskette from the scratch diskette in drive B. (Refer to the FORMAT command in Chapter 3.) Third, use the command COPY A:*.* B: to copy all user files from the DOS System diskette to the scratch diskette. You have now prepared a scratch copy of the DOS System diskette.

Before you proceed with the remainder of this section, you should open the latch on drive A. This will prevent you from accidentally deleting files from the DOS System diskette in drive A.

You will note that in each of the following examples, the DIR command is used with the file specification from the DEL command before the DEL command is entered and executed. This practice should be followed regularly when you are using wildcard file name characters in the DEL command. It gives you an opportunity to review the file names to be deleted before they are actually deleted from a disk's directory. The second DIR command is used after the DEL command to verify that the file names were deleted from the disk's directory.

With the cursor positioned after the A > prompt, type

```
DIR B:F*.COM
```

and press ↵. Directory information similar to the following is displayed on the screen:

```
 Volume in drive B has no label
 Directory of  B:\

FORMAT   COM    11616   3-18-87  12:00p
        1 File(s)     79872 bytes free
```

This information indicates that the directory of the diskette in drive B contains one file name that has F as the first character in the filename and .COM as an extension. It also states that there are 79,872 bytes free on the diskette.

With the cursor positioned after the A > prompt, type

```
DEL B:F*.COM
```

and press ↵. This DEL command instructs DOS to delete from the diskette's directory in the specified drive (drive B) each file having the same first character in the filename—in this case, F—and the same extension—in this case, .COM.

With the cursor positioned after the A > prompt, type

```
DIR B:F*.COM
```

and press ↵. Directory information similar to the following is displayed on the screen:

```
 Volume in drive B has no label
 Directory of  B:\

File not found
```

This information indicates that the directory of the diskette in drive B does not contain any file names that satisfy the specification F*.COM. In other words, the command DEL B:F*.COM deleted the file name from the directory of the diskette in drive B that originally satisfied this specification.

With the cursor positioned after the A > prompt, type

```
DIR B:S*.*
```

and press ↵. Directory information similar to the following is displayed on the screen:

```
 Volume in drive B has no label
 Directory of  B:\

SYS      COM      4766   3-17-87  12:00p
SHARE    EXE      8608   3-17-87  12:00p
SORT     EXE      1977   3-17-87  12:00p
SUBST    EXE      9909   3-17-87  12:00p
        4 File(s)      92160 bytes free
```

This information indicates that the directory of the diskette in drive B contains four file names that have S as the first character in the filename and any extension. It also states that there are now 92,160 bytes free on the diskette.

With the cursor positioned after the A > prompt, type

```
DEL B:S*.*
```

and press ↵. This DEL command instructs DOS to delete from the diskette's directory in the specified drive (drive B) each file name having the same first character in the filename—in this case, S—and any extension.

With the cursor positioned after the A> prompt, type

```
DIR B:S*.*
```

and press ⏎. Directory information similar to the following is displayed on the screen:

```
 Volume in drive B: has no label
 Directory of  B:\

File not found
```

This information indicates that the directory of the diskette in drive B does not contain any file names that satisfy the S*.* specification. In other words, the command DEL B:S*.* deleted the four file names from the directory of the diskette in drive B that originally satisfied this specification.

With the cursor positioned after the A > prompt, type `B:` and press ⏎ to change the default drive to drive B. With the cursor positioned after the B> prompt, type

```
DIR *.EXE
```

and press ⏎. Directory information similar to the following is displayed on the screen:

```
 Volume in drive B has no label
 Directory of  B:\

FASTOPEN EXE     3919   3-17-87  12:00p
REPLACE  EXE    11775   3-17-87  12:00p
APPEND   EXE     5825   3-17-87  12:00p
ATTRIB   EXE     9529   3-17-87  12:00p
FIND     EXE     6434   3-17-87  12:00p
JOIN     EXE     8969   3-17-87  12:00p
XCOPY    EXE    11247   3-17-87  12:00p
        7 File(s)    118784 bytes free
```

This information indicates that the directory of the diskette in drive B contains seven file names that have .EXE as an extension. It also states that there are now 118,784 bytes free on the diskette. Notice that each time a file name is deleted from the directory of the disk, the disk space occupied by the named file is freed for future use by DOS.

With the cursor positioned after the B> prompt, type

```
DEL *.EXE
```

and press ⏎. This DEL command instructs DOS to delete from the diskette's directory in the default drive (drive B) each file name having the same specified extension—in this case, .EXE.

With the cursor positioned after the B> prompt, type

```
DIR *.EXE
```

and press ↵. Directory information similar to the following is displayed on the screen:

```
 Volume in drive B has no label
 Directory of  B:\

File not found
```

This information indicates that the directory of the diskette in drive B does not contain any file names that satisfy the *.EXE specification. In other words, the command DEL *.EXE deleted the seven file names from the directory of the diskette in drive B that originally satisfied this specification.

With the cursor positioned after the B> prompt, type

```
DIR *.*
```

and press ↵. Directory information similar to the following is displayed on the screen:

```
 Volume in drive B has no label
 Directory of  B:\

COMMAND  COM    25307   3-17-87  12:00p
ASSIGN   COM     1561   3-17-87  12:00p
BACKUP   COM    31913   3-18-87  12:00p
CHKDSK   COM     9850   3-18-87  12:00p
COMP     COM     4214   3-17-87  12:00p
DISKCOMP COM     5879   3-17-87  12:00p
DISKCOPY COM     6295   3-17-87  12:00p
EDLIN    COM     7526   3-17-87  12:00p
GRAFTABL COM     6128   3-17-87  12:00p
GRAPHICS COM     3300   3-17-87  12:00p
LABEL    COM     2377   3-17-87  12:00p
MORE     COM      313   3-17-87  12:00p
PRINT    COM     9026   3-17-87  12:00p
RECOVER  COM     4299   3-18-87  12:00p
TREE     COM     3571   3-17-87  12:00p
       15 File(s)    179200 bytes free
```

The specification *.* specifies each file name that has any filename and any extension; that is, every user file name in the directory of the disk. The command DIR *.* is equivalent to the command DIR. Consequently, the above information indicates that the directory of the diskette in drive B contains fifteen file names. It also states that there are 179,200 bytes free on the diskette.

With the cursor positioned after the B> prompt, type

```
DEL *.*
```

and press ↵. This DEL command instructs DOS to delete every user file name from the diskette's directory in the default drive (drive B). Before the DEL command carries out its assigned task, it gives you a chance to reconsider the execution of this command by displaying the following message on the screen:

```
Are you sure (Y/N)?
```

With the cursor positioned after the question mark (?), type `Y` and press ↵. This response deletes all user file names from the diskette's directory in drive B. If you had typed `N` and pressed ↵, execution of the above DEL command would have terminated, no file names would have been deleted from the directory of the diskette, and control would have returned to command level.

With the cursor positioned after the B> prompt, type

```
DIR
```

and press ↵. Directory information similar to the following is displayed on the screen:

```
 Volume in drive B has no label
 Directory of  B:\

File not found
```

This information indicates that the directory of the diskette in drive B contains no user file names. You should be careful to properly interpret the above information. It does not state that the directory of the diskette is empty. It only states that there are no user file names in the diskette's directory in drive B. Recall that the DEL command is used to delete user file names from the directory of a disk.

The CHKDSK command (see Chapter 4) will be used to determine the status of the diskette in drive B. Insert the DOS System diskette in drive A. With the cursor positioned after the B> prompt, type

```
A:CHKDSK B: /V
```

and press ↵. Since CHKDSK is an external DOS command, it must be loaded into memory from the DOS System diskette. The specification A:CHKDSK instructs DOS to locate the CHKDSK command on the disk in the specified drive (drive A) and load it into memory. Once the loading process is complete, DOS initiates the execution of the CHKDSK program which analyzes the disk in the specified drive (drive B). The / V (view) parameter is used to display a list of the file names for both user and hidden files on the disk. The above CHKDSK command displays information similar to the following on the screen:

```
        Directory B:\
              B:\IBMBIO.COM
              B:\IBMDOS.COM

           362496 bytes total disk space
            53248 bytes in 2 hidden files
           309248 bytes available on disk

           655360 bytes total memory
           600160 bytes free
```

This information indicates that the diskette in drive B contains no user files but does contain two hidden files. These files are DOS's two hidden files, IBMBIO.COM and IBMDOS.COM. Since the directory of the diskette in drive B contains the names for these two hidden files, it is not an empty directory.

Keystroke Summary: Removing Selected User File Names from a Disk's Directory

Insert: diskette in drive A that contains selected files to be deleted from the disk's directory

Set: default drive to drive A

Type: `DEL` *filespec*

Press: ↵

What Can Go Wrong?

The DOS System diskette will be used here as a source of user files to help illustrate different interpretations of a file specification when used with different DOS commands.

1. When you enter the command DIR D*, information similar to the following is displayed on the screen:

```
         Volume in drive A has no label
         Directory of  A:\

        DISKCOMP COM     5879   3-17-87   12:00p
        DISKCOPY COM     6295   3-17-87   12:00p
                2 File(s)     30720 bytes free
```

When you enter the command DEL D*, the following message is displayed on the screen:

```
File not found
```

Cause: The file specification D* is interpreted differently in the DIR command than it is in other DOS commands such as the DEL, COPY, and CHKDSK commands.

Solution: When the file specification D* is used with the DIR command, it specifies each file name in the disk's directory that has D as the first character in the filename and any extension. When the file specification D* is used in the DEL command, it specifies each file name in the disk's directory that has D as the first character in the filename and no extension. If you want to delete all file names from the directory of a disk that have D as the first character in the filename and any extension, you should use the specification D*.* in the DEL command. If you want to display all file names in the directory of a disk that have D as the first character in the filename and no extension, you should use the specification D*. in the DIR command. Consequently, D*. in a DIR command acts like D* and D*. in a DEL command. Note that when a file specification such as D* is used in COPY and CHKDSK commands, its interpretation is the same for these commands as it is for the DEL command.

Changing File Names (RENAME)

In this section you will learn how to change the name of a file. Quite often you create a file with a particular filename and extension and later decide that you want to change the filename and/or extension. The **RENAME** command is an internal DOS command that allows you to change a file name or a selected group of file names within the same disk directory.

Since the RENAME command is an internal DOS command, the DOS System diskette does not have to be available to use the RENAME command. However, the DOS System diskette contains numerous user files. For this reason a scratch copy of the DOS System diskette will be used in the next section as a source of user files to help illustrate the use of the RENAME command. For the steps used to prepare a scratch copy of the DOS System diskette you should refer to the section on removing selected user file names from a disk's directory that appears earlier in this chapter.

Before you proceed with the remainder of this section, set the default drive to drive A and then open the latch on drive A. This will prevent you from accidentally renaming files on the DOS System diskette in drive A.

In the following examples the DIR command is used to display the results of the RENAME command. Insert the scratch copy of the DOS System diskette in drive B. With the cursor positioned after the A > prompt, type

```
DIR B:F*
```

and press ↵. Directory information similar to the following is displayed on the screen:

```
 Volume in drive B has no label
 Directory of  B:\

FASTOPEN EXE     3919   3-17-87  12:00p
FIND     EXE     6434   3-17-87  12:00p
FORMAT   COM    11616   3-18-87  12:00p
        3 File(s)     79872 bytes free
```

With the cursor positioned after the A > prompt, type

```
RENAME B:FORMAT.COM FRMT.COM
```

and press ↵. Two file specifications appear in the RENAME command. The first specification identifies the file name that is going to be changed—in this case, FORMAT.COM in the directory of the diskette in drive B. The second parameter specifies the new name for the file—in this case, FRMT.COM. Thus, the above RENAME command changes the file name FORMAT.COM in the diskette's directory of the specified drive (drive B) to FRMT.COM before returning to command level with the A> prompt. Notice that the disk drive specifier is not included as part of the new file specification (second parameter).

With the cursor positioned after the A > prompt, type

```
DIR B:F*
```

and press ↵. Directory information similar to the following is displayed on the screen:

```
 Volume in drive B has no label
 Directory of  B:\

FASTOPEN EXE     3919   3-17-87  12:00p
FIND     EXE     6434   3-17-87  12:00p
FRMT     COM    11616   3-18-87  12:00p
        3 File(s)     79872 bytes free
```

This information indicates that the file name FORMAT.COM has been changed to FRMT.COM in the directory of the diskette in drive B.

With the cursor positioned after the A > prompt, type `B:` and press ↵ to change the default drive to drive B. With the cursor positioned after the B> prompt, type

```
RENAME FIND.EXE FIND.OLD
```

and press ↵. This RENAME command changes the file name FIND.EXE in the diskette's directory of the default drive (drive B) to FIND.OLD.

With the cursor positioned after the B> prompt, type

```
RENAME FASTOPEN.EXE FSTOPN.BAK
```

and press ↵. This RENAME command changes the file name FASTOPEN.EXE in the diskette's directory of the default drive (drive B) to FSTOPN.BAK.

With the cursor positioned after the B> prompt, type

```
DIR F*
```

and press ↵. Directory information similar to the following is displayed on the screen:

```
  Volume in drive B has no label
  Directory of  B:\

FSTOPN   BAK      3919   3-17-87  12:00p
FIND     OLD      6434   3-17-87  12:00p
FRMT     COM     11616   3-18-87  12:00p
        3 File(s)     79872 bytes free
```

This information indicates that the names of the three files have been changed in the directory of the diskette in drive B. In fact, the above information verifies that the RENAME command can be used to change the filename part of a file name, the extension part of a file name, or both the filename and extension parts of a file name.

Keystroke Summary: Changing the Name of a File

Insert:	diskette in drive A that contains a file name to be changed
Set:	default drive to drive A
Type:	RENAME *current-file-name new-file-name*
Press:	↵

What Can Go Wrong?

1. After you enter a RENAME command, a message similar to the following is displayed on the screen:

```
Invalid Parameter
```

Cause: This message refers to an error in the second parameter of the RENAME command. For example, the command RENAME A:FORMAT.COM A:FRMT.COM generates the above message. The second parameter of the RENAME command can specify only a file name; it cannot include a disk drive specification.

Solution: Check the second parameter of the RENAME command. If the second parameter contains a disk drive specification, the drive specification must be removed from this parameter. For example, the drive specification A: in the second parameter (A:FRMT.COM) of the above RENAME command must be removed. Note that if the disk in the default drive contains the file name to be changed, no disk drive specification is necessary in the first parameter. If the file name to be changed is on a disk in a drive different from the default drive, then the first parameter of the RENAME command must contain the appropriate drive specification.

Protecting Files from Accidental Deletion

At one time or another, everyone who uses a personal computer accidentally deletes a file from a disk. If this happens to you and you have a backup copy of the file, you use the COPY command to copy the file back onto your disk. If you have no backup copy of the deleted file, you will experience a feeling of shock and helplessness. If (or when) this happens to you, the following comments might prove very helpful to you in dealing with the situation.

First, you should realize that when you use the DEL or ERASE command to remove a file name from the directory of a disk, it does not destroy the information in the file. The DEL and ERASE commands only mark the file name in the disk's directory as a deleted file. This means that a file name removed from a disk's directory with the DEL or ERASE command can sometimes be recovered through the use of special utility programs.

When a file name is removed from a disk's directory with the DEL or ERASE command, the directory position for that file name can be used by DOS to store a name of a new file that is added to the disk. The information in the deleted file is actually destroyed when the information from the new file is written on the disk. Consequently, if you accidentally delete a file from a diskette with the DEL command, immediately remove the diskette from the diskette drive so that no other modifications can be performed on the diskette. If you do this, you can purchase special utility programs that will allow you to recover the deleted file.

There are several things that you can do to help prevent the accidental deletion of file names from the directory of a disk.

1. You should be very careful when you use the DEL or ERASE command. Double-check the file names specified in the DEL command before you press the ↵ key.
2. Write protect a diskette when you want to protect the entire diskette. To write protect a 5.25 inch diskette, a write-protect tab should be placed over the write-protect notch on the edge of the diskette jacket (Fig. 3.1). To

write protect a 3.50 inch diskette, the plastic tab in the write-protect window should be moved so that the window is open (Fig. 3.2). When a diskette is write protected, no file can be deleted from the diskette. In fact, no information can be modified on a diskette that is write protected.

3. Before you use a wildcard file name character in a DEL command, you should use the same wildcard specification in a DIR command to verify that the files you intend to delete are the files that are included in the wildcard specification.
4. The **ATTRIB** (ATTRIBute) command is an external DOS command that allows you to selectively declare files as read-only files. Files that are declared as read-only files can be read but cannot be modified or deleted by using the DEL or ERASE command.

Changing the Read-Only File Access Attribute (ATTRIB)

Earlier in this chapter, you learned how to remove file names from the directory of a disk. Accidental deletion of file names from the directory of a disk becomes a major problem. The ATTRIB command is an external DOS command that can be used to help prevent files from being accidentally deleted from a disk.

Insert the DOS System diskette in drive A and a formatted diskette in drive B. With the cursor positioned after the A > prompt, type

```
COPY CHKDSK.COM B:
```

and press ⏎ to copy the file CHKDSK.COM from the DOS System diskette in drive A to the diskette in drive B. With the cursor positioned after the A > prompt, type

```
ATTRIB +R B:CHKDSK.COM
```

and press ⏎. After the ATTRIB program is loaded into memory from the DOS System diskette in drive A, the **+R** parameter instructs the ATTRIB command to declare the specified file(s)—in this case, CHKDSK.COM—on the diskette in the specified drive (drive B) as a read-only file. When the ATTRIB program is finished, control returns to command level with the A > prompt.

With the cursor positioned after the A > prompt, type

```
ATTRIB B:CHKDSK.COM
```

and press ⏎. After the ATTRIB program is loaded into memory from the DOS System diskette in drive A, it displays the following attribute status of a single file—in this case, CHKDSK.COM—on the diskette in the specified drive (drive B):

```
A    R  B:\CHKDSK.COM
```

The R indicates that the file CHKDSK.COM on the diskette in drive B is a read-only file. The initial A indicates that the archive bit for CHKDSK.COM is turned on. For additional information about the archive bit of a file, you should refer to the ATTRIB command in Appendix B.

Since CHKDSK.COM has been declared as a read-only file, it cannot be modified or deleted. With the cursor positioned after the A > prompt, type

```
DEL B:CHKDSK.COM
```

and press ↵. Since the file specification B:CHKDSK.COM in the DEL command specifies a read-only file, DEL displays the following message before returning to command level without deleting the file:

```
Access denied
```

With the cursor positioned after the A > prompt, type

```
ATTRIB -R B:CHKDSK.COM
```

and press ↵. The **-R** parameter instructs the ATTRIB command to remove the read-only status from the specified file—in this case, CHKDSK.COM—on the diskette in the specified drive (drive B). The file CHKDSK.COM on the diskette in drive B has been returned to its original status.

At this point a few remarks should be made about the general use of the ATTRIB command.

1. Since read-only files cannot be modified or deleted, a read-only file should not be specified as the target file in a COPY command. If a read-only file is specified as the target file in a COPY command, DOS displays the following message on the screen:

   ```
   File creation error
   ```

   When this occurs, the COPY command is terminated without copying over the read-only file and control returns to command level.

2. Since read-only files cannot be modified or deleted, data files used with application programs such as a word processor should not be given a read-only status.

3. The read-only status of a group of files can be turned on or off by using wildcard file name characters to specify multiple file names in the ATTRIB command.

4. When a read-only file is copied with the COPY or XCOPY command, the read-only status is not transferred to the target file. That is, if SAMPFILE is a read-only file on the diskette in drive A and it is copied to the diskette in drive B with the command line COPY A:SAMPFILE B:, then SAMPFILE on the diskette in drive B is not a read-only file.

