EQUITY™

EPSON®

MS-DOS®4.01 Reference Manual

EPSON®

MS-DOS® 4.01
Reference Manual

Y18699100200

IMPORTANT NOTICE
DISCLAIMER OF WARRANTY

Microsoft Corporation and Seiko Epson Corporation make no warranties or representations, either express or implied, by or with respect to anything in this manual, and shall not be liable for any implied warranties of merchantability or fitness for a particular purpose or for any indirect, special or consequential damages. Some states do not allow the exclusion of incidental or consequential damages, so this exclusion may not apply to you.

TRADEMARKS

Epson is a registered trademark of Seiko Epson Corporation.

FX-286 is a trademark of Epson America, Inc.

Hercules is a registered trademark of Hercules Computer Technology Corporation.

IBM is a registered trademark of International Business Machines Corporation.

Intel is a registered trademark of Intel Corporation.

Lotus is a registered trademark of Lotus Development Corporation.

Microsoft and MS-DOS are registered trademarks of Microsoft Corporation.

Seiko Epson Corporation
Nagano, Japan

Microsoft
Redmond, WA 98073

Contents

Introduction 1
Version 4.01 Enhancements 1
Epson Menu Utilities 4
How to Use This Manual 5
Conventions Used in This Manual 6
Where to Get Help 7

1 **MS-DOS and Your Computer** 1-1
Loading MS-DOS 1-3
Setting the Date and Time 1-3
Changing the Default Drive 1-4
Resetting the Computer 1-5
Using Diskette Drives 1-6
Formatting Diskettes 1-9
Using the Keyboard With MS-DOS 1-12

2 **Managing Files and Directories** 2-1
Managing Files 2-3
Using Directories 2-7

3 **Using MS-DOS Commands** 3-1
MS-DOS Command Overview 3-3
Entering Commands 3-11

4 **Alphabetical List of MS-DOS Commands** 4-1
Introduction 4-3
APPEND 4-5
ASSIGN 4-10
ATTRIB 4-12
BACKUP 4-15
BREAK 4-20
CHCP 4-21
CHDIR 4-23
CHKDSK 4-25
CLS 4-28
COMMAND 4-29
COMP 4-32

COPY 4-35
CTTY 4-39
DATE 4-41
DEL 4-43
DIR 4-45
DISKCOMP 4-47
DISKCOPY 4-51
DOSSHELL 4-54
ERASE 4-55
EXIT 4-57
FASTOPEN 4-59
FC 4-62
FDISK 4-71
FIND 4-85
FORMAT 4-87
GRAFTABL 4-94
GRAPH24 4-96
GRAPHICS 4-97
HELP 4-100
JOIN 4-101
KEYB 4-104
LABEL 4-108
MEM 4-110
MKDIR 4-112
MODE 4-114
MORE 4-129
NLSFUNC 4-130
PATH 4-132
PRINT 4-134
PROMPT 4-137
RECOVER 4-140
RENAME 4-142
REPLACE 4-143
RESTORE 4-146
RMDIR 4-150
SELECT 4-151
SET 4-152
SHARE 4-154

SORT 4-156
SUBST 4-158
SYS 4-160
TIME 4-161
TREE 4-163
TYPE 4-165
VER 4-166
VERIFY 4-167
VOL 4-168
XCOPY 4-169

5 **Epson Menu Utilities** 5-1
Introduction 5-3
MENU 5-3
SETPRINT 5-45

6 **Configuring Your System** 6-1
Defining the CONFIG.SYS File 6-3
Allocating Memory 6-5
Opening Files 6-6
Identifying Block Devices 6-7
Configuration Constraints 6-7
BREAK 6-8
BUFFERS 6-9
COUNTRY 6-12
DEVICE 6-15
DRIVPARM 6-17
FCBS 6-19
FILES 6-21
INSTALL 6-22
LASTDRIVE 6-23
REM 6-24
SHELL 6-25
STACKS 6-27
SWITCHES 6-29
Device Drivers 6-30
ANSI.SYS 6-31
DISPLAY.SYS 6-32

DRIVER.SYS 6-34
EEMM286.SYS 6-37
EMM386.SYS 6-40
HIMEM.SYS 6-42
PRINTER.SYS 6-44
SMARTDRV.SYS 6-46
VDISK.SYS 6-48
XMAEM.SYS 6-51
XMA2EMS.SYS 6-52