Keystroke Summary: Changing and Displaying the Read-Only File Access Attribute

Insert: DOS System diskette in drive A

Insert:	diskette in drive B that contains a file to have its read-only file access attribute changed or displayed
Set:	default drive to drive A
Type:	`ATTRIB +R B:filespec` to add read-only status
	`ATTRIB -R B:filespec` to delete read-only status
	`ATTRIB B:filespec` to display file access attributes
Press:	↵

Summary

Topic or Feature	Command or Reference	Page
Copy one user file to the same disk	`COPY source-file-name target-file-name`	67
	Fig. 5.2	69
Copy one user file to a second disk	`COPY source-file-name B:`	70
Copy selected user files	`COPY source-filespec B:`	73
	`COPY source-filespec B:target-filespec`	
	Figs. 5.3, 5.4	73, 75
Remove one user file name from a disk's directory	`DEL file-name`	77
Remove selected user file names from a disk's directory	`DEL filespec`	79
Change the name of a file	`RENAME current-file-name new-file-name`	85
Add read-only status to a file	`ATTRIB +R filespec`	89
Delete read-only status of a file	`ATTRIB -R filespec`	90
Display file access attributes	`ATTRIB filespec`	89

Self-Test

1. The ________________ command is used to copy a user file to the same disk or to a second disk. Is this command an internal or external DOS command?

   ____

2. Given the following DOS prompt(s), write the complete DOS command that copies the specified source file(s) to the specified target file(s).

   a. Copy the file CHKDSK.COM from the diskette in drive A to CHKDSK.BAK on the diskette in drive A.

A > ____

B > ____

b. Copy the file FORMAT.COM from the diskette in drive A to the diskette in drive B using the same file name.

A > ____

B > ____

3. The ________________ command is used to remove specified user file names from the directory of a disk. Is this command an internal or external DOS command? ________________

4. Given the following DOS prompt(s), write the complete DOS command that removes the specified user file names from the directory of the specified disk.

a. Remove the file name AUTO.BAT from the directory of the diskette in drive A.

A > ____

B > ____

b. Remove all user file names from the directory of the diskette in drive B that begin with the character T and have .DOC as an extension.

A > ____

B > ____

5. The ________________ command is used to change a file name in the directory of a disk. Is this command an internal or external DOS command?

____

6. Given the following DOS prompt(s), write the complete DOS command to change the file name DATA.DOC to TEXT.DOC in the directory of the diskette in drive A.

A > ____

B > ____

7. State four things that you can do to help prevent the accidental deletion of file names from the directory of a disk. ________________

8. The ________________ command is used to declare files as read-only files. Is this command an internal or external DOS command? ________________

Exercises

1. The DOS System diskette (or a diskette supplied by your instructor) should be used as the source diskette for this exercise. Use an appropriate DOS command to perform each of the assigned tasks.

a. Format a data diskette with volume label *namef#5-1,* where *name* denotes the first six characters of your last name or your entire last name (whichever is shorter) and *f* denotes the first character of your first name.

b. Copy all user files having file names that begin with the character F and have any extension from the source diskette to the data diskette using the same file names.

c. Use the output redirection operator to obtain a hard copy listing of the directory information for the data diskette.

d. Copy all user files having file names that begin with the character S and have .COM as an extension from the source diskette to the data diskette using the same filename but changing the extension to .EX.

e. Use the output redirection operator to obtain a hard copy listing of the directory information for the data diskette.

f. Copy all user files having file names that have .COM as an extension from the source diskette to the data diskette using the same filename but changing the extension to .BAK.

g. Use the output redirection operator to obtain a hard copy listing of the directory information for the data diskette.

h. Remove all user file names from the directory of the data diskette.

i. Use the output redirection operator to obtain a hard copy listing of the directory information for the data diskette.

2. The DOS System diskette (or a diskette supplied by your instructor) should be used as the source diskette for this exercise. Use an appropriate DOS command to perform each of the assigned tasks.

a. Format a data diskette with volume label *namef*#5-2, where *name* denotes the first six characters of your last name or your entire last name (whichever is shorter) and *f* denotes the first character of your first name.

b. Copy all user files from the source diskette to the data diskette using the same file names.

c. Turn the printer echo mode on so that you can obtain a hard copy listing of your work in parts (d) through (o).

d. Display the directory information for the data diskette.

e. Display all user file names that have .EXE as an extension from the directory of the data diskette.

f. Remove from the directory of the data diskette all user file names that have .EXE as an extension.

g. Display the directory information for the data diskette.

h. Display all user file names that begin with the character C and have .COM as an extension from the directory of the data diskette.

i. Remove from the directory of the data diskette all user file names that begin with the character C and have .COM as an extension.

j. Display the directory information for the data diskette.

k. Remove all user file names from the directory of the data diskette.

l. Display the directory information for the data diskette.

m. Turn the printer echo mode off.

6

Backing Up Diskettes and Files

The material in this chapter introduces the DOS commands that allow you to:

- back up an entire diskette by making an identical copy of a diskette using the DISKCOPY command
- back up individual and selected files using the COPY command
- interactively select files to back up using the XCOPY command
- back up files based on a specified date using the XCOPY command
- compare the contents of two files using the COMP command

Backup Copies of Files and Diskettes

Making backup copies of important files and complete diskettes minimizes the amount of time and information you might lose if something goes wrong with a particular file or an entire diskette. Even though you are very careful with your diskettes, they can be misplaced or damaged by accident, and files can be inadvertently modified or deleted. The amount of time and effort it takes to make these backup copies is small in comparison to the time and effort it would take to recreate lost information. Making backup copies of files and diskettes is a very wise investment of your time.

Making Identical Diskette Copies (DISKCOPY)

In Chapter 5 you learned how to copy user files from one diskette (source diskette) to another diskette (target diskette) using the COPY command. The COPY command cannot be used to copy hidden files from the source diskette to the target diskette. If you want to copy both hidden and user files from one diskette to another diskette, you must use a different command, such as the DISKCOPY command. The **DISKCOPY** command is an external DOS command that allows you to copy the entire contents of one diskette (source diskette) onto another diskette (target diskette) of the same format. If the target diskette is not formatted, the DISKCOPY command will automatically format the target diskette to the source diskette format. The DISKCOPY command reproduces the contents of the source diskette sector by sector on the target diskette.

Insert the DOS System diskette in drive A and boot the system. With the cursor positioned after the A > prompt, type

```
DISKCOPY A: B:
```

and press ↵. Since DISKCOPY is an external DOS command, it must be loaded into memory from the DOS System diskette in drive A. Once the loading process is complete, DOS initiates the execution of the DISKCOPY program, which displays the following messages on the screen:

```
Insert SOURCE diskette in drive A:

Insert TARGET diskette in drive B:

Press any key when ready . . .
```

Following the above messages, remove the DOS System diskette from drive A and insert the diskette to be copied (duplicated) in drive A. Insert a blank (target) diskette in drive B, and press any key to begin the copying process. The DISKCOPY program determines the format of the source diskette by reading information from its boot record. Once the format type of the source diskette is determined, a message similar to the following is displayed on the screen:

```
Copying 40 tracks
9 Sectors/Track, 2 Side(s)
```

The DISKCOPY program now begins to read the source diskette sector by sector, temporarily storing the information read in memory. After all of the information has been read from the source diskette (or no more memory is available to store information), the DISKCOPY program tests the format of the target diskette in drive B. If the target diskette has the same format type as the source diskette, the DISKCOPY program begins copying the information from memory onto the target diskette. If the target diskette is not formatted or if the target diskette has a different format type than the source diskette, a message similar to the following is displayed on the screen:

```
Formatting while copying
```

This message indicates that the DISKCOPY program is formatting the target diskette to agree with the format type of the source diskette. After the target diskette is formatted, the DISKCOPY program begins copying the information from memory onto the target diskette. The information is copied onto the target diskette sector by sector in exactly the same order that it was read from the source diskette. In other words, the target diskette is an identical copy of the source diskette. After the copying process is completed, the DISKCOPY program displays the following message on the screen:

```
Copy another diskette (Y/N)?
```

If you want to make a duplicate copy of another diskette, type Y and the above process will be repeated. If you want to terminate the DISKCOPY program, type N and control returns to command level with the A> prompt.

Keystroke Summary: Making Identical Diskette Copies

Insert:	DOS System diskette in drive A
Set:	default drive to drive A
Type:	`DISKCOPY A: B:`
Press:	↵
Remove:	DOS System diskette from drive A after the prompt messages about source and target diskettes are displayed
Insert:	diskette to be copied (source diskette) in drive A
Insert:	blank diskette (target diskette) in drive B
Press:	↵ or any key to begin the copying process
Type:	Y to copy another diskette or
	N to terminate the DISKCOPY program.

Disadvantages of Using the DISKCOPY Command

The DISKCOPY command is an easy command to use when making backup copies of diskettes. If you want to make an identical copy of a source diskette or if the source diskette has any hidden files that you want copied, then you should use the DISKCOPY command. There are, however, some disadvantages to using the DISKCOPY command.

- One disadvantage to using the DISKCOPY command occurs when the source diskette contains bad sectors. If the source diskette has any bad sectors on it, the same sectors on the target diskette will be marked as bad sectors. In fact, when the source diskette contains bad sectors, the DISKCOPY command will display messages similar to the following on the screen:

  ```
  Unrecoverable read error on drive A:
  Side 1, track 37
  Target diskette may be unusable
  ```

  The DISKCOPY command is informing you through these messages that the target diskette may not be a usable diskette.

- A second disadvantage to using the DISKCOPY command occurs when the target diskette contains bad sectors. If the target diskette contains bad sectors, the information written to these sectors may not be usable. The DISKCOPY command does not check the target diskette for bad sectors; it copies the source diskette sector by sector to the target diskette. Consequently, no message about this possible problem is ever displayed. This problem can be alleviated by first formatting the target diskette with the FORMAT command and then checking it for bad sectors with the CHKDSK command. If a diskette contains bad sectors, do not use it as the target diskette for the DISKCOPY command.

- A third disadvantage to using the DISKCOPY command involves the duplication of fragmented files. A **fragmented file** is a file that is stored on a disk in nonadjacent sectors. If the source diskette contains fragmented files, the file fragmentation is duplicated on the target diskette by the DISKCOPY program. Regular use of fragmented files will slow DOS's reading speed, since the disk drive's read/write heads will have to be moved to several disjointed locations on the disk when accessing a fragmented file.

Backing Up Files Using Two Diskette Drives

In the preceding section you learned how to use the DISKCOPY command to back up an entire diskette. This process actually produced a target diskette that was an identical copy of the source diskette. If you do not need an identical copy of a diskette or if you do not want to copy any hidden files, you may want to choose one of the procedures outlined in this section to back up files or to make backup copies (not identical copies) of diskettes. Three different methods are outlined in this section. The first method allows you to back up files on the basis of certain characteristics in the file names. The second method allows you to back up selected files by interactively selecting their file names from the directory of the source disk. The final method allows you to back up files that were created or last modified on or after a specified date.

Backing Up Files with the COPY Command

The COPY command, introduced in Chapter 5, is an internal DOS command that can be used to back up individual files, groups of files, or all files specified in the directory of a disk. The files to be backed up are defined in the source file specification part of the COPY command. The following example illustrates the use of the COPY command to back up all user files on a diskette.

Insert the diskette to be backed up (source diskette) in drive A. Insert a blank formatted diskette (target diskette) in drive B. With the cursor positioned after the A > prompt, type

```
COPY *.* B:
```

and press ↵. The COPY command copies all user files from the diskette in the default drive (drive A) to the diskette in the specified drive (drive B) using the same file names. If the directory of the diskette inserted in drive B is empty, the files are copied onto this diskette using contiguous sectors. This means that file fragmentation on the source diskette will be eliminated on the target diskette through the use of the above copying process.

Backing Up Files with the XCOPY Command (Interactive Selection)

The **XCOPY** command is an external DOS command that allows you to selectively copy files from a disk in one drive to a disk in another drive. Although the XCOPY command functions very much like the COPY command, it supports different ways to identify the files to be copied. One of the most commonly used options in the XCOPY command allows you to interactively select or reject the name of each file to be copied from the group of files specified in the XCOPY command.

Insert the DOS System diskette in drive A and a formatted target diskette in drive B. With the cursor positioned after the A > prompt, type

```
XCOPY A: B: /W /P
```

and press ↵. Since XCOPY is an external DOS command, it must be loaded into memory from the DOS System diskette in drive A. Once the loading process is complete, DOS initiates the execution of the XCOPY program. The **/ W** (wait) parameter instructs the XCOPY command to display the following message on the screen and wait before it starts to copy any files:

```
        Press any key to begin copying files
```

Remove the DOS System diskette from drive A, and insert in drive A the source diskette that contains files to be backed up. Now that the source diskette is in drive A and the target diskette is in drive B, press any key. The **/ P** (prompt) parameter instructs XCOPY to prompt you with (Y/N)? after each file name in the directory of the source diskette in drive A before it copies that file from the diskette in drive A to the diskette in drive B.

Note that a space does not have to be used before either slash (/) in the above XCOPY command. Since a slash is interpreted by DOS as a parameter field delimiter (separator), the spaces before the / W and / P parameters are used only to help make the command more readable.

The XCOPY command does not have to specify all file names in the directory of the source diskette. You can restrict the file specifications displayed by the XCOPY command to certain groups of files by using wildcard file name characters. The use of wildcard file name characters in the XCOPY command is analogous to their use in the COPY command.

Keystroke Summary: Copying Files with the XCOPY Command (Interactive Selection)

Insert:	DOS System diskette in drive A
Insert:	formatted target diskette in drive B
Set:	default drive to drive A
Type:	`XCOPY` *`source-filespec`* `B: /W /P`
Press:	↵
Remove:	DOS System diskette from drive A
Insert:	source diskette in drive A
Press:	↵ or any key to begin the copying process
Type:	`Y` to copy the specified file or
	`N` to not copy the specified file

Backing Up Files with the XCOPY Command (Dated Selection)

The XCOPY command can also be used to selectively copy files on the basis of a specified date. If a date parameter is included in the XCOPY command, the XCOPY command will copy all specified files that have creation or modification dates on or after the stated date. The general format of the date parameter in the XCOPY command is **/D:mm-dd-yy**, where mm denotes the month (01 through 12), dd denotes the day (01 through 31), and yy denotes the year (80 through 99, or 1980 through 1999). The / W (wait) and / P (prompt) parameters can also be used with the date parameter in the XCOPY command.

Keystroke Summary: Copying Files with the XCOPY Command (Dated Selection)

Insert:	DOS System diskette in drive A
Insert:	formatted target diskette in drive B
Set:	default drive to drive A
Type:	`XCOPY` *`source-filespec`* `B: /W /D:`*`mm-dd-yy`* where *mm* denotes the month (01 through 12), *dd* denotes the day (01 through 31), and *yy* denotes the year (80 through 99, or 1980 through 1999)
Press:	↵
Remove:	DOS System diskette from drive A
Insert:	source diskette in drive A
Press:	↵ or any key to begin the copying process

Comparing the Contents of Files (COMP)

The DOS commands introduced in the previous sections of this chapter are commands that deal with making backup copies of whole diskettes or selected files on a diskette. You might want to compare the contents of two files to determine whether the files contain exactly the same information. The **COMP** (COMPare) command is an external DOS command that allows you to compare the contents of two files or two sets of files and display any differences about the files on the screen. Refer to Appendix B for more information about the COMP command.

Summary

Topic or Feature	Command or Reference	Page
Make an identical copy of a diskette	`DISKCOPY A: B:`	95
Back up individual or selected files	`COPY` *`source-filespec`* `B:`	98
Back up files interactively	`XCOPY` *`source-filespec`* `B: /W /P`	98

Back up files by dated selection	`XCOPY` *`source-filespec`*
	`B: /W /D:mm-dd-yy`
		99

Compare two files	`COMP`	100

Self-Test

1. The ________________ command is used to make an identical copy of a diskette. Is this command an internal or external DOS command? ________________ This command can be used to copy both ________________ and ________________ files from one diskette to another diskette.
2. State three disadvantages of using the DISKCOPY command to make backup copies of diskettes. ________________
3. The ________________ command can be used to back up individual files, groups of files, or all files specified in the directory of a disk. Is this command an internal or external DOS command? ________________
4. The ________________ command can be used to allow you to interactively select or reject files to be copied. Is this command an internal or external DOS command? ________________ The ________________ parameter is used with this command to instruct it to wait before it starts to copy any files. The ________________ parameter is used with this command to prompt you with (Y/N)? after each specified file name before it copies that file from the source diskette to the target diskette.
5. The ________________ command can be used to selectively copy files on the basis of a specified date. The ________________ parameter is used with this command to instruct it to copy each source file that has a creation or modification date on or after the specified date.
6. The ________________ command is used to compare the contents of two files. Is this command an internal or external DOS command? ________________

Exercises

1. Use an appropriate DOS command to perform each of the assigned tasks.
   a. Format a blank diskette.
   b. Verify that the diskette has no bad sectors. If the diskette has bad sectors, repeat part (a) using a different diskette.
   c. Using the formatted diskette as the target diskette, make a duplicate (identical) copy of the DOS System diskette.
   d. Verify that the two DOS hidden files are on the duplicate copy of the DOS System diskette. Display their names.
   e. Assign the volume label SYSTEM-DISK to the duplicate copy of the DOS System diskette.
2. Use an appropriate DOS command to perform each of the assigned tasks.
   a. Format a bootable diskette with volume label DOS-DISK.
   b. Verify that the formatted diskette has no bad sectors and show that it contains the three DOS system files by displaying their names.
   c. Complete the duplication process of the DOS System diskette by copying all user files from the DOS System diskette to the diskette prepared in part (a).

7

Creating and Viewing Text Files

The material in this chapter introduces the DOS commands that allow you to:

- create a text file using the output redirection operator
- create a text file using the COPY CON command
- display information in a text file using the TYPE command
- print the contents of a text file using the PRINT command
- send the contents of a text file to the printer using the COPY command
- send the contents of a text file to the printer using the TYPE command with the output redirection operator

Text Files

Text (character, ASCII) files contain character information, that is, information similar to what is found on this page or in this book. Remember that ASCII (pronounced "asskey") is the acronym for American Standard Code for Information Interchange. ASCII refers to a code scheme that computers use to represent character type information and to facilitate the interchange of data (character information) among various types of computing equipment.

There are many ways in which text files can be created and viewed. Many application programs allow you to generate text (ASCII) files. For example, most word processing programs give you the option to create text (ASCII) files. DOS also contains commands that can be used to create and to view information in text files.

Creating Text Files

In this section, two methods will be introduced to show you how to create a text file using DOS commands. You should note that these methods are not the only ways in which text files can be created using DOS commands.

Creating a Text File Using the Output Redirection Operator

The output redirection operator > can be used with DOS commands like DIR and CHKDSK to create text files. Usually, these commands are used to display information on the screen. When they are used with the output redirection operator >, the text information that is normally sent to the screen can be redirected to a file.

Insert the DOS System diskette in drive A and boot the system. Insert any diskette containing a nonempty directory in drive A and a formatted diskette in drive B. With the cursor positioned after the A > prompt, type

```
DIR > B:DIRFILE.TXT
```

and press ↵. Remember that the DIR command is an internal DOS command, so the DOS System diskette does not have to be available to execute this command. The DIR command instructs DOS to generate the complete directory information from the directory of the diskette in the default drive (drive A). Instead of the directory information (output from the DIR command) being displayed on the screen, it is redirected by the output redirection operator > to the specified file—in this case, the file DIRFILE.TXT on the diskette in drive B. After DOS copies the directory information to the specified file, control returns to command level with the A > prompt being displayed on the screen. The file DIRFILE.TXT is called a text (ASCII) file.

Keystroke Summary: Creating a Text File Using the Output Redirection Operator >

Insert: any usable diskette containing a nonempty directory in drive A

Insert: formatted diskette in drive B

Set: default drive to drive A

Type: *command* > B:*filespec* where *command* denotes a DOS command that normally displays information on the screen

Press: ↵

Creating a Text File with the COPY CON Command

Recall from Chapter 5 that the COPY command is an internal DOS command that can be used to copy information from one file (source file) to a second file (target file) on the same or different disk. The source information for the COPY command does not have to come from a file. It can come directly from the keyboard. Consequently, the COPY command can be used to create text files by copying information entered directly from the keyboard console (CON) to the file specified in the command. This process is a convenient way to generate short text files.

Insert a formatted diskette in drive A. With the cursor positioned after the A > prompt, type

```
COPY CON A:INFO.TXT
```

and press ↵. After ↵ is pressed, the COPY command instructs DOS to position the cursor at the beginning of the next line and wait to capture information that is entered from the keyboard (CON). Type the following information on your keyboard:

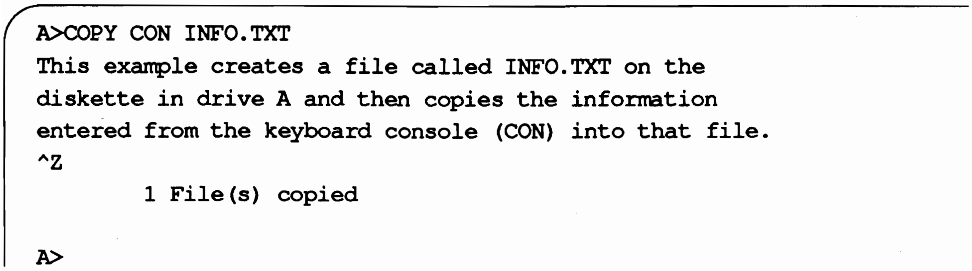
```
This example creates a file called INFO.TXT on the
diskette in drive A and then copies the information
entered from the keyboard console (CON) into that file.
```

At the end of each line of information that you type, you should press ↵. All information that you type is temporarily saved in the computer's memory. This process is terminated by entering an **end-of-file marker**. That is, hold the Ctrl key down and type Z (denoted by Ctrl *and* Z); then release the Ctrl key and press ↵. At this point, the COPY command copies the information entered from the keyboard and temporarily saved in memory to the target file specified in the COPY CON command—in this case, the file INFO.TXT on the diskette in drive A. When the copying process is completed, the message

```
1 File(s) copied
```

is displayed on the screen before control returns to command level with the A > prompt. The information displayed on the screen is similar to the information in Fig. 7.1. Note that the drive designator A: does not have to be specified in the above command, since drive A is the default drive.

This method for creating text files is used in Chapter 9 to create batch files. The external DOS command EDLIN (LINe EDitor) that allows you to modify a text file, is also introduced in Chapter 9.



```
A>COPY CON INFO.TXT
This example creates a file called INFO.TXT on the
diskette in drive A and then copies the information
entered from the keyboard console (CON) into that file.
^Z
        1 File(s) copied

A>
```

Figure 7.1 When the COPY CON command is used to create a text file, you can enter information from the keyboard, similar to the example displayed above. This information is copied as text to the file INFO.TXT on the diskette in the default drive (drive A).

Keystroke Summary: Creating a Text File with the COPY CON Command

Insert:	formatted diskette in drive A
Set:	default drive to drive A
Type:	`COPY CON` *`filespec`*
Press:	↵
Type:	information you want copied to the specified file
Press:	↵ at the end of each line
Enter:	end-of-file marker by holding the Ctrl key down and typing Z (Ctrl *and* Z)
Press:	↵

Displaying Information in a Text File (TYPE)

Once you know how to create a text file, you need to be able to display the information in a text file on the screen. The **TYPE** command is an internal DOS command that allows you to display the contents of a specified file on the standard output device (normally the screen). Although the TYPE command can be used to display the contents of any file, only text files are displayed in a legible format. Using the TYPE command to display information in a nontext file usually produces unreadable information because of the presence of special character codes, control characters, and escape sequences (nondisplayable characters) in the file. It might also cause strange things to happen to your personal computer. Your system might crash, your keyboard might lock up, or your screen might change characteristics. If this happens, you might have to restart (boot) your system.

Insert a diskette containing a text file in drive A. Assume for this example that the file name INFO.TXT specifies a text file on this diskette. With the cursor positioned after the A> prompt, type

```
TYPE INFO.TXT
```

and press ↵. The TYPE command instructs DOS to display the contents of the specified file—in this case, INFO.TXT on the diskette in the default drive (drive A)—on the screen. After the information in the specified file is displayed on the screen, control returns to command level with the A> prompt.

Keystroke Summary: Displaying Information in a Text File

Insert:	diskette containing a text file in drive A
Set:	default drive to drive A
Type:	`TYPE` *`text-filespec`*
Press:	↵

Printing the Contents of a Text File (PRINT)

In addition to being able to display a text file on the screen, DOS also contains a command that allows you to obtain a printed (hard) copy of the information in a text file. The **PRINT** command is an external DOS command that allows you to print the contents of a text file or group of text files on a printer while you are executing other DOS commands. Refer to Appendix B for more information about the PRINT command.

Keystroke Summary: Printing the Contents of a Text File

Insert:	DOS System diskette in drive A
Insert:	diskette containing a text file in drive B
Set:	default drive to drive A
Ready:	printer
Type:	`PRINT B:`*`text-filespec`*
Press:	↵
Press:	↵ to install the standard printer device PRN, or type the name of the printer device for your personal computer and press ↵

Sending the Contents of a Text File to the Printer

There are two disadvantages in using the PRINT command to print a text file on a printer. First, since the PRINT command is an external DOS command, it must be available on a disk in a drive before it can be used to print a text file. Second, the PRINT command always issues a page ejection command to the printer at the completion of the printing of each file. If the text files that you are printing are short, you might want to print more than one file on a page. The internal DOS commands COPY and TYPE can be used to send text file information to the printer and to eliminate the two disadvantages stated above.

Copying Text Files to the Printer (COPY filespec PRN)

The COPY command can be used to send a text file to the printer by using the printer specification (PRN) in place of the target file specification in the COPY command line. Insert a diskette containing a text file in drive A. Assume that the file name INFO.TXT specifies a text file on this diskette. With the cursor positioned after the A> prompt, type

```
COPY INFO.TXT PRN
```

and press ↵. The COPY command instructs DOS to copy from the default drive (drive A) the contents of the specified text file—in this case, INFO.TXT—to the printer that is denoted by the specification PRN. After the information in the text file is printed by the printer, the following message is displayed on the screen before control returns to command level with the A> prompt:

```
1 File(s) copied
```

The printer is now ready to print new information on the very next line. No line feed or page ejection commands are sent to the printer. Consequently, to separate the printing of the next text file from this one, you need to manually line feed the paper forward in the printer before printing the next text file.

Keystroke Summary: Copying Text Files to the Printer

Insert:	diskette containing a text file in drive A
Set:	default drive to drive A
Ready:	printer
Type:	`COPY` *`text-filespec`* `PRN`
Press:	↵

Redirecting the Contents of a Text File to the Printer (TYPE filespec > PRN)

The TYPE command can be used with the output redirection operator > to send the contents of a text file to the printer. Insert a diskette containing a text file in drive A. Assume that the file name INFO.TXT specifies a text file on the diskette. With the cursor positioned after the A> prompt, type

```
TYPE INFO.TXT > PRN
```

and press ↵. Normally, the TYPE command is used to display the contents of a text file on the screen. In the above command the output redirection operator > instructs DOS to direct the output of the TYPE command—in this case, the information in the specified file INFO.TXT on the diskette in the default drive (drive A)—to the printer that is denoted by PRN. After the information in the specified file is printed by the printer, control returns to command level with the A> prompt.

Keystroke Summary: Redirecting the Contents of a Text File to the Printer

Insert:	diskette containing a text file in drive A
Set:	default drive to drive A
Ready:	printer
Type:	`TYPE` *`text-filespec`* `> PRN`
Press:	↵

Summary

Topic or Feature	Command or Reference	Page
Create a text file	`COPY CON` *`filespec`* Fig. 7.1	104 105
Create end-of-file marker	Ctrl *and* Z	104
Display a text file	`TYPE` *`filespec`*	105
Print a text file	`PRINT` *`filespec`*	106
Copy a text file to the printer	`COPY` *`filespec`* `PRN`	107
Redirect a text file to the printer	`TYPE` *`filespec`* `> PRN`	107

Self-Test

1. The ________________ operator can be used to direct output information from a DOS command to a specified file. This operator is denoted by the ________________ symbol. Write the DOS command line necessary to send the directory information from the diskette in drive B to the file DIRINFO.TXT on the diskette in drive B. A> ________________
2. The ________________ command can be used to create a text file by copying information typed on the keyboard to a specified file. Is this command an internal or external DOS command? ________________

3. The ________________ command is used to display the contents of a specified file on the screen. Is this command an internal or external DOS command?

   ____

4. The ________________ command is used to print the contents of a text file on a printer while you are executing other DOS commands. Is this command an internal or external DOS command? ________________ State two disadvantages of using this command. ________________

5. The internal DOS commands ________________ and ________________ can be used to send the contents of a text file to a printer. Assume that the text file TEXTFILE.TXT is on the diskette in drive A. Write the DOS command line for each of the above commands to print the contents of TEXTFILE.TXT on the printer. A> ________________ A> ________________

Exercises

1. Insert the DOS System diskette (or a diskette supplied by your instructor) in drive A. Insert a formatted data diskette in drive B. Use an appropriate DOS command to perform each of the assigned tasks.

   a. Assign the volume label *namef*#7-1 to the data diskette, where *name* denotes the first six characters of your last name or your entire last name (whichever is shorter) and *f* denotes the first character of your first name.

   b. Use the output redirection operator to create a text file on the data diskette in drive B that contains the complete directory information for the diskette in drive A. Use DIRECT.TXT for the name of this text file.

   c. Create the text file KEYBOARD.TXT on the formatted diskette in drive B that contains the following information:

   ```
   Text files are data files that contain alphanumeric (both
   alphabetic and numeric) and special symbol characters in
   an ASCII format. ASCII (pronounced "asskey") is the
   acronym for American Standard Code for Information
   Interchange and refers to a code scheme that computers use
   to represent characters of information. Text files are
   sometime referred to as ASCII files or ASCII format files.
   ```

   d. Display the information in DIRECT.TXT and KEYBOARD.TXT on the screen.

   e. Print the contents of DIRECT.TXT and KEYBOARD.TXT on the printer.

2. Insert the DOS System diskette (or a diskette supplied by your instructor) in drive A. Insert a formatted data diskette in drive B. Use an appropriate DOS command to perform each of the assigned tasks.

   a. Boot DOS on your personal computer. If DOS is already running on your personal computer, do a system restart. Make sure that the correct date and time are entered.

b. Assign the volume label *namef*#7-2 to the data diskette, where *name* denotes the first six characters of your last name or your entire last name (whichever is shorter) and *f* denotes the first character of your first name.

c. Display how many bytes of total memory are installed in your personal computer and how many bytes of memory are free.

d. Use the output redirection operator to create a text file on the data diskette in drive B containing the output from the command in part (c). Use MEMCHK1.TXT for the name of this file.

e. Display the contents of MEMCHK1.TXT on the screen.

f. Use the PRINT command to print the contents of MEMCHK1.TXT on the printer.

g. Display how many bytes of total memory are installed in your personal computer and how many bytes of memory are free.

h. Use the output redirection operator to create a text file on the data diskette in drive B containing the output from the command in part (g). Use MEMCHK2.TXT for the name of the file.

i. Use the PRINT command along with a wildcard file name character to print the contents of the text files MEMCHK1.TXT and MEMCHK2.TXT on the printer.

j. Use an internal DOS command along with a wildcard file name character to print the contents of the text files MEMCHK1.TXT and MEMCHK2.TXT on the printer.

k. Use the output from the two files MEMCHK1.TXT and MEMCHK2.TXT to determine how many bytes of memory the resident part of the PRINT command uses when it is installed in memory.

8

Using a Hard Disk

The material in this chapter introduces the DOS commands that allow you to:

- manage files on a hard disk
- create subdirectories on a disk using the MKDIR or MD command
- change the current directory of a disk using the CHDIR or CD command
- display the current directory of a drive using the CD and DIR commands
- automatically display the current directory name using the PROMPT command
- copy files between different directories on a disk
- display information about the directory structure of a disk using the TREE and CHKDSK commands
- set an extended search path for DOS to locate commands using the PATH command
- delete (remove) subdirectories from a disk using the RMDIR or RD command

Background Information

This chapter is devoted primarily to DOS commands that are used to help manage information on a hard disk. You should note that each command presented in this chapter can also be applied to a diskette drive. Therefore, even if you do not have a personal computer with a hard disk, you can perform the commands presented in this chapter on a diskette in a diskette drive.

If your personal computer contains a hard disk, the material presented in this chapter assumes the following:

1. the hard disk is formatted,
2. DOS is installed on the hard disk,
3. all files from the DOS Startup diskette and the DOS Operating diskette are in the main directory of the hard disk, and
4. the hard disk drive is referenced as drive C.

If any of these assumptions is not correct, you should consult with your instructor.

Managing Disk Files

When you first begin to use a personal computer, you probably have three or four diskettes that you regularly use: a DOS System diskette, one or two diskettes containing application programs such as word processing programs, and a data diskette. Initially, you have one diskette to hold all of your data files. As the number of files grows, you realize that managing a large number of unrelated files on one data diskette is not a good idea. At this point you decide to separate your data files into similar categories and put each category of files onto a separate diskette containing an appropriate label. The same thing occurs with application programs. You separate different application programs by putting them on different diskettes. Through the use of multiple diskettes you are better able to manage your computer application programs and their associated data files.

Managing information on a hard disk is similar to managing information in an office file cabinet. Suppose you store letters, memorandums, reports, and other items in a file cabinet without using any labeled file folders and without using any topical organization of the information in the file cabinet. The lack of any file cabinet organization is fine until you try to locate a particular piece of information. The more information you have in an unorganized file cabinet, the more trouble you will have in trying to locate any particular item in the cabinet. Consequently, you put labels on file folders and separate groups of file folders according to certain common characteristics so that information can be retrieved in a timely manner. These same ideas carry over to storing files of information on a hard disk. You must organize your computer files in a manner that allows you to use a hard disk effectively. This is accomplished through the use of multiple directories.

Types of Directories

Recall that a directory is a table of contents that DOS maintains for information that is stored on a disk. The directory of a disk contains an entry for each file on the disk that consists of the filename, the filename extension, the file size in bytes, the date and time the file was created or last updated, and the location of the beginning of the file data on the disk.

Root Directory

A directory is created by DOS each time the FORMAT command is used to format a disk. This directory is called the **root (main) directory** of a disk. The main directory is commonly called the root directory because a multilevel directory structure, called a tree structure, can grow from it.

Subdirectories

To make your computer filing system more flexible, DOS allows you to create additional directories on a disk; these are called **subdirectories**. Subdirectories allow you to divide a disk into separate storage areas that can be used like separate disks. A subdirectory is a special file that contains directory entries. These entries have the same form as the entries in the root directory. Since a subdirectory is a directory, the terms subdirectory and directory may be used interchangeably in some places. This is similar to the idea that all members of your family are human beings, but at the same time you are also family members with special names and titles.

Naming Subdirectories

Subdirectory names follow the same format as file names. They contain one to eight characters followed by an optional extension that consists of a period and one to three characters. All valid characters for file names are also valid for subdirectory names. Normally subdirectory names are short and do not include extensions; this makes them convenient to use.

Current Directory

The **current directory associated with a disk** is the directory that you are currently working in or the directory that you were working in on another drive. DOS keeps track of the current directory in the same way that it keeps track of the current (default) disk drive. When the system is booted, the current drive is the drive containing the disk from which the DOS system programs were loaded into memory, and the current directory is the root directory of the disk in this drive. The terminology **current directory** is used to denote the current directory of the disk in the default drive. The terminology **current directory of a drive** is used to denote the current directory of the disk in that drive.

Directory Path

When DOS is attempting to locate a file and no particular directory is specified, DOS looks for the file only in the current directory. If the file is in another directory, DOS must be given directions on how to access that directory. A **directory path** or **path** is the route you must follow to trace your way from the disk's main or root directory to a particular place in the directory structure. The description of a path is called the **directory path name, path name**, or **path specification**.

Consider the multilevel directory structure in Fig. 8.1. Note that this tree structure grows out of the root directory. The root directory contains two subdirectories SD-1 and SD-2. Directory SD-1 contains no subdirectories, and directory SD-2 contains two subdirectories (SD-21 and SD-22).

Suppose that you want to access a file in subdirectory SD-21. You get to this directory through the following path of directories: root directory to subdirectory SD-2 to subdirectory SD-21. The path name (specification) for this path is \SD-2\SD-21. In this specification the initial backslash (\) instructs DOS to start the path at the root directory, while the second backslash is used as a directory name delimiter.

Suppose that the file name EXAMPLE.DOC is contained in subdirectory SD-21. The following specification contains the directory path name and the file name and denotes the complete specification for the file EXAMPLE.DOC:

```
\SD-2\SD-21\EXAMPLE.DOC
```

Note that the specification begins with the root directory and gets to the file EXAMPLE.DOC in subdirectory SD-21 by going through subdirectory SD-2.

Path Name Notation

A path consisting of only the **backslash (\)** specifies the root directory. A path name beginning with a backslash (\) specifies that the path starts in the

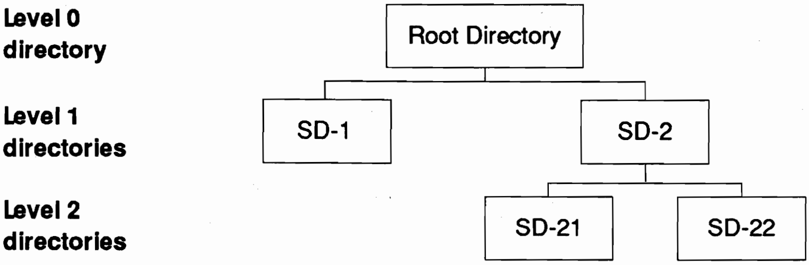


Figure 8.1 Multilevel directory structure.

root directory. The backslash (\) is also used as a delimiter between two directory names and between a directory name and a file name. Recall that a space is not a valid character for a file specification or path specification, since it is interpreted by DOS as a parameter field delimiter (separator). Thus, spaces should never be used in a file or path specification. Figure 8.2 illustrates how the backslash is used in a file specification that contains a directory path.

Directory Structure Terminology

If U and V are subdirectories of a directory T, then U and V are **offspring directories** of T, and T is called the **parent directory** for U and V. U and V are called **sibling directories**. Using the notation of Fig. 8.1, subdirectories SD-21 and SD-22 are sibling directories and have directory SD-2 as their parent directory. Subdirectory SD-21 is an offspring directory of directory SD-2.

Subdirectory Commands

You are now ready to create and use a directory structure similar to the one given in Fig. 8.1. Since subdirectories are special types of files, special DOS commands must be used to manipulate them. That is, special DOS commands must be used to create (MKDIR, MD) a subdirectory, change (CHDIR, CD) the current directory to a different directory, and remove (RMDIR, RD) a subdirectory.

If your personal computer contains a hard disk, C> is the DOS prompt for this drive. If you do not have a hard disk on your personal computer, insert a

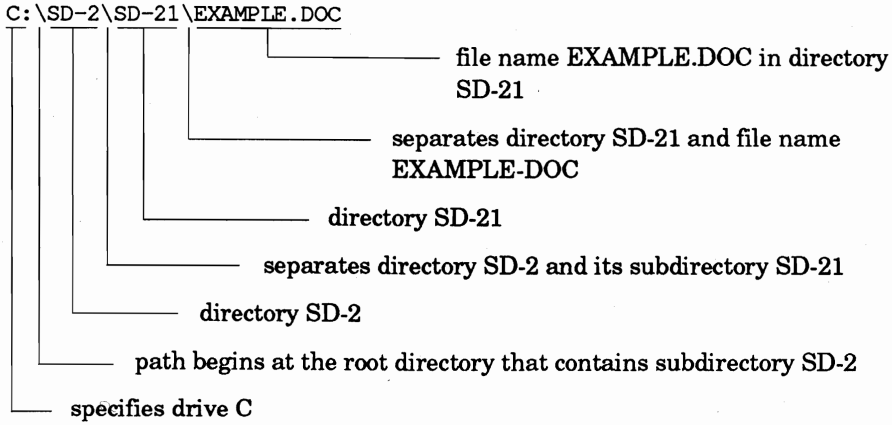

C:\SD-2\SD-21\EXAMPLE.DOC
file name EXAMPLE.DOC in directory SD-21
separates directory SD-21 and file name EXAMPLE-DOC
directory SD-21
separates directory SD-2 and its subdirectory SD-21
directory SD-2
path begins at the root directory that contains subdirectory SD-2
specifies drive C

Figure 8.2 This example illustrates how a backslash (\) is used in a file specification that contains a directory path. Note that the file specification contains no spaces, since a space is not a valid character for a file or path specification.

bootable diskette in drive A that contains the three external DOS commands CHKDSK, FORMAT, and TREE, and use drive A instead of drive C to practice the DOS commands covered in the next several sections.