7 Batch Processing Commands 7-1
Creating Batch Files 7-3
Using an AUTOEXEC.BAT File 7-5
Using Replaceable Parameters 7-6
Using Environment Variables 7-7
Using the Batch Processing Commands 7-9
CALL 7-10
ECHO 7-11
FOR 7-13
GOTO 7-15
IF 7-16
PAUSE 7-18
REM 7-19
SHIFT 7-20

8 Programming Tools 8-1
Introduction 8-3
DEBUG 8-4
EDLIN 8-29
EXE2BIN 8-48
LINK 8-50

9 Code Page Switching 9-1
Introduction 9-3
Using Code Page Switching 9-4
Setting Up Your System for Code Page Switching 9-7
Switching Between Code Pages 9-9

Appendixes

A Error Messages A-1
Introduction A-3
Error Message Listing A-4

B International Keyboard Layouts B-1
Introduction B-3
Belgium B-4
Canada B-4
Denmark B-5
France B-5
Germany B-6
Italy B-6
Latin America B-7
Netherlands B-7
Norway B-8
Portugal B-8
Spain B-9
Sweden/Finland B-9
Switzerland B-10
United Kingdom B-10
United States B-11
Generating Special Characters B-12
Entering Characters Using Alt Gr B-13

C Code Page Tables C-1
Introduction C-3
Code Page 437 (United States-English) C-4
Code Page 850 (Multilingual) C-5
Code Page 860 (Portuguese) C-6
Code Page 863 (Canada-French) C-7
Code Page 865 (Nordic-Norwegian and Danish) C-8

D ANSI Escape Sequences D-1
Introduction D-3
Cursor Functions D-4
Modes of Operation D-5
Keyboard Reassignment D-7

Index ..... Index 1

Introduction

Your Epson® computer comes with MS-DOS®, the disk operating system by Microsoft®. A disk operating system controls the computer's hardware and the software you are using, enabling you to create, store, and retrieve data on a diskette or hard disk.

Using MS-DOS, you can run the most powerful, sophisticated software on the market today. While running your application program, you're not even aware that MS-DOS is in the background, but it is handling all the communication between the different components of your system.

Version 4.01 of MS-DOS consists of the standard commands plus special menu utilities created by Epson.

This Reference Manual provides a complete description of the MS-DOS operating system and each MS-DOS command. In addition to this manual, MS-DOS 4.01 comes with the following documentation:

- MS-DOS 4.01 Installation Guide—Follow the instructions in this booklet to install MS-DOS 4.01 on your computer.
- MS-DOS 4.01 Shell User's Guide—See this booklet for a complete description of the new Shell feature.
- MS-DOS 4.01 Command Summary—Use this booklet when you need a quick reminder about an MS-DOS command.

Version 4.01 Enhancements

If you are familiar with version 3.3 of MS-DOS, you'll notice the following new features provided by version 4.01:

- The MS-DOS Shell, which lets you run programs and choose operating system commands from menus instead of the MS-DOS command prompt

- Support for disk partitions larger than 32MB
- New and enhanced commands.

The MS-DOS Shell is designed for both new and experienced users of MS-DOS. It lets you tailor your system to your own needs and manage your programs and data more effectively.

The support for large partitions means that hard disks with capacities greater than 32MB no longer have to be partitioned into two or more logical drives. You can still partition large disks into smaller units if you want, but this is no longer a requirement.

New Commands

You'll find these new commands in this version of MS-DOS:

DOSSHELL lets you access the MS-DOS Shell.

MEM provides information about the amount and type of memory installed in your system, as well as the amount of memory currently available for application programs and data.

INSTALL lets you invoke certain terminate-and-stay-resident commands (FASTOPEN, KEYB, NLSFUNC, and SHARE) from your CONFIG.SYS file.

SWITCHES instructs your system to use conventional keyboard functions only, ignoring enhanced keyboard functions. This configuration command is provided for compatibility with applications that do not recognize the enhanced keyboard.

Enhanced Commands and Device Drivers

The following commands and device drivers have been enhanced in version 4.01:

The ANSI.SYS device driver includes three new switches:

- /K prevents the use of extended functions when an enhanced keyboard is in use. This option is provided for compatibility with applications that do not recognize the enhanced keyboard.

- /L instructs the system to retain the number of screen rows specified in the last MODE command, even with an application program that normally resets the rows to 25.
- /X lets you redefine keys on the enhanced keyboard that have extended values.

APPEND includes a /PATH switch. Setting /PATH:OFF causes APPEND not to search appended paths when the original file specification includes a pathname.

BACKUP now automatically formats unformatted diskettes.