Creating (Making) Subdirectories (MKDIR, MD)

The **MKDIR** (MaKe DIRectory) command is an internal DOS command that instructs DOS to create a subdirectory with the directory path and name being specified in the command. **MD** is an abbreviation of the MKDIR command. Since MD requires fewer keystrokes than MKDIR, MD is used in all examples in this chapter. The MD command is used in the next two sections to create the directory structure of Fig. 8.1.

Creating Subdirectories in the Root Directory (First Level)

A **level 1 directory** in a directory structure is defined as a subdirectory of the root directory. The examples presented in this section illustrate the creation of subdirectories in the root directory. In particular, the level 1 directories SD-1 and SD-2 of the directory structure in Fig. 8.1 are created. Assume that the current directory of the hard disk in drive C is the root directory. With the cursor positioned after the C> prompt, type

```
MD SD-1
```

and press ↵. With the cursor positioned after the C> prompt, type

```
MD SD-2
```

and press ↵. The MD commands instruct DOS to create the specified subdirectories—in this case, SD-1 and SD-2—in the root directory of the hard disk in drive C. After DOS creates each subdirectory, control returns to command level with the C> prompt.

The DIR command can be used to verify that the two subdirectories were created in the root directory of the hard disk in drive C. With the cursor positioned after the C> prompt, type

```
DIR SD-*
```

and press ↵. The DIR command instructs DOS to display all entries in the current directory of the default drive (drive C) having SD- as their first three characters. DOS displays information similar to the following on the screen:

```
 Volume in drive C is HARD-DISK
 Directory of  C:\

SD-1         <DIR>       1-31-89  10:46p
SD-2         <DIR>       1-31-89  10:46p
        2 File(s)  31139840 bytes free
```

Note that C:\ in the second line indicates that the two listed directories are subdirectories of the root directory in drive C. Also note that <DIR> specifies that the listed names are directory names.

The DIR command can also be used to display the contents of a subdirectory. With the cursor positioned after the C> prompt, type

```
DIR SD-1
```

and press ⏎. The DIR command instructs DOS to display the contents of the specified directory—in this case, subdirectory SD-1 on the disk in the default drive (drive C). DOS displays information similar to the following on the screen:

```
 Volume in drive C is HARD-DISK
 Directory of  C:\SD-1

.            <DIR>      1-31-89  10:46p
..           <DIR>      1-31-89  10:46p
       2 File(s)  31139840 bytes free
```

Note that the newly created directory SD-1 contains two special directory entries. The **single-period entry (.)** in a subdirectory represents the current directory. The **double-period entry (..)** in a subdirectory represents its parent directory. All directories except the root directory will always contain these two entries.

Keystroke Summary: Creating Subdirectories in the Root Directory (First Level)

Set: default drive to drive C with the current directory set to the root directory

Type: `MD` *`subdirname`*

Press: ⏎

Creating Second-Level Subdirectories from the Root Directory

A **level 2 directory** in a directory structure is defined as a subdirectory of a level 1 directory. The examples presented in this section illustrate the creation of second-level subdirectories from the root directory. In particular, the level 2 directories SD-21 and SD-22 of the directory structure in Fig. 8.1 are created. Note that directories SD-21 and SD-22 are subdirectories of directory SD-2, a level 1 directory. Assume that the current directory of the hard disk in drive C is the root directory. With the cursor positioned after the C> prompt, type

```
MD \SD-2\SD-21
```

and press ↵. The MD command instructs DOS to create the specified subdirectory—in this case, subdirectory SD-21 in directory SD-2 on the disk in the default drive (drive C). Note that \SD-2 in the above command specifies a directory path that begins with the root directory. Also note that the leading backslash in the path specification of the above command is not necessary, since the current directory is the root directory and SD-2 is a subdirectory of the root directory.

With the cursor positioned after the C> prompt, type

```
MD SD-2\SD-22
```

and press ↵. The MD command instructs DOS to create the specified subdirectory—in this case, subdirectory SD-22 in directory SD-2 on the disk in the default drive (drive C). Note that the path specification SD-2 in the above MD command does not begin with a backslash, since the current directory is the root directory and since SD-2 is a subdirectory of the root directory. The creation of the multilevel directory structure given in Fig. 8.1 is now complete.

Again, the DIR command can be used to verify that directories SD-21 and SD-22 were created in directory SD-2. With the cursor positioned after the C> prompt, type

```
DIR SD-2
```

and press ↵. The DIR command instructs DOS to display the contents of the specified directory—in this case, subdirectory SD-2 on the disk in the default drive (drive C). The DIR command knows that SD-2 is a directory name because of the <DIR> designator for SD-2 in the root directory. Thus, DOS displays information similar to the following on the screen:

```
 Volume in drive C is HARD-DISK
 Directory of  C:\SD-2

.            <DIR>       1-31-89  10:46p
..           <DIR>       1-31-89  10:46p
SD-21        <DIR>       1-31-89  10:50p
SD-22        <DIR>       1-31-89  10:50p
        4 File(s)   31135744 bytes free
```

Note that directory SD-2 contains four subdirectory entries, the two special directory entries (. for the current directory and .. for its parent directory) and the two newly created directories SD-21 and SD-22.

Once you have created several directories on a disk, you can use the CHKDSK command to give you information about the amount of disk space that is occupied by files and directories on the disk. With the cursor positioned after the C> prompt, type

```
CHKDSK
```

and press ↵. After the CHKDSK program is loaded into memory from the current directory of the disk in the default drive—in this case, the root directory of the hard disk in drive C—it displays a status report for the disk in the default drive (drive C) on the screen. The information displayed in the status report is similar to the information in Fig. 8.3. Note that the information in Fig. 8.3 now includes a specification for the number of bytes used in directories created on the disk. In this case, 8192 bytes are used by four directory entries. This means that 2048 bytes of disk space were allocated for each created directory. In general, the type of disk determines the number of bytes that DOS allocates for each subdirectory that it creates on a disk.

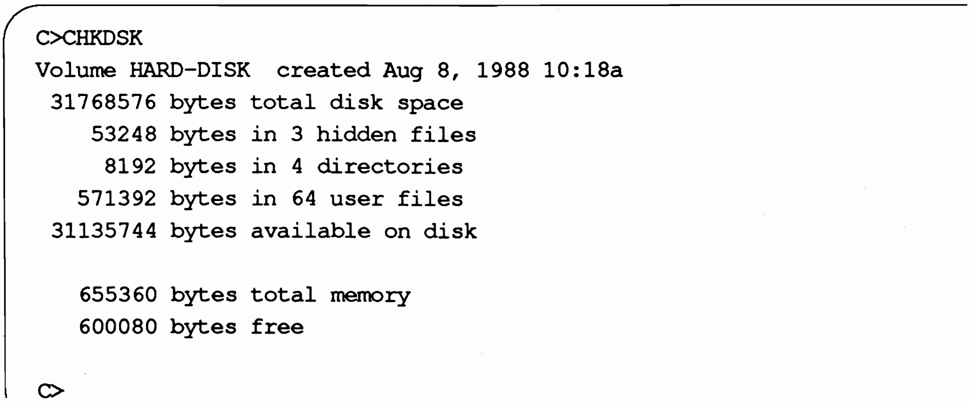
Note that all of the subdirectories in Fig. 8.1 have unique names. DOS allows you to create two or more subdirectories with the same name as long as they have different path specifications. For example, the three commands

```
MD \SD-1\DATA

MD \SD-2\DATA

MD \SD-2\SD-21\DATA
```

create three different subdirectories with the name DATA. The first is a subdirectory of SD-1 with path specification \SD-1\DATA; the second is a subdirectory of SD-2 with path specification \SD-2\DATA; and the third is a subdirectory of SD-21 with path specification \SD-2\SD-21\DATA. The directory names being the same in the first two examples is similar to two cousins having the same name. The directory names being the same in the last two examples is similar to a child having the same name as a parent.



```
C>CHKDSK
Volume HARD-DISK  created Aug 8, 1988 10:18a
 31768576 bytes total disk space
    53248 bytes in 3 hidden files
     8192 bytes in 4 directories
   571392 bytes in 64 user files
 31135744 bytes available on disk

   655360 bytes total memory
   600080 bytes free

C>
```

Figure 8.3 When the CHKDSK command is used to generate a status report for the hard disk in drive C, information similar to the above is displayed on the screen.

Keystroke Summary: Creating Second-Level Subdirectories from the Root Directory

Set: default drive to drive C with the current directory set to the root directory

Type: `MD` *Level-1-subdirname\Level-2-subdirname*

Press: ↵

Changing the Current Directory (CHDIR, CD)

Once a directory has been created, it can be referenced by specifying the appropriate path to the directory. DOS allows you to declare one directory on a disk as the current directory. This means that you can use a particular directory without having to specify its path. The material in this section shows you how to declare any directory on a disk as the current directory. The **CHDIR** (CHange DIRectory) command is an internal DOS command that instructs DOS to change the current directory of a disk to the directory specified in the command. This command can also be used to display the current directory. **CD** is an abbreviation of the CHDIR command. Since CD requires fewer keystrokes than CHDIR, CD is used in all examples in this chapter. Remember that a leading backslash (\) in a path name tells DOS to begin the path at the root directory.

Changing the Current Directory to a Subdirectory of the Current Directory

The material presented in this section shows you how to change the current directory to a subdirectory of the current directory. Assume that the directory structure in Fig. 8.1 is on the hard disk in drive C and that the current directory of the hard disk is the root directory. With the cursor positioned after the C> prompt, type

```
CD SD-2
```

and press ↵. The CD command instructs DOS to change the current directory of the disk in the default drive—in this case, the root directory of the hard disk in drive C—to the specified directory—in this case, subdirectory SD-2. After the current directory of drive C is changed to SD-2, control returns to command level with the C> prompt. With the cursor positioned after the C> prompt, type

```
CD SD-21
```

and press ↵. The CD command instructs DOS to change the current directory of the disk in the default drive—in this case, directory SD-2 of the hard disk in drive C—to the specified directory—in this case, subdirectory SD-21, a subdirectory of SD-2.

Keystroke Summary: Changing the Current Directory to a Subdirectory of the Current Directory

Set: default drive to drive C with *subdirname* a subdirectory of the current directory

Type: `CD` *`subdirname`*

Press: ↵

Changing the Current Directory to the Root Directory

The material presented in this section shows you how to change the current directory to the root directory from any directory on the disk. Assume that the directory structure in Fig. 8.1 is on the hard disk in drive C and that the current directory of the hard disk is subdirectory SD-21. With the cursor positioned after the C> prompt, type

```
CD \
```

and press ↵. The command instructs DOS to change the current directory of the disk in the default drive (drive C) to the root directory.

Keystroke Summary: Changing the Current Directory to the Root Directory

Set: default drive to drive C

Type: `CD \`

Press: ↵

Displaying the Current Directory of a Drive

When the root directory of the hard disk in drive C is the current directory, C> is the DOS command prompt. When subdirectory SD-2 of the hard disk in drive C is the current directory, C> is the DOS command prompt. When subdirectory SD-21 of the hard disk in drive C is the current directory, C> is the DOS command prompt. How do you determine the current directory of a disk drive? The CHDIR (CD) command can be used without any parameters to display the path specification for the current directory of the disk in the default drive. In addition to displaying entries in a directory, the DIR command can also be used to display the current directory of a drive.

Assume that the directory structure in Fig. 8.1 is on the hard disk in drive C and that the current directory of the hard disk is the root directory. With the cursor positioned after the C> prompt, type

```
CD
```

and press ↵. The CD command instructs DOS to display the path specification for the current directory of the default drive (drive C) on the screen. The following directory path specification is displayed on the screen before control returns to command level with the C> prompt:

```
C:\
```

This specification indicates that the current directory of the default drive (drive C) is the root directory. With the cursor positioned after the C> prompt, type

```
CD SD-2
```

and press ⏎. This command instructs DOS to change the current directory of the default drive (drive C) to subdirectory SD-2. With the cursor positioned after the C> prompt, type

```
CD
```

and press ⏎. The CD command instructs DOS to display the path specification for the current directory of the default drive on the screen. The following directory path specification is displayed on the screen before control returns to command level with the C> prompt:

```
C:\SD-2
```

This specification indicates that the current directory of the default drive (drive C) is directory SD-2, a subdirectory of the root directory.

With the cursor positioned after the C> prompt, type

```
DIR
```

and press ⏎. When the DIR command is used without any parameters, it instructs DOS to display on the screen the contents of the current directory for the default drive—in this case, the contents of directory SD-2 on the hard disk in drive C. The DIR command displays directory information similar to the following on the screen:

```
 Volume in drive C is HARD-DISK
 Directory of  C:\SD-2

.            <DIR>      1-31-89  10:46p
..           <DIR>      1-31-89  10:46p
SD-21        <DIR>      1-31-89  10:50p
SD-22        <DIR>      1-31-89  10:50p
        4 File(s)  31135744 bytes free
```

Note that the second line of the directory information states that \SD-2 is the path specification for the current directory on the hard disk in drive C.

With the cursor positioned after the C> prompt, type

```
CD SD-21
```

and press e. This command instructs DOS to change the current directory of the default drive (drive C) to subdirectory SD-21. With the cursor positioned after the C> prompt, type

```
CD
```

and press ↵. The CD command instructs DOS to display the path specification for the current directory of the default drive on the screen. The following directory path specification is displayed on the screen before control returns to command level with the C> prompt:

```
C:\SD-2\SD-21
```

This specification indicates that the current directory of the default drive (drive C) is directory SD-21, a subdirectory of directory SD-2, which is a subdirectory of the root directory.

Even though the CD command can be used to display the current directory specification, the normal DOS prompt (C>) is of no help to you in determining the current directory. When the current directory was changed in the above examples, the DOS prompt (C>) did not change. Later in this chapter you will learn how to use the PROMPT command to instruct DOS to replace the normal DOS prompt with a prompt that displays both the default drive and the path specification for the current directory.

Keystroke Summary: Displaying the Current Directory of a Drive

Set:	default drive to drive C
Type:	CD
Press:	↵

Changing the Current Directory with a Path from the Root Directory

The material presented in this section shows you how to change the current directory to a different directory by specifying its path from the root directory. Assume that the directory structure in Fig. 8.1 is on the hard disk in drive C. In each of the following examples, notice that the path specified in each CD command begins with the root directory of the hard disk, since the first character of the directory path specification is a backslash. With the cursor positioned after the C> prompt, type

```
CD \SD-2\SD-22
```

and press ↵. The CD command instructs DOS to change the current directory of the disk in the default drive (drive C) to the specified directory—in this case, directory SD-22, a subdirectory of directory SD-2, which is a subdirectory of the root directory.

With the cursor positioned after the C> prompt, type

```
CD \SD-1
```

and press ↵. The CD command instructs DOS to change the current directory of the disk in the default drive (drive C) to the specified directory—in this case, directory SD-1, a subdirectory of the root directory.

Note that every directory on a disk can be specified by a path that begins with the root directory. Thus, you do not need to know the current directory to use the CD command to change the current directory to another directory. To change the current directory of the default drive to a new directory using the CD command, just specify the path of the new directory that begins with the root directory.

Keystroke Summary: Changing the Current Directory with a Path from the Root Directory

Set: default drive to drive C

Type: `CD` *`pathname`* where *pathname* is a valid directory path name that begins with a backslash (\)

Press: ↵

Creating Subdirectories of the Current Directory

Recall from the presentation of the MKDIR (MD) command that all subdirectories of the directory structure in Fig. 8.1 were created from the root directory by specifying a path that began in the root directory. The CHDIR (CD) command allows you to change directories so that you can create subdirectories of the current directory. This means that you do not have to type long directory path specifications when you create directories. For example, the commands MD SD-2\SD-21 and MD SD-2\SD-22 were used in a previous section to create subdirectories SD-21 and SD-22 from the root directory of the hard disk in drive C. A better method for creating these subdirectories is given below.

Assume that the current directory of the hard disk in drive C is the root directory. Assume that directories SD-1 and SD-2 are the only subdirectories of the directory structure in Fig. 8.1 that have been created. With the cursor positioned after the C> prompt, type

```
CD SD-2
```

and press ↵. The CD command instructs DOS to change the current directory of the default drive (drive C) to subdirectory SD-2. With the cursor positioned after the C> prompt, type

```
MD SD-21
```

and press ↵. With the cursor positioned after the C> prompt, type

```
MD SD-22
```

and press ↵. The two MD commands instruct DOS to create directories SD-21 and SD-22, respectively, as subdirectories of the current directory (SD-2) of the default drive (drive C).

This example illustrates the normal way in which the MD command is used to create subdirectories of a directory. First use the CD command to change the current directory of the default drive to the appropriate directory. Then use the MD command to create subdirectories of the current directory.

Automatically Displaying the Current Directory Name with PROMPT

When you use a disk that has several subdirectories, you need a convenient way to keep track of the current directory. The PROMPT command can be used to perform this task. The **PROMPT** command is an internal DOS command that allows you to change the normal DOS prompt to a customized DOS prompt. You can define a customized DOS prompt in the PROMPT command that includes the default drive and the path specification for the current directory. The PROMPT command instructs DOS to automatically display the customized prompt in place of the normal DOS prompt.

Assume that the directory structure given in Fig. 8.1 is on the hard disk in drive C and that the current directory of the hard disk is the root directory. With the cursor positioned after the C> prompt, type

```
PROMPT $P$G
```

and press ↵. The PROMPT command instructs DOS to change the DOS prompt to the character string specified by PG. $P defines the first part of the new prompt to be the current directory specification. $G defines > to be the last character in the new prompt. When control returns to command level, the following prompt is displayed on the screen:

```
C:\>
```

The new prompt indicates that drive C is the default drive and that the current directory is the root directory. With the cursor positioned after the C:\> prompt, type

```
CD SD-2\SD-21
```

and press ↵. The CD command instructs DOS to change the current directory of the hard disk in the default drive (drive C) to the specified directory—in this case, subdirectory SD-21. When control returns to command level, the following prompt is displayed on the screen:

```
C:\SD-2\SD-21>
```

The new prompt indicates that drive C is the default drive and that the current directory is subdirectory SD-21.

With the cursor positioned after the C:\SD-2\SD-21> prompt, type

```
PROMPT
```

and press ↵. When the PROMPT command is used without any parameters, it instructs DOS to reset its prompt to the normal DOS prompt. Notice that the normal DOS prompt C> is displayed on the screen after control returns to command level.

Keystroke Summary: Changing the Normal DOS Prompt with the PROMPT Command

Set:	default drive to any drive
Type:	`PROMPT $P$G`
Press:	↵

Copying Files between Different Directories

Now that you know how to specify paths to different directories, you can use the COPY and XCOPY commands to copy files from one directory to another directory in the same way that you copy files between diskettes. Recall from the beginning of this chapter that the assumption was made that all files on the DOS Startup diskette and the DOS Operating diskette are in the root directory of the hard disk in drive C.

Copying Files from the Root Directory to a Subdirectory

The material presented in this section shows you how to copy files from the root directory of a disk to a subdirectory of a disk. Assume that the directory structure in Fig. 8.1 is on the hard disk in drive C and that the current directory of the hard disk is the root directory. With the cursor positioned after the C> prompt, type

```
COPY CHKDSK.COM SD-1
```

and press ↵. The COPY command instructs DOS to copy the specified file CHKDSK.COM from the current directory (root directory) of the default drive (drive C) to the specified subdirectory SD-1 (target directory) of the default drive using the same file name. Note that when the target parameter of the COPY command specifies a directory, the file name specified in the source parameter is used as the target file name. The complete specification for the copied file is C:\SD-1\CHKDSK.COM. With the cursor positioned after the C> prompt, type

```
COPY FORMAT.COM SD-1\FRMT.*
```

and press ↵. The COPY command instructs DOS to copy the specified file FORMAT.COM from the current directory (root directory) of the default drive (drive C) to the specified subdirectory SD-1 of the default drive (drive C) using FRMT.COM as the target file name.

With the cursor positioned after the C> prompt, type

```
DIR SD-1
```

and press ⏎. The DIR command instructs DOS to display the contents of directory SD-1 for the default drive (drive C). Information similar to the following is displayed on the screen:

```
 Volume in drive C is HARD-DISK
 Directory of  C:\SD-1

.            <DIR>       1-31-89  10:46p
..           <DIR>       1-31-89  10:46p
CHKDSK    COM     9850   3-18-87  12:00p
FRMT      COM    11616   3-18-87  12:00p
        4 File(s)  31113216 bytes free
```

Note that subdirectory SD-1 now contains four entries. Two of the entries are directories, and two of the entries are the names of files that were just copied to directory SD-1.

With the cursor positioned after the C> prompt, type

```
COPY F*.COM SD-2\SD-22\*.BAK
```

and press ⏎. The COPY command instructs DOS to copy from the current directory (root directory) of the default drive (drive C) each file having the same first character in the filename—in this case, F—and the same extension—in this case, .COM—to the specified subdirectory SD-22 of the default drive (drive C) using the same filename with the specified extension .BAK.

With the cursor positioned after the C> prompt, type

```
DIR SD-2\SD-22
```

and press ⏎. The DIR command instructs DOS to display the contents of directory SD-22 for the default drive (drive C). Information similar to the following is displayed on the screen:

```
 Volume in drive C is HARD-DISK
 Directory of  C:\SD-2\SD-22

.            <DIR>       1-31-89  10:50p
..           <DIR>       1-31-89  10:50p
FDISK     BAK    48216   3-18-87  12:00p
FORMAT    BAK    11616   3-18-87  12:00p
        4 File(s)  31051776 bytes free
```

Note that subdirectory SD-22 now contains the two files copied to it by the above COPY command.

Copying Files from One Subdirectory to Another Subdirectory

The material presented in this section shows you how to copy files from one subdirectory to a different subdirectory. This method involves first changing the current directory to the directory that contains the files to be copied and then copying the files from the current directory to the specified directory. Assume that the directory structure in Fig. 8.1 is on the hard disk in drive C and that the current directory of the hard disk is the root directory. With the cursor positioned after the C> prompt, type

```
        CD SD-1
```

and press ↵. The CD command instructs DOS to change the current directory of the default drive (drive C) to the specified directory SD-1. With the cursor positioned after the C> prompt, type

```
        COPY CHKDSK.COM \SD-2\SD-21
```

and press ↵. The COPY command instructs DOS to copy the specified file CHKDSK.COM from the current directory of the default drive—in this case, subdirectory SD-1 of drive C—to the specified subdirectory SD-21 of the default drive (drive C) using the same file name. Note that the target directory specification begins with a backslash (\). The leading backslash is necessary because directory SD-2 is not a subdirectory of the current directory SD-1 on the default drive (drive C). Consequently, the target directory specification needs to begin with the root directory to define a valid directory path to subdirectory SD-21.

Displaying Information in a Directory Structure

In several of the previous sections the DIR command was used to display the contents of a particular directory. The DIR command, however, does not give you a complete picture of the directory structure of a disk. The **TREE** command is an external DOS command that allows you to display information describing the directory structure of a disk. Refer to Appendix B for more information about the TREE command.

The CHKDSK command can be used with the /V (view) parameter to display the path specification for each directory and the file specifications for all entries in each directory on a disk. Refer to Chapter 4 for more information about the CHKDSK /V command.

Setting an Extended Search Path for DOS to Locate Commands (PATH)

Recall that an external DOS command must be loaded into memory from a disk before it can be executed. If such a command is entered and if no path specification precedes the command name, DOS searches only the current directory of the disk in the default drive to find the command. If the command is not in the current directory, DOS displays the following message on the screen:

```
        Bad command or file name
```

When you boot the system on your personal computer, the default search path is set to null (no path). This means that DOS searches only the current directory for an external DOS command. The **PATH** command is an internal DOS command that allows you to extend the **search path** that DOS uses to locate an external DOS command. When an extended search path is set by the PATH command, DOS uses the extended search path to search for external commands and batch files (files having an extension of .COM, .EXE, or .BAT) that were not found while searching the current directory.

The following examples illustrate how DOS searches only the current directory and directories specified in the PATH command to locate external DOS commands on a disk. The CHKDSK command is used as a sample external DOS command in the following examples. The information presented applies to any external DOS command. Assume that the directory structure in Fig. 8.1 is on the hard disk in drive C and that the current directory of the hard disk is the root directory. With the cursor positioned after the C> prompt, type

```
        PATH
```

and press ↵. The PATH command instructs DOS to display the following message on the screen before control returns to command level with the C> prompt:

```
        No Path
```

With the cursor positioned after the C> prompt, type

```
        CHKDSK
```

and press ↵. Since there is no extended search path, DOS searches for the CHKDSK program in the current directory—in this case, the root directory of the disk in the default drive (drive C). Since the CHKDSK program is located in the current directory, it is loaded into memory and executed.

With the cursor positioned after the C> prompt, type

```
        CD SD-2
```

and press ↵. The CD command changes the current directory of the disk in the default drive (drive C) to subdirectory SD-2. With the cursor positioned after the C> prompt, type

```
        CHKDSK
```

and press ↵. DOS searches the current directory of the disk in the default drive—in this case, subdirectory SD-2 on the hard disk in drive C—to find the CHKDSK program. Since the CHKDSK program is not in subdirectory SD-2, DOS displays the following message on the screen:

```
Bad command or file name
```

The next example shows you how to instruct DOS to look in a specified directory for an external DOS command. With the cursor positioned after the C> prompt, type

```
\CHKDSK
```

and press ↵. The leading backslash (\) specifies a path to the root directory of the default drive (drive C). Consequently, the path specification (\) before the command name instructs DOS to look in the specified directory—in this case, the root directory of drive C—for the CHKDSK program.

The PATH command can be used to automatically tell DOS to search a particular directory (or directories) if the specified command is not found in the current directory. With the cursor positioned after the C> prompt, type

```
PATH \
```

and press ↵. The PATH command instructs DOS to set the extended search path to the root directory of the default drive. With the cursor positioned after the C> prompt, type

```
PATH
```

and press ↵. This PATH command instructs DOS to display the extended search path that was set in the first PATH command. The following information is displayed on the screen:

```
PATH=\
```

With the cursor positioned after the C> prompt, type

```
CHKDSK
```

and press ↵. DOS searches for the CHKDSK program in the current directory of the disk in the default drive—in this case, subdirectory SD-2 on the hard disk in drive C. When DOS does not find the CHKDSK program in the current directory, it checks to see whether an extended search path is defined. DOS determines that the root directory of the disk in the default drive (drive C) is specified as an extended search path, and this is where DOS finds the CHKDSK program.

Insert a blank formatted diskette in drive A. With the cursor positioned after the C> prompt, type `A:` and press ↵ to change the default disk drive to drive A. With the cursor positioned after the A > prompt, type

```
CHKDSK
```

and press ↵. DOS searches the current directory of the disk in the default drive (drive A) and then the root directory (extended search path) of the disk in the default drive (drive A) for the CHKDSK program. DOS cannot find the

CHKDSK program on the diskette in drive A, so it displays the following message on the screen:

```
Bad command or file name
```

Since the DOS commands are in the root directory of the hard disk in drive C, the search path must specify more than the root directory of the disk in the default drive. It needs to specify the root directory of the hard disk in drive C.

With the cursor positioned after the A > prompt, type

```
PATH C:\
```

and press ↵. This PATH command instructs DOS to set the extended search path to the root directory of the hard disk in drive C. With the cursor positioned after the A > prompt, type

```
CHKDSK
```

and press ↵. DOS uses the extended search path—in this case, the root directory of the hard disk in drive C—to locate and execute the CHKDSK program.

Just as the PATH command can be used to set an extended search path, it can also be used to remove an extended search path, that is, to set the extended search path to null (no path). With the cursor positioned after the A > prompt, type

```
PATH;
```

and press ↵. When the PATH command is followed by only a semicolon (;), it instructs DOS to reset the search path to no path.

Keystroke Summary: Displaying the Extended Search Path

Type: `PATH`

Press: ↵

Keystroke Summary: Setting the Extended Search Path

Type: `PATH` *`pathspec`* where *pathspec* specifies a valid path specification

Press: ↵

Keystroke Summary: Resetting the Extended Search Path to No Path

Type: `PATH;`

Press: ↵

Deleting (Removing) Subdirectories (RMDIR, RD)

Now that you have learned how to create and work with subdirectories, you need to know how to delete (remove) subdirectories from your directory structure. The **RMDIR** (ReMove DIRectory) command is an internal DOS command that instructs DOS to remove (delete) a subdirectory entry from its parent directory. Note that a subdirectory must be empty (except for the period (.) and double-period (..) entries) before it can be removed from its parent directory. Also note that you cannot remove the current directory or the root directory of a disk. **RD** is an abbreviation of the RMDIR command. Since RD requires fewer keystrokes than RMDIR, RD is used in all examples in this chapter.

Assume that the directory structure in Fig. 8.1 is on the hard disk in drive C and that the current directory of the hard disk is the root directory. The following examples illustrate how to use the RMDIR (RD) command by using it to remove (delete) the entire directory structure of Fig. 8.1 from your hard disk. Before a subdirectory can be removed from the directory structure of a disk, all file and subdirectory entries must be deleted from that subdirectory. With the cursor positioned after the C> prompt, type

```
DEL SD-1\*.*
```

and press ↵. The DEL command instructs DOS to delete all file entries from subdirectory SD-1 on the disk in the default drive (drive C). After the files are deleted, subdirectory SD-1 contains only the period (.) and double-period (..) entries. Thus, subdirectory SD-1 can now be removed from its parent directory. With the cursor positioned after the C> prompt, type

```
RD SD-1
```

and press ↵. The RD command instructs DOS to remove subdirectory SD-1 from its parent directory—in this case, the root directory of the hard disk in drive C.

With the cursor positioned after the C> prompt, type

```
CD SD-2
```

and press ↵. The CD command changes the current directory of the disk in the default drive (drive C) to subdirectory SD-2. With the cursor positioned after the C> prompt, type

```
DEL SD-21
```

and press ↵. This DEL command is equivalent to the command DEL SD-21*.* and instructs DOS to delete all file entries from subdirectory SD-21 on the disk in the default drive (drive C). With the cursor positioned after the C> prompt, type

```
RD SD-21
```

and press ↵. The RD command instructs DOS to remove subdirectory SD-21 from its parent directory—in this case, subdirectory SD-2 on the hard disk in drive C.

With the cursor positioned after the C> prompt, type

```
DEL SD-22
```

and press ↵. The DEL command instructs DOS to delete all file entries from subdirectory SD-22 on the disk in the default drive (drive C). With the cursor positioned after the C> prompt, type

```
RD SD-22
```

and press ↵. The RD command instructs DOS to remove subdirectory SD-22 from its parent directory—in this case, subdirectory SD-2 on the hard disk in drive C. With the cursor positioned after the C> prompt, type

```
DEL *.*
```

and press ↵. The DEL command instructs DOS to delete all file entries from the current directory of the default drive—in this case, subdirectory SD-2 on the hard disk in drive C. Subdirectory SD-2 is now ready to be removed from the directory structure.

The current directory needs to be changed to the parent directory of the current directory—in this case, the root directory—since the current directory cannot be removed from the directory structure of a disk. With the cursor positioned after the C> prompt, type

```
CD ..
```

and press ↵. The CD command changes the current directory of the default drive to the parent directory—in this case, the root directory of the hard disk in drive C. With the cursor positioned after the C> prompt, type

```
RD SD-2
```

and press ↵. The RD command instructs DOS to remove subdirectory SD-2 from its parent directory—in this case, the root directory of the hard disk in drive C.

At this point the directory structure in Fig. 8.1 on the hard disk in drive C should be deleted, and your disk should be back to the original structure that it had when you began this chapter.

Keystroke Summary: Removing a Subdirectory from the Current Directory

Set: default drive to drive C with *subdirname* a subdirectory of the current directory

Type: `DEL` *subdirname*

Press: ↵

Type: `RD` *subdirname*

Press: ↵

Summary

Topic or Feature	Command or Reference	Page
Create (make) a subdirectory	`MKDIR` *subdirspec*	116
	`MD` *subdirspec*	116
Change the current directory	`CHDIR` *pathspec*	120
	`CD` *pathspec*	120
Change the current directory to the root directory	`CD \`	121
Display the current directory	`CD`	121
Change the current directory to the parent directory	`CD ..`	133
Change the normal DOS prompt to a customized DOS prompt	`PROMPT $P$G`	125
Reset the DOS prompt to the normal DOS prompt	`PROMPT`	125
Display the directory structure of a disk	`TREE`	128

Display the directory structure of a disk along with file and directory names	`TREE /F`
	`CHKDSK /V`
		128
		128

Display the extended search path	`PATH`	129
Set the extended search path	`PATH` *pathspec*	130
Reset the extended search path to no path	`PATH;`	131
Remove (delete) a subdirectory	`RMDIR` *subdirspec*	132
	`RD` *subdirspec*	132

Self-Test

1. The ________________ directory is created by DOS each time the FORMAT command is used to format a disk.
2. ________________ are used to divide a disk into separate storage areas that can be used like separate disks.
3. Subdirectory names follow the same format as ________________ names.
4. The ________________ of a disk is the directory that you are currently working in or the directory that you were working in on another drive.
5. A ________________ is the route you must follow to trace your way from the disk's root directory to a particular place in the directory structure.

6. State two ways in which a backslash (\) is used in a path name.

    ____

7. If Q is a subdirectory of P, then P is called the ______________ directory of Q.

8. The ______________ command is used to create subdirectories on a disk. Is this command an internal or external DOS command? ______________

9. The ______________ command is used to change the current directory of a disk. Is this command an internal or external DOS command?

    ____

10. State two DOS commands that can be used to display the path specification for the current directory of a disk. ______________

11. State the complete DOS command that changes the current directory to the root directory. ______________ State the complete DOS command that changes the current directory to its parent directory. ______________

12. The ______________ command can be used to change the normal DOS prompt to a customized DOS prompt. Is this command an internal or external DOS command? ______________

13. State the complete DOS command that instructs DOS to display the DOS prompt as the current directory specification followed by the > symbol.

    ____

14. The ______________ command can be used to display information describing the directory structure of a disk. Is this command an internal or external DOS command? ______________ This command can be used with the ______________ parameter to display the directory structure of a disk along with all file names and directory names.

15. The ______________ command can be used to extend the search path that DOS uses to locate an external DOS command. Is this command an internal or external DOS command? ______________

16. State the complete DOS command that is used to display the extended search path. ______________ State the complete DOS command that is used to reset the extended search path to no path. ______________

17. The ______________ command is used to delete a subdirectory entry from its parent directory. Is this command an internal or external DOS command?

    ____

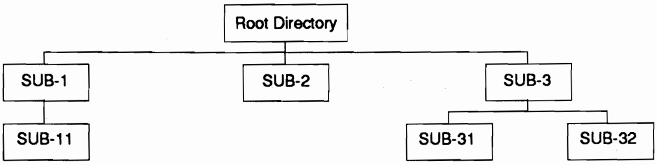
Exercises

In each exercise, choose the environment that exists on your personal computer: (1) your personal computer has a hard disk (drive C) that contains the DOS programs and a diskette drive (drive A) that contains a target diskette, or (2) your personal computer contains two diskette drives, drive A that contains the DOS System diskette and drive B that contains a target diskette.

1. Insert a blank diskette in the target diskette drive. Use an appropriate DOS command to perform each of the assigned tasks.

   a. Format the target diskette as a bootable diskette with volume label *namef*#8-1, where *name* denotes the first six characters of your last name or your entire last name (whichever is shorter) and *f* denotes the first character of your first name.

   b. Create the directory structure given in the following diagram on the target diskette:



   c. Display the directory structure for the target diskette on the screen.

   d. Use the output redirection operator to obtain a hard copy listing of the directory structure for the target diskette.

   e. Make the current directory of the target diskette the root directory.

   f. Copy all DOS files that have file names beginning with the character F into subdirectory SUB-2 on the target diskette.

   g. Make subdirectory SUB-32 the current directory of the target diskette.

   h. Change the DOS prompt to a customized DOS prompt that contains the current directory specification followed by the > symbol.

   i. Copy all DOS files that have file names beginning with the character C into subdirectory SUB-32 of the target diskette.

   j. Copy COMMAND.COM from the current directory (SUB-32) of the target diskette into subdirectory SUB-1.

   k. Make subdirectory SUB-11 the current directory of the target diskette.

   l. Copy CHKDSK.COM from subdirectory SUB-32 into the current directory (SUB-11) of the target diskette.

   m. Display the directory structure for the target diskette along with all file names and directory names on the screen.

   n. Use the output redirection operator to obtain a hard copy listing of the directory structure for the target diskette along with all file names and directory names.

   o. Delete subdirectories SUB-1, SUB-11, and SUB-31 from the directory structure of the target diskette.

   p. Repeat parts (m) and (n).

9

Customizing a System Using Batch Files

The material in this chapter introduces the DOS commands that allow you to:

- create and use batch files
- create and use the special batch file AUTOEXEC.BAT
- display comments on the screen from a batch file
- modify a batch file using the EDLIN command

What Is a Batch File?

This chapter introduces information that allows you to automate some of the processes presented in previous chapters. The automation procedure utilizes batch files. A **batch file** is a text file that contains DOS commands and has .BAT as the extension of the file name. When the name of a batch file is entered at command level, DOS executes the commands in the file. A **batch command** is the name of a batch file entered at command level. Note that you can break out of the execution of any batch command by entering the key combination Ctrl *and* Break or Ctrl *and* C, that is, by holding the Ctrl key down and pressing Break or holding the Ctrl key down and typing C.

The AUTOEXEC.BAT File

Each time you boot (start or restart) the system on your personal computer, the DOS command processor searches the root directory of the disk that is used in the boot process for a file named AUTOEXEC.BAT. The **AUTOEXEC.BAT** file is a special type of batch file that is automatically executed by DOS each time the system is booted. Thus, an AUTOEXEC.BAT file allows you to execute a fixed set of DOS commands every time you boot the system on your personal computer.

AUTOEXEC.BAT for an Application Program Diskette

The AUTOEXEC.BAT file can be used on an application diskette to help automate the execution of an application such as word processing. Assume that the three DOS system files are installed on your word processing system diskette so that it is a bootable diskette. When you boot the system with this diskette, you normally respond to date and time prompts before DOS displays the system prompt. At this point, you enter the name of the word processing program you want to execute—say, WP. The following steps show you how to create an AUTOEXEC.BAT file on this diskette that will boot the system, display a remark, request date and time information, display the version of DOS, and then execute the word processing program. In the following example the COPY CON command (see Chapter 7) is used to create the AUTOEXEC.BAT file.

Assume that the working copy of the word processing system diskette is in drive A. With the cursor positioned after the A> prompt, type

```
COPY CON AUTOEXEC.BAT
```

and press ↵. Type the following DOS commands on your keyboard, making sure that you press ↵ after each command is typed:

```
REM ***** AUTOEXEC.BAT is being executed *****
DATE
TIME
VER
WP
```

This process is terminated by entering an end-of-file marker. That is, hold the Ctrl key down and type Z (denoted by Ctrl *and* Z); then release the Ctrl key and press ↵. At this point, the COPY command copies the information entered from the keyboard and temporarily saved in memory to the target file specified in the COPY CON command. Thus, AUTOEXEC.BAT is copied to the current directory of the disk in the default drive—in this case, the root directory of the word processing system diskette in drive A.

The following process illustrates the effect of an AUTOEXEC.BAT file that resides in the root directory of the disk that is used to boot the system. Use the Ctrl *and* Alt *and* Del key combination to reboot (restart) the system in your personal computer. After the three DOS system files are loaded into memory and begin execution, DOS checks the root directory of the disk in the boot drive (drive A) for an AUTOEXEC.BAT file. Since the root directory of the word processing system diskette in drive A contains an AUTOEXEC.BAT file, DOS loads and executes the commands in this file.

AUTOEXEC.BAT for a Hard Disk

The AUTOEXEC.BAT file can be used on a hard disk to help automate certain commands, like the PATH and PROMPT commands, each time you boot the system. For the example presented in this section, the PATH command is used to set the extended search path to the subdirectory that contains the DOS program files—in this case, subdirectory DOS33 on the hard disk in drive C. The PROMPT command is used to change the standard DOS prompt to a customized DOS prompt that includes the current directory specification followed by the > character. Similar to the example in the previous section, the COPY CON command is used to create the AUTOEXEC.BAT file.

Assume that the current directory of the hard disk is the root directory. With the cursor positioned after the C> prompt, type

```
COPY CON AUTOEXEC.BAT
```

and press ↵. Type the following DOS commands on your keyboard, pressing ↵ after each command is typed:

```
REM ***** AUTOEXEC.BAT is being executed *****
DATE
TIME
VER
PATH C:\DOS33
PROMPT $P$G
```

Terminate the above process by using the key combination Ctrl *and* Z to create an end-of-file marker; then release the Ctrl key and press ↵. At this point the COPY command copies the information entered from the keyboard and temporarily saved in memory to the target file AUTOEXEC.BAT in the current directory of the default drive—in this case, the root directory of the hard disk in drive C.

The following process illustrates the effect of an AUTOEXEC.BAT file that resides in the root directory of the hard disk (drive C) when the hard disk is used to boot the system. Use the Ctrl *and* Alt *and* Del key combination to reboot (restart) the system in your personal computer. After the three DOS system files are loaded into memory and begin execution, DOS checks the root directory of the disk in the boot drive (drive C) for an AUTOEXEC.BAT file. Since the root directory of the hard disk in drive C contains an AUTOEXEC.BAT file, DOS loads and executes the commands in this file. After a remark is displayed, DOS prompts you for the current date and time by executing the DATE and TIME commands, and then displays the version of DOS. Next, DOS executes the PATH command that sets the extended search path to C:\DOS33, the subdirectory containing the DOS program files. Finally, DOS executes the PROMPT command that changes the standard DOS prompt to a customized DOS prompt that includes the current directory specification followed by the > character. At this point, control returns to command level with the C:\> prompt.

Keystroke Summary: Executing an AUTOEXEC.BAT File

Place: an AUTOEXEC.BAT file in the root directory of a disk used to boot the system

Boot: the system, and the AUTOEXEC.BAT file is automatically executed

Modifying a Batch File with EDLIN

There are times when you want to modify the contents of a batch file. One way to change a batch file is to use the COPY CON command and retype the entire batch file, including the changes. If the batch file contains more than a few short commands, retyping the file might not be the best way to modify its contents. DOS contains a utility program called **EDLIN** (EDit LINe or LINe EDitor) that allows you to create, edit (modify), and display text files. Refer to Appendix B for more information about the EDLIN command.

Keystroke Summary: Modifying a Text File Using the EDLIN Program

Insert: DOS System diskette in drive A, or assume that the DOS program files are on a hard disk

Insert: diskette containing the text file *filespec* in drive B

Set: default drive to drive A or to drive C with the current directory being the DOS files directory

Type: `EDLIN B:`*`filespec`*

Press: ↵

Type:
`L` to list the lines in the file *filespec*
n`I` to insert a text line at line number *n* (use the key combination Ctrl *and* Break or Ctrl *and* C to terminate the insert mode)
n`D` to delete text line number *n*
n to edit line *n*
`E` to exit EDLIN

Press: ↵

Summary

Topic or Feature	Command or Reference	Page
Create a batch file	COPY CON *filename*.BAT	138
Automatically execute a specified set of DOS commands during the boot process	Place AUTOEXEC.BAT file in the root directory of the boot disk	138
Execute the batch file *filename*.BAT	*filename*	137
Display remarks from within an executing batch file	REM *remark-line*	138
Edit (modify) a text file	EDLIN *filespec*	140
List lines in the file	L	140
Insert line *n*	*n*I	140
Terminate insert mode	Ctrl *and* Break or Ctrl *and* C	140
Delete line *n*	*n*D	140
Edit line *n*	*n*	140
Exit EDLIN	E	140

Self-Test

1. A ________________ is a text file that contains DOS commands and has .BAT as the extension of the filename.
2. The ________________ file is a special type of batch file that is automatically executed by DOS each time the system is booted. The file must reside in the ________________ directory of the boot disk to be executed.
3. The ________________ command can be used to create a batch file. When this command is used to create a file, the ________________ key sequence is used to assign an end-of-file marker to the file.
4. The ________________ command is used to display remarks on the screen from within an executing batch file. Is this command an internal or external DOS command? ________________
5. DOS contains a ________________program called ________________ that can be used to create, modify, and list text files.

Exercises

In each exercise, choose the environment that exists on your personal computer: (1) your personal computer has a hard disk (drive C) that contains the DOS programs and a diskette drive (drive A) that contains a target diskette, or (2) your personal computer contains two diskette drives, drive A that contains the DOS System diskette and drive B that contains a target diskette.

1. Insert a blank diskette in the target diskette drive. Use an appropriate DOS command to perform each of the assigned tasks.

a. Format the target diskette as a bootable diskette with volume label *namef*#9-1, where *name* denotes the first six characters of your last name or your entire last name (whichever is shorter) and *f* denotes the first character of your first name.

b. Copy the DOS files CHKDSK.COM and EDLIN.COM into the root directory of the target diskette.

c. Create the batch file AUTOEXEC.BAT in the root directory of the target diskette that contains the following commands in the stated order:

```
DATE
TIME
VER
PAUSE
CHKDSK /V
```

d. Display the contents of AUTOEXEC.BAT on the screen.

e. Insert the target diskette in drive A and use it to boot the system on your personal computer. Be sure to respond to the date and time prompts.

f. Use the output redirection operator to obtain a hard copy listing of the contents of AUTOEXEC.BAT.

g. Use the DOS line editor to insert the following remarks at the beginning of the batch file AUTOEXEC.BAT:

```
REM *** Batch file AUTOEXEC.BAT for Exercise 1, Chapter 9 ***
REM     NAME:  your-name
REM     COURSE:  course-number
REM     DATE:  today's-date
```

h. Display the contents of AUTOEXEC.BAT and AUTOEXEC.BAK on the screen.

i. Insert the target diskette in drive A and use it to boot the system on your personal computer. Be sure to respond to the date and time prompts.

j. Use the output redirection operator to obtain a hard copy listing of the contents of AUTOEXEC.BAT.

DOS Command Notation and Special Keys

TABLE A.1 NOTATION USED IN DESCRIBING DOS COMMANDS

Notation	Description
UPPERCASE LETTERS	Words in uppercase letters in DOS commands are called keywords. DOS command names are shown in uppercase letters and are keywords. In the DOS command format descriptions, uppercase letters and uppercase words may be entered on the command line in any combination of uppercase and lowercase letters.
lowercase italic letters	Command items shown in lowercase italic letters indicate that you are to substitute your information for these items.
[]	Items shown within square brackets indicate optional parameters for a command. If you decide to include the optional parameters, do not include the square brackets in the command.

{ }

Items shown within braces indicate that one and only one of the items inside the braces can appear in the command. The items are separated by a vertical bar. Do not include the braces in the command.

|

The vertical bar is used to separate items contained within braces or square brackets. One and only one of the items separated by the vertical bar can be used in the command.

¦

The split vertical bar indicates piping of output and input. This character is used between two DOS commands and instructs DOS to pipe (send) the output from the first (left) command to the second command as input.

<

The "less than" symbol is the standard input redirection operator. Input data for the command specified to the left of the symbol is received from the file or device specified to the right of the symbol. The format is *command < filespec*.

>

The "greater than" symbol is the standard output redirection operator. The output information generated by the command specified to the left of the symbol is sent to the file or device specified to the right of the symbol. The format is *command > filespec*.

...

Ellipsis points (repeat indicators) show that the items following the first unmatched left bracket can be repeated any number of times. Do not include the ellipsis points in the command.

Other Symbols

Any other symbols or number on the command line must be typed exactly as shown.

TABLE A.2 TERMS USED IN DESCRIBING DOS COMMANDS

Term	Description
command	A DOS command.
command line	A command line consists of [*pathspec*] *command* [*parameters*] where *pathspec* specifies the drive and the pathname in which *command* is located, *command* is a DOS command, and *parameters* denotes any additional information appearing on the command line.

parameter

A parameter is any item appearing on a command line after the DOS command name. Parameters reside in fields that are separated by command line delimiters such as spaces, commas, and slashes.

d:
d1: d2: d3:

d: is a drive designator. When a command contains multiple drive specifications within its parameter list, *d1:*, *d2:*, and *d3:* may be used as drive designators to aid in the description of the command.

filename

filename denotes the first part of the name of a file and consists of at least one and up to eight of the following characters:

A-Z a-z 0-9 $ & # @ ! % - _ ` () {}

DOS does not distinguish between uppercase and lowercase characters in a filename.

.ext

.ext denotes a filename extension or the second part of the name of a file and consists of a period followed by one to three valid *filename* characters. Filename extensions are optional.

file name

A file name consists of *filename*[*.ext*].

wildcard
(global) file name
characters

The characters * and ? are called wildcard (global) file name characters. * can be used to represent zero or more characters at the end of a filename or filename extension, and ? can be used to represent a single character in a filename or filename extension. The wildcard characters can be used in file names in many DOS commands.

filespec
filespec1
filespec2
filespec3

filespec denotes the file specification [*d:*][*pathname*]*filename*[*.ext*], which is equivalent to [*pathspec*]*filename*[*.ext*]. In a few cases, to simplify notation, *filespec* may be used to denote [*d:*][*pathname*][*filename*][*.ext*]. When a command contains multiple file specifications within its parameter list, *filespec1, filespec2,* and *filespec3* may be used as file specifications to aid in the description of the command.

subdirname

subdirname denotes a subdirectory name on a disk. *subdirname* has the same format and valid character set as a file name.

subdirspec

subdirspec denotes the subdirectory specification [*d:*]*subdirname*.

directory path — A directory path or path is the route you must follow to trace your way from the disk's main or root directory to a particular place in the directory structure.

pathname — `pathname` denotes the directory search path `[\][subdirname][\subdirname]...]`. It specifies a series of subdirectory names separated by backslashes (\). A `pathname` beginning with a backslash states that the directory search path begins with the root directory. A `pathname` beginning with a double period (..) states that the directory search path begins with the parent directory of the current directory.

pathspec — `pathspec` denotes the path specification `[d:]pathname`.

TABLE A.3 SPECIAL DOS CONTROL KEYS

Control Key	Description of Action
Ctrl *and* Alt *and* Del	Instructs DOS to start a warm system boot (restart or reset DOS).
Ctrl *and* Break Ctrl *and* C	Instructs DOS to cancel whatever the system or a program is doing and return you to command level.
Ctrl *and* Num Lock	Instructs DOS to suspend (halt) whatever the system is doing until you press another key.
Ctrl *and* PrtSc Ctrl *and* P	Acts as a printer echo switch. Pressing Ctrl *and* PrtSc (Ctrl *and* P) the first time instructs DOS to turn the printer echo mode on. That is, DOS starts printing every line at the same time as it is being displayed on the screen. Pressing Ctrl *and* PrtSc (Ctrl *and* P) a second time instructs DOS to turn the printer echo mode off.
Ctrl *and* S Pause	Instructs DOS to suspend output to the display unit (screen).
Ctrl *and* Z F6	Ctrl *and* Z generates an end-of-file marker. This can also be accomplished by pressing the F6 function key.
⇧ *and* PrtSc	Instructs DOS to copy the information that is currently displayed on the screen to the printer.

Note: When you type the characters for a DOS command, they are stored in a temporary storage area called the **command line buffer** or **keyboard buffer**. DOS always displays the contents of the command line (keyboard) buffer. The system processes the command line buffer after you press the ↵ key. Pressing the ↵ key initiates three tasks.

1. The contents of the command line buffer are sent to COMMAND.COM for interpretation and execution.
2. The command line buffer is copied into another temporary storage area called the **template**.
3. The command line buffer is cleared.

The last DOS command you entered is usually saved in the template, but not all commands are saved in the template.

TABLE A.4 DOS EDITING AND FUNCTION KEYS

Editing Key	Editing Key Description
F1 →	Copies the next character from the template to the command line buffer.
F2 *c*	Copies all characters up to (not including) the specified character *c* from the template to the command line buffer.
F3	Copies all remaining characters in the template to the command line buffer.
F4 *c*	Skips over (does not copy) the characters in the template up to (not including) the specified character *c*.
F5	Copies the command line buffer to the template and clears the command line buffer.
← Backspace	Erases (deletes) the previous character from the command line buffer without changing the template.
Esc	Clears the command line buffer without changing the template.
Ins	Puts DOS in insert mode. This allows you to enter characters from the keyboard without changing the current reference position in the template.
Del	Skips over (does not copy) a character in the template.
NumLock	Activates the numeric key pad mode or activates the arrow key/special function key mode. It is used to toggle between these two modes.

B
DOS Version 3.30 Commands

INTERNAL VERSUS EXTERNAL DOS COMMANDS

DOS commands are separated into two fundamental categories, internal DOS commands and external DOS commands. DOS commands that are resident in main memory after you boot your system are referred to as **internal DOS commands.** All other DOS commands are categorized as **external DOS commands.**

This means that any internal DOS command is always available for use at command level. When an external DOS command is used, the system routine associated with that command does not reside in main memory and must be available to the system on some external media such as a diskette or a hard disk. Consequently, when you use an external DOS command, you must make sure that the appropriate path specification is given so that this command can be referenced, loaded into memory, and executed by the system. This may be accomplished by having the DOS System diskette in the default disk drive.

Table B.1 gives a complete list of the DOS Version 3.30 commands. It also indicates the following:

1. DOS 3.xx commands that are the same as DOS 2.xx commands.
2. DOS 3.xx commands that contain modifications of the corresponding DOS 2.xx commands.

3. DOS 3.xx commands that are not available in DOS 2.xx.
4. DOS 3.30 commands that contain modifications of the corresponding commands in previous versions of DOS.
5. DOS 3.30 commands that are not available in previous versions of DOS.

Note that the commands are separated into three categories: internal DOS commands, internal DOS batch processing subcommands, and external DOS commands.

TABLE B.1 DOS COMMANDS BY NAME

Notation Used in the Table: Commands marked V3 are not available in versions of DOS prior to Version 3. Commands marked V3m are available in versions of DOS prior to Version 3, but have been modified in DOS Version 3. Commands marked V3.2 (V3.3) are new commands available in DOS Version 3.20 (3.30) that are not available in previous versions of DOS. Finally, commands marked with an asterisk (*) have been modified in DOS Version 3.30. Commands that are not marked have remained fundamentally unchanged through all versions of DOS. The internal batch commands have been listed separately, owing to differences in their use. A bullet (•) is used to denote commands that are not included in this appendix. If you need to use any commands not included in this appendix, you should refer to *PC-DOS & MS-DOS: A Ready Reference Manual* by Craig A. Wood or to your DOS reference manual.

Internal DOS Commands

	•	BREAK		MKDIR (MD)
V3.3	•	CHCP		PATH
		CHDIR (CD)		PROMPT
		CLS		RENAME (REN)
		COPY		RMDIR (RD)
	•	CTTY	•	SET
V3m	*	DATE	*	TIME
		DEL (ERASE)		TYPE
		DIR		VER
		ERASE (DEL)	•	VERIFY
	•	EXIT		VOL

Internal DOS Commands (Batch Processing Subcommands)

V3.3	•	CALL	•	IF
	*	ECHO		PAUSE
	•	FOR %%		REM
	•	GOTO	•	SHIFT

External DOS Commands

V3.3	•	APPEND (also internal)
	•	ASSIGN
V3	*	ATTRIB
V3m	*•	BACKUP
V3m	*	Batch (.BAT)
		CHKDSK
V3	•	COMMAND
		COMP
V3m		DISKCOMP
V3m		DISKCOPY
		EDLIN (DOS Utility)
	•	EXE2BIN (Not in DOS 3.30)
V3.3	•	FASTOPEN
V3m	*•	FDISK
	•	FIND
V3m	*	FORMAT
V3	*•	GRAFTABL
V3m	*•	GRAPHICS
V3	•	JOIN
V3	*•	KEYB
V3		LABEL
V3m	*•	MODE
		MORE
V3.3	•	NLSFUNC
V3m		PRINT
	•	RECOVER
V3.2	•	REPLACE
V3m	*•	RESTORE
V3	•	SELECT
V3	•	SHARE
		SORT
	•	SUBST
		SYS
		TREE
V3.2		XCOPY

NOTATION USED FOR COMMAND FORMATS

The way that a command is presented in Appendix B is not the way that you would enter the command at the keyboard. Lowercase italic letters and words that appear in the format of a command description represent information that must be supplied by you when you use that command. For example, consider the following command format for the RENAME command:

RENAME *filespec filename*[*.ext*]

When you use the RENAME command, you must specify the file to be renamed (denoted by *filespec*) and the new file name (denoted by *filename*[*.ext*]), which has an optional filename extension. The following example illustrates an explicit use of the RENAME command to rename a file:

```
RENAME A:FORMAT.COM FRMT.COM
```

You should refer to Appendix A for a complete listing of the notation used in Appendix B.

ATTRIB (SET/RESET/DISPLAY READ-ONLY ATTRIBUTE AND ARCHIVE BIT)

Format ATTRIB [+R | -R][+A | -A] *filespec*[/S]

Type External

Summary ATTRIB allows you to set the read-only file attribute on or off, to set the archive bit on or off, and to display the current settings of these attributes. The ATTRIB command operates upon a file or set of files specified by *filespec*.

Remarks

- +R instructs ATTRIB to make *filespec* read-only (cannot be modified or deleted).
- -R instructs ATTRIB to remove the read-only attribute of *filespec* (can be modified or deleted).
- +A instructs ATTRIB to set the archive bit on for *filespec*.
- -A instructs ATTRIB to set the archive bit off for *filespec*.
- /S instructs ATTRIB to process all files in the specified directory and its subdirectories. (New feature of DOS 3.30)
- The commands BACKUP /M and XCOPY /M copy a file only if its archive bit is set on.
- When you create a new file, its read-only attribute is set off and the archive attribute bit is set on.
- When *filespec* is the only specified parameter, ATTRIB displays the read-only status (R) and the archive bit status (A) for each file included in *filespec*.

Example The following command sets the read-only attribute on and the archive attribute bit off for the file TEST.DOC:

```
ATTRIB +R -A TEST.DOC
```

Batch [.BAT] (CALL BATCH FILE FOR EXECUTION)

Format `[pathspec]filename[.BAT] [parameters]`

Type External

Summary A Batch command consists of a *filename* with extension .BAT which references a special file containing DOS commands. This special file is called a batch file. When the name of a batch file is used as a command, the DOS commands that are included in the file are executed in sequence.

Remarks

- There are eight subcommands that can be used in a batch file to control batch processing. They are CALL, ECHO, FOR, GOTO, IF, PAUSE, REM, and SHIFT.

- Each time you boot the system, DOS searches the root directory of the boot disk for a special file named AUTOEXEC.BAT. If a file named AUTOEXEC.BAT is in the root directory of the boot disk, it is automatically executed each time the system is booted.

CHDIR [CD] (CHANGE/DISPLAY CURRENT DIRECTORY)

Format `CHDIR [pathspec]` or
`CD [pathspec]`

Type Internal

Summary CHDIR (CD) instructs DOS to change the current directory to the specified directory given by *pathspec* or to display the pathspec of the current directory.

Remarks

- Each disk drive has a current directory associated with it. DOS keeps track of the current directory for each drive in the system.
- To change the current directory of a drive other than the current drive, you enter the CHDIR (CD) command with the drive designator and pathname of the desired new current directory for the specified drive.
- A leading backslash (\) in a pathname tells DOS to begin the path at the root directory.
- The pathname specification cannot be longer than 63 characters starting from the root directory.

CHKDSK (CHECK DISK)

Format `CHKDSK [d:][pathname][filename[.ext]][/F][/V]`

Type External

Summary CHKDSK analyzes the directories, files, and File Allocation Table (FAT) on the designated or default drive and displays a disk and memory status report. If you specify *filename* in the command, CHKDSK displays the number of noncontiguous areas occupied by the file or files.

Remarks

- /F specifies the Fix parameter and instructs CHKDSK to try to correct any errors that it finds in the directory or file allocation table.
- /V specifies the View parameter and instructs CHKDSK to display the pathspecs of directories and the filespecs of files that have been checked. This listing includes any hidden files that may be on the disk and the disk's volume label if it exists.

- The CHKDSK status report for a bootable disk with a volume label shows at least three hidden files: the volume label and the DOS system files IBMBIO.COM and IBMDOS.COM. These hidden files do not appear in normal directory searches. Some application programs also create hidden files.
- You can use *.* in place of *filename.ext* in the CHKDSK command to determine the extent of file fragmentation on a disk.
- You should run CHKDSK occassionally for each hard disk drive and frequently used diskettes. This helps to ensure the integrity of the file structure and checks for file fragmentation.

CLS (CLEAR SCREEN)

Format `CLS`

Type Internal

Summary CLS clears all displayed information from the screen except the system prompt, which appears in the upper left-hand corner of the screen.

COMP (COMPARE TWO SETS OF FILES)

Format `COMP [filespec1] [filespec2]`

Type External

Summary COMP allows you to compare the contents of two files or two sets of files and display their differences. The file(s) denoted by *filespec1* (primary file(s)) are compared against the file(s) denoted by *filespec2* (secondary file(s)).

Remarks

- The paths and the names of the files being compared are displayed as they are compared.
- Wildcard characters can be used in the file specifications. They allow you to specify groups of files to be compared instead of two files at a time.
- The comparison operation is not performed if the sizes of the two files being compared do not match.
- You can compare all the files in one directory with all the corresponding files in another directory.
- If *filespec1* and *filespec2* are not specified or if *filespec2* is not specified, COMP prompts you for this information.

- COMP will display error messages whenever information does not match.

COPY (COPY NAMED FILES)

Format

```
COPY [/A][/B]filespec[/A][/B] [filespec2]
     [/A][/B][/V]  or
COPY [/A][/B]filespec[/A][/B] [+filespec1
     [/A][/B]...] [filespec2][/A][/B][/V]  or
COPY CON {filespec | PRN}
```

Type Internal

Summary COPY instructs DOS to copy one or more specified files to the default or specified disk. COPY copies files only from the current or specified directory. *filespec* (and *filespec1*) specifies the source file(s) or source group of files to be copied. *filespec2* specifies the target file or target path. The COPY CON command instructs DOS to copy input from the keyboard console (CON) to *filespec* or the printer (PRN).

Remarks

- The parameters /A and /B apply to the *filespec* immediately preceding it and to all remaining filespecs in the command until another /A or /B is encountered.
- /A (with a source *filespec*) indicates that the file is to be treated as a text or ASCII file. A Ctrl *and* Z (1A hexadecimal) is interpreted as the end-of-file marker. The file is copied up to, but not including, the first end-of-file marker. Also, the file is not copied beyond the end-of-file marker.
- /A (with a target *filespec2*) instructs COPY to place a Ctrl *and* Z character (end-of-file marker) as the last character in the target file.
- /B (with a source *filespec*) indicates that the file is to be treated as a binary file. The entire file as specified in directory file size (including Ctrl *and* Z characters) is copied up to the physical end-of-file.
- /B (with a target *filespec2*) instructs COPY to not place an end-of-file marker (Ctrl *and* Z) at the end of the target file.
- /V instructs COPY to verify that the source and target files are identical, after copying.
- + indicates that COPY is to perform concatenation of files.
- When a read-only file is copied with the COPY command, the read-only status is not transferred to the target file.
- The COPY command can be used to combine two or more files into one file by adding the additional files to the end of the first file. Combining files is normally done in the text (or ASCII) mode.

- When the COPY CON command is used, an end-of-file marker (Ctrl *and* Z) must be entered to complete the copying process.

DATE (DISPLAY/SET CURRENT DATE)

Format DATE [*mm-dd-yy* | *dd-mm-yy* | *yy-mm-dd*]

Type Internal

Summary DATE instructs DOS to display and set the system date where *mm* specifies the month (1 through 12), *dd* specifies the day (1 through 31), and *yy* specifies the year (80 through 99 or 1980 through 1999).

Remarks

- If you enter DATE with no parameters, the current system date and a prompt message are displayed on the screen. At this point you may either accept the current date or enter a new date.
- The parts of the date may be separated with a dash (-), a slash (/), or a period (.).
- When a file is created or modified, the current system date and time are recorded in the directory with the file name.
- If you have a real-time clock/calendar on your system, the DATE command in DOS 3.30 changes both the system date and the clock/calendar date. The DATE command in DOS versions previous to DOS 3.30 will not change the clock/calendar date.

DEL (DELETE/ERASE NAMED FILES)

Format DEL *filespec*

Type Internal

Summary DEL instructs DOS to delete (erase) *filespec* from the directory of the disk.

Remarks

- The DEL and ERASE commands perform the same operations.
- The DEL command does not destroy or remove the information in *filespec*; it only marks the file name(s) in the disk's directory as a deleted (erased) file(s).
- DEL will not delete files that are marked as read-only files.
- Wildcard characters may be used in *filespec*. Before using DEL with wildcard characters in *filespec*, enter the command DIR *filespec* to obtain a list of the files that will be deleted.
- DEL should be used with extreme care after using ASSIGN, JOIN, or SUBST.

- DEL will not delete hidden files such as IBMBIO.COM and IBMDOS.COM.
- If the parameter of the DEL command is a path specification, DEL deletes all files in the specified directory.

DIR (DISPLAY DIRECTORY INFORMATION)

Format `DIR [filespec][/P][/W]`

Type Internal

Summary DIR instructs DOS to display all directory entries (file and subdirectory names) in the current or specified directory, or to display only the directory entries specified by *filespec*. Directory entries for hidden files (such as IBMBIO.COM and IBMDOS.COM) are not displayed by the DIR command.

Remarks

- The general information produced by the DIR command includes the volume label (if it exists), the number of directory entries displayed, and the amount of free space left on the disk (rounded up to the nearest 1024 bytes).
- The information produced for files includes filenames, extensions, file size in bytes, and the date and time that each file was created or last modified. The DIR command identifies subdirectories by displaying <DIR> after each subdirectory name.
- /P (Pause) instructs DIR to pause the listing of directory information when the display screen is full. Pressing any key continues the listing.
- /W (Wide display format) instructs DIR to display five filenames with extensions or subdirectory names on each line. The file size and date and subdirectory markers do not appear in this format.

DISKCOMP (COMPARE DISKETTES)

Format `DISKCOMP [d1: [d2:]][/1][/8]`

Type External

Summary DISKCOMP allows you to compare the contents of the diskette in drive *d1* (source drive) to the contents of the diskette in drive *d2* (target drive).

Remarks

- /1 instructs DISKCOMP that you want to compare only the first side of each diskette, regardless of the format of each diskette and the drive type.

- /8 instructs DISKCOMP that you want to compare only 8 sectors per track, regardless of the number of sectors per track on each diskette.
- The DISKCOMP command should be used to compare two entire diskettes on a track-by-track basis. The COMP command should be used to compare files to each other.
- If no side or track parameters are specified, DISKCOMP automatically determines the number of sides and sectors per track that can be read from the source diskette (diskette in drive *d1*). These characteristics are used in the comparison with the target diskette (diskette in drive *d2*).
- DISKCOMP does not recognize assigned drives (ASSIGN), joined drives (JOIN), substituted drives (SUBST), or networked drives.
- DISKCOMP can be used only with compatible diskette drives and diskettes.

DISKCOPY (COPY DISKETTES)

Format `DISKCOPY [d1: [d2:]][/1]`

Type External

Summary DISKCOPY allows you to copy the entire contents of the diskette in drive *d1* (source drive) to the diskette in drive *d2* (target drive). If necessary, DISKCOPY will format the target diskette while copying.

Remarks

- /1 instructs DISKCOPY to copy only the first side of the source diskette, regardless of the diskette or drive type.
- The DISKCOPY command should be used to copy an entire diskette. The COPY command should be used to copy individual files.
- If /1 is not specified, DISKCOPY automatically determines, from the source diskette, the number of sides and sectors per track to copy.
- The source and target drives cannot be VDISKS, joined drives (JOIN), substituted drives (SUBST), or networked drives.
- If DISKCOPY determines an error in reading the source diskette or writing to the target diskette, it displays the track and side where the error occured but continues copying. If such an error occurs, bad data may have been transferred, making the target diskette unusable.
- DISKCOMP can be used after DISKCOPY has finished copying to determine whether the two diskettes are identical.

ECHO (SET/DISPLAY ECHO STATUS IN BATCH PROCESSING)

Format `ECHO [ON | OFF | message]`

Type Internal, batch processing subcommand

Summary ECHO controls the screen display of DOS commands executed from a batch file. Messages generated from the execution of batched DOS commands are not affected by the ECHO command.

Remarks

- If the ECHO command is used without any parameters, the current ECHO setting (ON or OFF) is displayed.
- If a command line in a batch file begins with the @ character, that line is not displayed when it is executed in a batch file. (New feature of DOS 3.30)

EDLIN (CALL LINE EDITOR UTILITY)

Format `EDLIN filespec[/B]`

Type External (Utility program)

Summary The EDLIN command is used to invoke the line editor utility program. This program allows you to create, edit, and display text files.

Remarks

- EDLIN is normally considered to be a DOS utility program and not a DOS command. It is included here because of its importance in creating and editing files.
- /B instructs EDLIN to load *filespec* up to the physical end-of-file. You should use this parameter when you know that *filespec* contains embedded end-of-file markers (Ctrl *and* Z). If you do not use the /B parameter, EDLIN uses *filespec* up to the first end-of-file marker (Ctrl *and* Z).
- The EDLIN prompt is an asterisk (*).
- The current line denotes the location of the last change to a file and is denoted by an asterisk immediately following the line number.
- Several fundamental EDLIN commands are listed below where n, n1, n2, and n3 are used to specify line numbers. This is not a complete set of EDLIN commands.

`[n1],[n2],n3[,count]C` • Copies lines *n1* through *n2* to lines beginning with *n3*. *count* specifies the number of times the range is to be duplicated. *count* defaults to one.

[*n1*][,*n2*]D

- Deletes lines *n1* through *n2*.

n

- Allows you to edit line *n*.

E

- Ends EDLIN and saves the edited file as *filespec*. The original file that was specified in EDLIN is given a .BAK filename extension.

[*n*]I

- Inserts lines of text immediately before line *n*. When you are creating a new file, you must enter an I to insert text lines into the file. You can terminate the insert command by entering Ctrl *and* Break or Ctrl *and* C.

[*n1*][,*n2*]L

- List lines *n1* through *n2* on the display screen. If *n2* is omitted, lines *n1* to the end of the file are listed. When L is used with no line numbers, the entire file is listed.

[*n1*],[*n2*],*n3*M

- Moves lines *n1* through *n2* in front of line *n3*.

Q

- Quit (terminate) the EDLIN session without saving any changes.

ERASE (ERASE/DELETE NAMED FILES)

Format ERASE *filespec*

Type Internal

Summary ERASE instructs DOS to erase (delete) *filespec* from the directory of the disk.

Remarks

- The ERASE and DEL commands perform the same operations.
- The ERASE command does not destroy or remove the information in *filespec*; it only marks the file name(s) in the disk's directory as an erased (deleted) file(s).
- ERASE will not erase files that are marked as read-only files.
- Wildcard characters may be used in *filespec*. Before using ERASE with wildcard characters in *filespec*, enter the command DIR *filespec* to obtain a list of the files that will be erased.

- ERASE should be used with extreme care after using ASSIGN, JOIN, or SUBST.
- ERASE will not erase hidden files such as IBMBIO.COM and IBMDOS.COM.
- If the parameter of the ERASE command is a path specification, ERASE erases all files in the specified directory.

FORMAT (FORMAT A DISKETTE OR HARD DISK)

Format `FORMAT d:[/S][/1][/8][/V][/B][/4][/N:ss][/T:tt]`

Type External

Summary FORMAT prepares the disk in the specified drive *d* to accept information from DOS by addressing the usable area of the disk, by creating the the disk directory and the File Allocation Table (FAT), and by placing the boot record at the beginning of the disk (side 0, track 0, sector 1). FORMAT also analyzes the entire disk for any defective tracks, marking these tracks as reserved to prevent them from being allocated to a data file.

Remarks

- Please remember that when you format a disk, all information on the disk is destroyed!
- /S instructs FORMAT to copy the DOS system files (IBMBIO.COM, IBMDOS.COM, and COMMAND.COM) from the disk in the default drive to the newly formatted disk. Because IBMBIO.COM and IBMDOS.COM are hidden files, executing a DIR command on the newly formatted disk lists only COMMAND.COM. DOS marks IBMBIO.COM and IBMDOS.COM as read-only files when they are created using FORMAT /S. If you do not specify the /S parameter, no system files are copied onto the disk.
- /1 instructs FORMAT to format a 5.25 inch diskette for single-sided use. If this parameter is not specified, the disk drive type will determine the format.
- /8 instructs FORMAT to format the diskette in the specified drive (5.25 inch drives only) for 8 sectors per track. If this parameter is not specified, the FORMAT command will default to 9 or 15 sectors per track depending upon the disk drive type. Note that FORMAT always generates 9 or 15 physical sectors on each diskette track, but it tells DOS to use 8 sectors per track if you use the /8 parameter. The /V parameter cannot be used with the /8 parameter.

- /V instructs FORMAT to prompt you for a volume label after the disk has been formatted. The volume label is used for identification purposes only. The label can contain from 1 to 11 characters, using the same characters allowed for filenames. The LABEL command may be used to add or change a volume label.

- /B instructs FORMAT to format a diskette for 8 sectors per track and permanently reserve space on the diskette for the hidden system files IBMBIO.COM and IBMDOS.COM. It does not place these files on the diskette. This parameter allows you to create a diskette on which any version of DOS can be placed by using that version's SYS command. A diskette formatted without the /B option must have the hidden system files transferred to it before any other files are placed on the diskette. The /S and /V parameters cannot be used with the /B parameter.

- /4 instructs FORMAT to format a 360KB diskette in a 1.2MB diskette drive. You can format a single-sided diskette in a 1.2MB diskette drive by using both the /1 and /4 parameters.

- /N:*ss* instructs FORMAT to format *ss* sectors per track. (Valid only when used with 720KB/1.44MB and 1.2MB drives. New feature of DOS 3.30.)

- /T:*tt* instructs FORMAT to format *tt* tracks on the diskette. (Valid only when used with 720KB/1.44MB and 1.2MB drives. New feature of DOS 3.30.)

- If either /N or /T is specified, then they both must be specified. They are used to format a diskette to less than the maximum supported capacity of the diskette drive.

- FORMAT ignores any disk drive reassignments (ASSIGN).

- FORMAT should not be used with joined drives (JOIN) or substituted drives (SUBST).

- FORMAT does not work on networked drives.

- Refer to your DOS manual for a listing of which FORMAT parameters may be specified with the different disk drive types.

- Some of the parameters in the FORMAT command are used to format diskettes for different capacities. Table B.2 outlines some of these characterstics.

TABLE B.2 DISKETTE CAPACITY CHARACTERISTICS

Diskette Types	Diskette Size (Inches)	Sides	Tracks /Side	Sectors /Track	Bytes /Sector	Total Bytes	Total Bytes Available	Diskette Overhead (Bytes)
160KB	5.25	1	40	8	512	163,840	160,256	3,584
180KB	5.25	1	40	9	512	184,320	179,712	4,608
320KB	5.25	2	40	8	512	327,680	322,560	5,120
360KB	5.25	2	40	9	512	368,640	362,496	6,144
720KB	3.50	2	80	9	512	737,280	730,112	7,168
1.2MB	5.25	2	80	15	512	1,228,800	1,213,952	14,848
1.44MB	3.50	2	80	18	512	1,474,560	1,457,664	16,896
			/T	/N				

LABEL (ASSIGN/CHANGE VOLUME LABEL)

Format LABEL [*d*:][*label*]

Type External

Summary LABEL allows you to assign, change, or remove a volume label on a disk.

Remarks

- *d* specifies the drive for the disk that you want to label.
- *label* specifies the volume label that is being assigned. *label* can contain from 1 to 11 characters, using the same characters allowed for filenames plus a space. The volume label is used for identification purposes only.
- FORMAT /V can also be used to assign a volume label to a new disk.
- If you want to check or change the volume label on a disk, enter LABEL with the appropriate drive designator and follow the prompts that are displayed by LABEL.

MKDIR [MD] (MAKE A DIRECTORY)

Format MKDIR *pathspec* or

MD *pathspec*

Type Internal

Summary MKDIR or MD instructs DOS to create a directory on the specified or default drive with the path given in *pathspec*.

Remarks • Subdirectory names follow the same format rules as file names.

- The maximum length of any single path, from the root directory to its level, is 63 characters, including imbedded backslashes (\).

MORE (DISPLAY SCREENS OF DATA)

Format `MORE < filespec` or
`command ¦ MORE`

Type External

Summary MORE instructs DOS to display up to one full screen of information at a time. In the first format, MORE reads *filespec* and sends it to the standard output device (normally the display screen). In the second format, the pipe symbol (|) causes the output from the DOS `command` to be sent (piped) to the MORE command, and MORE sends this output to the standard output device.

Remarks • After each full screen of output, MORE pauses and displays the following message at the bottom of the screen:

- -*More* --

You can press any key to view the next full screen of information.

- If MORE is entered without any parameters, it echoes all lines entered back to the screen without executing them. To return to the DOS prompt, press the Ctrl *and* Break or Ctrl *and* C keys.

PATH (SET SEARCH DIRECTORY)

Format `PATH [;][pathspec1 [;pathspec2]...]`

Type Internal

Summary PATH instructs DOS to search *pathspec1*, *pathspec2*, ... for external commands and batch files (files having an extension of .COM, .EXE, or .BAT) that were not found while searching the current directory.

Remarks • When PATH is entered with no parameters, it instructs DOS to display the pathspecs you specified in your last PATH command.

- Entering PATH followed by a semicolon (;) resets the extended search path to null; that is, it deletes the extended search path. This is the default condition when you boot your system.

- The PATH command searches only for executable files, that is, files having an extension of .COM, .EXE, or .BAT. The APPEND command allows you to set a search path for nonexecutable files.

PAUSE (PAUSE DURING BATCH PROCESSING)

Format PAUSE [*message*]

Type Internal, batch processing subcommand

Summary PAUSE is used in batch processing to suspend execution. If the *message* part of the command is present, it is displayed.

Remarks

- The optional *message* can be up to 121 characters long. The piping character (¦) and the I/O (Input/Output) redirection characters (< and >) cannot be used in *message*.
- You can insert the PAUSE command at strategic points in a batch file to stop the system so that you can attend to other tasks, such as changing diskettes. Pressing any key continues the execution of the batch file. Entering Ctrl *and* Break or Ctrl *and* C terminates the batch processing.

Example The following PAUSE command will prompt you to insert the correct diskette when it is executed in a batch file:

```
PAUSE Insert the worksheet diskette in drive B.
```

PRINT (PRINT TEXT FILES)

Format

```
PRINT [/D:device] [/B:buffsize] [/U:busytick]
      [/M:maxtick] [/S:timeslice] [/Q:quesize]
      [/C] [/T] [/P] [filespec...]
```

Type External

Summary PRINT allows you to print a text file or group of text files specified by *filespec...* while you are executing other DOS commands. PRINT also allows you to display file names in the print queue.

Remarks

- /D:*device* specifies the print *device*. If this parameter is not specified, the default device PRN is assumed. Valid values for *device* are LPT1, LPT2, LPT3, PRN, COM1, COM2, COM3, and AUX.
- /B:*buffsize* specifies the size in bytes of the print buffer. The default *buffsize* is 512 bytes.
- /Q:*quesize* specifies the maximum number of files you can have in the print queue at the same time. The default *quesize* is 10, and the range is from 1 to 32.
- /C specifies cancel mode and allows you to selectively delete (cancel) files in the print queue. The file name preceding /C and all following file names will be removed from the print queue until /P is encountered or ↵ is pressed.

- /T specifies terminate mode and allows you to delete all files in the print queue.
- /P specifies print mode and allows you to add file names to the print queue until /C is encountered or ↵ is pressed.
- The parameters /D, /B, /Q, /S, /U, and /M can be specified only on the first execution of the PRINT command.
- When PRINT is entered with no parameters, it instructs DOS to display the file names currently in the print queue.
- PRINT prints text files in a legible format. If you use PRINT to print another type of file (nontext file), such as a binary file, the printed information may be unreadable owing to the presence of special character codes, control characters, and escape sequences (nonprintable characters) in the file.

PROMPT (SET/RESET DOS PROMPT)

Format `PROMPT [{$param | string}...]`

Type Internal

Summary PROMPT allows you to set a new DOS prompt as specified by *$param* and/or *string* or to reset the prompt to the normal DOS prompt.

Remarks

- *$param* specifies particular characters to be used by PROMPT for a new DOS prompt. The dollar sign ($) is a parameter delimeter and instructs DOS to interpret the meaning of the next character (denoted by *param*) according to the following table (this table is not complete):

param	**meaning**
G	The > character
L	The < character
N	The default drive letter
P	The current directory path of the default drive
Q	The = character

- When PROMPT is entered with no parameters, PROMPT resets the prompt to the normal DOS prompt.
- You can freely intermix *$param* and the prompt *string* parameters on the PROMPT command line.

REM (DISPLAY REMARK DURING BATCH PROCESSING)

Format `REM [`*`message`*`]`

Type Internal, batch processing subcommand

Summary REM is used to display remarks from within an executing batch file. If *message* is specified, REM instructs DOS to display *message* when executed in a batch file.

Remarks
- If ECHO is OFF, *message* is not displayed.
- The optional *message* can be up to 123 characters long. The piping character (¦) and the I/O (Input/Output) redirection characters (< and >) cannot be used in *message*.
- The REM command helps make batch files more readable.

Example If you want to display an informative message, insert the following REM commands in your batch file:

```
REM  Processing is complete.
REM  Insert a blank diskette in drive B.
```

RENAME [REN] (RENAME FILES)

Format `RENAME `*`filespec filename`*`[.`*`ext`*`]` or
`REN `*`filespec filename`*`[.`*`ext`*`]`

Type Internal

Summary RENAME allows you to rename a file or group of files within the same directory. *filespec* specifies the file or files to be renamed. *filename*[*.ext*] specifies the new file name or names.

Remarks
- You can use wildcard characters to rename groups of files as long as you keep a one-to-one correspondence between the two specified parameters.

RMDIR [RD] (REMOVE DIRECTORY)

Format `RMDIR `*`pathspec`* or
`RD `*`pathspec`*

Type Internal

Summary RMDIR (RD) instructs DOS to remove (delete) the directory specified by *pathspec*.

Remarks

- The directory denoted by *pathspec* must be empty before it can be removed from the disk.
- You cannot remove the current directory or the root directory.

SORT (SORT TEXT INFORMATION)

Format `SORT [/R][/+n][<filespec1][>filespec2]` or
`command ¦ SORT [/R][/+n]`

Type External

Summary SORT allows you to sort text information alphanumerically by its binary coded value. In the first format, SORT reads data from *filespec1* (input), sorts the data, and then sends the sorted data to *filespec2* (output). In the second format, SORT accepts input from *command*, sorts the data, and then sends the sorted data to the display unit.

Remarks

- /R instructs SORT to sort the data in reverse order.
- /+*n* instructs SORT to start the sorting processing in column *n*.
- *filespec1* specifies the input file. If it is not specified, the default is the console.
- *filespec2* specifies the output file. If it is not specified, the default is the display unit.
- *command* specifies any DOS command and its output is piped (¦) as input to SORT.
- When the piping feature is utilized, DOS generates temporary files on the disk in the default drive to store the input and output data being piped. Consequently, the disk in the default drive cannot be write protected when you use piping in a DOS command sequence.

SYS (TRANSFER SYSTEM FILES)

Format `SYS d:`

Type External

Summary SYS instructs DOS to transfer the DOS system files IBMBIO.COM and IBMDOS.COM from the disk in the default drive to the disk in drive *d*.

Remarks

- The destination disk in drive *d* must be formatted and must have an empty directory or must have been formatted with the /S or /B parameters. (See the FORMAT command.) This is necessary because DOS startup requires that the hidden files IBMBIO.COM and IBMDOS.COM occupy the first two directory entries and because IBMBIO.COM must start at the beginning of the disk's data area.
- After the DOS system files IBMBIO.COM and IBMDOS.COM are transferred, they are marked as read-only files.
- SYS cannot be used on a network drive.

TIME (DISPLAY/SET CURRENT TIME)

Format TIME [*hh*:*mm*:[*ss*[.*xx*]]]

Type Internal

Summary TIME instructs DOS to display and set the system time where *hh* specifies the hours (0 through 23), *mm* specifies minutes (0 through 59), *ss* specifies seconds (0 through 59), and *xx* specifies hundredths of a second (0 through 99).

Remarks

- If you enter TIME with no parameters, the current system time and a prompt message are displayed on the screen. At this point you may either accept the current time or enter a new time.
- The hour, minute, and second parts of the time may be separated by using a colon (:) or a period (.). A period is used to separate seconds from hundredths of a second.
- Any field in the TIME command that is not entered will be set to zero.
- When a file is created or modified, the current system date and time are recorded in the directory with the file name.
- If you have a real-time clock/calendar on your system, the TIME command in DOS 3.30 changes both the system time and the clock/calendar time. The TIME command in DOS versions previous to DOS 3.30 will not change the clock/calendar time.

TREE (DISPLAY DIRECTORY STRUCTURE)

Format TREE [*d*:][/F]

Type External

Summary TREE allows you to display the directory structure of the disk in drive *d* by displaying all of the directory paths that are on the disk in the specified drive. TREE also allows you to list all subdirectory and file names in the root directory and each subdirectory.

Remarks
- /F instructs TREE to list all file names that are in the root directory and each subdirectory of the disk in drive *d*.
- If *d* is not specified, the default drive is assumed.

TYPE (DISPLAY TEXT FILES)

Format `TYPE` *`filespec`*

Type Internal

Summary TYPE instructs DOS to display the contents of the file specified by *filespec* on the standard output device.

Remarks
- TYPE displays text files in a legible format. If you use TYPE to display another type of file (nontext file), such as a binary file, the displayed information may be unreadable, owing to the presence of special character codes, control characters, and escape sequences (nondisplayable characters) in the file.

VER (DISPLAY DOS VERSION NUMBER)

Format `VER`

Type Internal

Summary VER allows you to display your DOS version number on the standard output device.

Remarks
- The version number is of the form *X.xx*, where *X* specifies a single-digit major version number and *xx* specifies a two-digit minor revision level.

VOL (DISPLAY DISK VOLUME LABEL)

Format `VOL [`*`d`*`:]`

Type Internal

Summary VOL instructs DOS to display the volume identification label of the disk in the specified drive.

Remarks
- If drive *d* is not specified, the default drive is assumed.

XCOPY (COPY SELECTED GROUPS)

Format

```
XCOPY filespec1 [filespec2][/A]
      [/D:mm-dd-yy][/E][/M][/P][/S][/V][/W]
```

Type External

Summary XCOPY allows you to selectively copy files specified by *filespec1* (source) to a drive, file, or directory specified by *filespec2* (target). *filespec1* and *filespec2* may specify a drive, file, or directory path.

Remarks

- /A instructs XCOPY to copy those source files that have their archive attribute bit set to one (on). This option does not modify the archive attribute bit of the source file.
- /D:*mm-dd-yy* instructs XCOPY to copy source files that have creation or modification dates on or after the specified date, *mm-dd-yy.*
- /E instructs XCOPY to copy any subdirectory, even if it is empty. If /E is not specified, empty directories are not created. This parameter is useful when used with the /S parameter.
- /M instructs XCOPY to copy those source files that have their archive attribute bit set to one (on). This option sets off the archive attribute bit of the source file. The parameters /M and /A are incompatible.
- /P instructs XCOPY to prompt you with (Y/N)? before copying each file.
- /S instructs XCOPY to copy files in the source directory and in all subdirectories below the initial directory unless they are empty. If /S is not specified, XCOPY copies only from the specified source directory.
- /V instructs XCOPY to verify each file as it is written on the target disk.
- /W instructs XCOPY to wait before it starts copying files. This allows you to insert diskettes before XCOPY begins searching for source files.
- If no path is specified in *filespec1* (source), XCOPY begins searching for files to copy in the current directory of the source drive.
- If no file names are specified in *filespec1*, the default file name *.* is used.
- When a read-only file is copied with the XCOPY command, the read-only status is not transferred to the target file.

C

Making Copies of the DOS 3.30 Diskettes

The DOS Version 3.30 package contains two 360KB (5.25 inch) diskettes (the DOS Startup diskette and the DOS Operating diskette) and one 720KB (3.50 inch) diskette (the DOS Startup/Operating diskette). For personal computers with 360KB diskette drives, the DOS Startup diskette must be used to boot the system. It also contains a few external DOS commands that are used initially to set up a personal computer. The 360KB DOS Operating diskette contains the remaining external DOS commands, but it cannot be used to boot the system. A single 360KB diskette does not have enough room to hold the DOS system files (IBMBIO.COM, IBMDOS.COM, and COMMAND.COM) along with all of the external DOS 3.30 commands (programs). Steps are outlined in this appendix to show you how to create a bootable 360KB diskette that does contain most of the external DOS 3.30 commands. Before you work through these steps, however, you should make backup copies of your original DOS diskettes.

Making Backup Copies of the DOS Diskettes

As a general rule, you should never use the original copies of the DOS diskettes as working copies. You should make backup copies of the DOS diskettes, store the original diskettes in a safe place for future reference, and use the backup copies as your regular working diskettes. You can make backup copies by following the steps outlined below for a personal computer with two diskette drives.

1. Insert the DOS Startup diskette (5.25 inch) or the DOS Startup/Operating dskette (3.50 inch) in drive A and boot the system.
2. For 5.25 inch diskette drives remove the DOS Startup diskette from drive A, and insert the DOS Operating diskette in drive A.
3. With the cursor positioned after the A> prompt, type `DISKCOPY A: B:`
4. Press ↵ and wait for three short DISKCOPY messages to be displayed on the screen.
5. Insert a blank (target) diskette in drive B.
6. Press ↵. When the DISKCOPY command finishes copying the DOS diskette, it displays a message similar to the following on the screen:

   ```
   Copy another diskette (Y/N)?
   ```

7. Type `Y` to copy another diskette, or type `N` to terminate the DISKCOPY program.
8. After you copy each DOS diskette, remove the target diskette from drive B, and write protect and properly label the diskette. Remove the original DOS diskette from drive A and store it in a safe place. (*Note*: In this book the backup copy of the 3.50 inch DOS Startup/Operating diskette will be referred to as the 3.50 inch DOS System diskette.)

Making a 360KB DOS 3.30 System Diskette Using Two Diskette Drives

Once you have created working (backup) copies of the two 360KB DOS diskettes, you are ready to create a bootable 360KB diskette that contains most of the external DOS 3.30 commands. This diskette will be referred to as the DOS System diskette.

1. Insert the DOS Startup diskette in drive A and boot the system.
2. With the cursor positioned after the A> prompt, type `FORMAT B: /S`
3. Press ↵ and wait for the FORMAT message instructing you to insert a diskette to be displayed on the screen.

4. Insert a blank diskette in drive B.

5. Press ↵. When the formatting process is complete, FORMAT displays a message similar to the following:

   ```
   Format another (Y/N)?
   ```

6. Type N and press ↵. Control returns to the A> prompt.

7. With the cursor positioned after the A> prompt, type the following commands, and press ↵ after each command.

   ```
   COPY FASTOPEN.EXE B:
   COPY REPLACE.EXE B:
   COPY SYS.COM B:
   ```

8. Remove the DOS Startup diskette from drive A, and insert the DOS Operating diskette in drive A.

9. With the cursor positioned after the A> prompt, type the following DOS commands, and press ↵ after each command.

   ```
   COPY *.COM B:
   DEL B:BASIC*.COM
   DEL B:DEBUG.COM
   DEL B:RESTORE.COM
   COPY *.EXE B:
   ```

10. Remove the target diskette from drive B and label it “IBM DOS System, Version 3.30.” This 5.25 inch diskette is your DOS System diskette. FDISK, KEYS, MODE, NLSFUNC, RESTORE, and SELECT are the only DOS 3.30 commands that are not on your newly prepared DOS 3.30 System diskette.

D

Answers to Chapter Self-Tests

Chapter 1

1. keyboard
2. display screen; printer
3. hardware
4. program
5. software; application software and systems software
6. operating system
7. computer system or computer
8. command processor
9. disk operating system
10. IBMBIO.COM (or IO.SYS), IBMDOS.COM (or MSDOS.SYS), and COMMAND.COM; IBMBIO.COM (or IO.SYS) and IBMDOS.COM (or MSDOS.SYS)
11. internal DOS commands
12. external DOS commands
13. delimiter
14. parameter

Chapter 2

1. the boot process, booting the system, booting DOS, or starting DOS
2. turning the power switch from OFF to ON (or 0 to 1)
3. holding the Ctrl and Alt keys down and pressing Del (denoted by Ctrl *and* Alt *and* Del)
4. Hardware self-test program
5. Two hidden files (IBMBIO.COM and IBMDOS.COM) and COMMAND.COM
6. date; time
7. there is a formatted diskette in drive A
8. DATE; internal; mm-dd-yy
9. TIME; internal; hh:mm
10. B:; B>

11. CLS; internal; upper-left corner of the screen
12. VER; internal
13. Hold the ⇧ key down and press PrtSc (denoted by ⇧ *and* PrtSc).
14. Hold the Ctrl key down and press PrtSc, or hold the Ctrl key down and type P (denoted by Ctrl *and* PrtSc or Ctrl *and* P).
15. first removing your diskette(s) from appropriate diskette drive(s) and then turning the computer's power switch off

Chapter 3

1. formatted; FORMAT; external
2. tracks; sectors
3. Addresses the usable area of a disk using the side, track, and sector numbers; identifies and logs bad sectors of a disk; creates a directory for the disk; creates a file allocation table for the disk; places a boot record on the first sector of the disk.
4. Means that DOS can read from the diskette but cannot write to the diskette. For a 5.25 inch diskette, place a write-protect tab over the write-protect notch on the diskette jacket. For a 3.50 inch diskette, open the write-protect window.
5. three DOS system; IBMBIO.COM, IBMDOS.COM, COMMAND.COM
6. transfer the three DOS system files to a disk after it has been formatted
7. allow you to assign a volume label (name) to a disk after it has been formatted
8. yes
9. SYS; empty; external

Chapter 4

1. DIR; internal; volume label if it has one, file names (filenames and extensions) in the disk's directory, file size in bytes, creation or last modification date and time, free space on disk in bytes
2. use of the /P parameter in the DIR command and use of the Ctrl *and* S key combination or Pause key
3. display only file names in a wide-line format
4. a. DIR A:T*.* or DIR T:*.* or DIR T*

   b. DIR B:*.WK* or DIR B:*.WK?

   c. DIR A:*.

   d. DIR B:*.DOC or DIR *.DOC

   e. DIR A:CHAP.*
5. >; DIR B: > PRN
6. CHKDSK; external; analyzes the directory of the specified disk and reports any errors, displays a summary of the utilization of the specified disk, and displays a report on the amount of memory available in the computer; user; hidden; user
7. VOL; internal
8. LABEL; external

Chapter 5

1. COPY; internal
2. a. A>COPY A:CHKDSK.COM
         A:CHKDSK.BAK or
         COPY CHKDSK.COM
         CHKDSK.BAK or
         COPY CHKDSK.COM *.BAK
      B>COPY A:CHKDSK.COM
         A:CHKDSK.BAK or
         COPY A:CHKDSK.COM
         A:*.BAK

   b. A>COPY A:FORMAT.COM
         B:FORMAT.COM or
         COPY FORMAT.COM B:*.* or
         COPY FORMAT.COM B:
      B>COPY A:FORMAT.COM
         B:FORMAT.COM or
         COPY A:FORMAT.COM B:*.*
         or COPY A:FORMAT.COM B:
         or COPY A:FORMAT.COM
3. DEL (or ERASE); internal

4. a. A>DEL A:AUTO.BAT or
   DEL AUTO.BAT
   B>DEL A:AUTO.BAT
   b. A>DEL B:T*.DOC
   B>DEL B:T*.DOC or
   DEL T*.DOC
5. RENAME; internal
6. A>RENAME A:DATA.DOC TEXT.DOC or
   RENAME DATA.DOC TEXT.DOC
   B>RENAME A:DATA.DOC TEXT.DOC
7. double check the file specifications in the DEL command; write protect an entire diskette; before using a wildcard file name character in a DEL command, check the file names being affected by using the same specification in the DIR command; use the ATTRIB command to declare files as read-only files.
8. ATTRIB; external

Chapter 6

1. DISKCOPY; external; user; hidden
2. If the source diskette contains bad sectors, the same sectors on the target diskette will be marked as bad sectors. If the target diskette contains bad sectors, any information written to these sectors might not be usable. Fragmented files on the source diskette are duplicated as fragmented files on the target diskette.
3. COPY; internal
4. XCOPY; external; /W; /P
5. XCOPY; /D:mm-dd-yy
6. COMP; external

Chapter 7

1. output redirection; >;
   DIR B: > B:DIRINFO.TXT
2. COPY CON; internal
3. TYPE; internal
4. PRINT; external; PRINT is an external DOS command, so it must be available on a disk to be used; the PRINT command issues a page ejection after each text file is printed
5. COPY; TYPE;
   COPY TEXTFILE.TXT PRN;
   TYPE TEXTFILE.TXT > PRN

Chapter 8

1. root (main)
2. subdirectories
3. file
4. current directory
5. directory path (path)
6. used at the beginning of a path name to denote that the path starts in the root directory; used as a delimiter (separator) between two directory names and between a directory name and a file name.
7. parent
8. MKDIR or MD; internal
9. CHDIR or CD; internal
10. CD and DIR
11. CD \; CD ..
12. PROMPT; internal
13. PROMPT PG
14. TREE; external; /F
15. PATH; internal
16. PATH   PATH;
17. RMDIR or RD; internal

Chapter 9

1. batch file
2. AUTOEXEC.BAT; root
3. COPY CON; Ctrl *and* Z
4. REM; internal
5. utility; EDLIN

Index