BUFFERS lets you specify up to eight lookahead buffers. If you specify lookahead buffers, MS-DOS reads not only the disk sector called for by the application program but one or more additional sectors as well. This may enhance the performance of some programs that perform sequential file operations.

CHKDSK reports the volume serial number of a disk, if any, as well as the size of the disk's allocation units (clusters). The volume serial number is assigned by the FORMAT command.

COUNTRY supports four additional countries: Japan, Korea, the People's Republic of China, and Taiwan.

DEL and ERASE include a /P switch, which instructs the system to display a confirmation prompt before deleting a file.

The DISPLAY.SYS driver can check your hardware to determine the type of video adapter in use, if the *type* parameter is omitted.

FASTOPEN provides a /X switch to store directory and FAT (File Allocation Table) information in expanded memory. You can also specify the number of directory/file entry and continuous space buffers.

FDISK presents partition size information in megabytes and percentage of total disk space, instead of cylinders. You can specify partition size information in the same way.

FORMAT lets you assign a volume label before formatting takes place. FORMAT also includes a /F switch to specify the capacity of the diskette you want to format and automatically generates a volume serial number on newly formatted diskettes.

GRAPHICS now supports the EGA and VGA display modes and offers enhanced support for IBM printers.

KEYB includes a /ID switch to specify a keyboard identification code. This is useful for countries, such as France, Italy, and the United Kingdom, that use more than one enhanced keyboard.

MODE lets you set the keyboard typematic delay time and repeat rate, as well as the number of video display lines (25, 43, or 50). There are also additional baud rate and parity options.

REPLACE includes a /U switch that instructs the system to copy only those files that have more recent dates on the source disk than on the target.

SELECT offers improved full-screen display.

TIME lets you use 12-hour time specifications under certain conditions.

TREE has an improved visual display.

Epson Menu Utilities

The MENU and SETPRINT utilities make it easier for you to run MS-DOS commands. Rather than typing long, complex command lines with multiple variables, you can just select the appropriate options from simple screen menus. See Chapter 5 for more information about these utilities.

How to Use This Manual

The first three chapters in this manual provide some basic information about using MS-DOS. If you have never used an operating system, these chapters will help you get started. The rest of this manual gives more detailed information about MS-DOS and its commands.

For more information, you might want to buy a book about MS-DOS, such as Van Wolverton's *Running MS-DOS*, (Redmond, Washington, Microsoft Press, 1988).

The contents of this manual are as follows:

Chapter 1 provides some general information on using MS-DOS with your computer.

Chapter 2 describes how to manage your files and organize them in directories.

Chapter 3 provides an overview of the MS-DOS commands and the procedure for entering commands on your keyboard.

Chapter 4 is an alphabetical listing of most of the MS-DOS commands. These are the commands you use most often.

Chapter 5 describes the menu utilities: MENU and SETPRINT.

Chapter 6 is an alphabetical listing of the commands you can use to configure your system if you want to change the default settings.

Chapter 7 is an alphabetical listing of the MS-DOS batch processing commands you can use to create batch files. Batch files let you group commands in a single file to automate frequently used procedures.

Chapter 8 is an alphabetical listing of commands used as programming tools.

Chapter 9 describes code page switching.

Appendix A lists and describes all MS-DOS 4.01 error messages.

Appendix B illustrates the international keyboard layouts supported by MS-DOS.

Appendix C illustrates the MS-DOS code pages.

Appendix D lists the escape sequences used by the ANSI.SYS device driver.

Conventions Used in This Manual

This manual uses certain symbols and typographical conventions:

- All keys and function keys appear in boldface type, such as **A**, **Esc**, and **Enter**.
- When you press two keys in combination, the names appear in bold separated by a space, as in **Ctrl Z**. This means hold down the **Ctrl** key and press **Z**.
- Special symbols indicate different types of command parameters. You do not type these symbols when you enter the command; they simply describe the valid command format.

  `[ ]` Square brackets indicate an optional entry.

  `...` Ellipses indicate that you may repeat a parameter as many times as needed or desired.

- Lowercase italic letters indicate a command parameter that varies:

  ```
  TYPE d:filename
  ```

  In the above command format, *d* and *filename* are parameters that vary. You can use any disk drive descriptor and any filename when entering this command. An example of this command is the following:

  ```
  TYPE A:AUTOEXEC.BAT
  ```

- When instructions are given to type a command, the command appears in uppercase letters like this: DISKCOPY, CHDIR, or MENU. However, it is not necessary to type the command in uppercase letters. You can type the command in uppercase or lowercase letters.

Where to Get Help

Customer support and service for Epson products is provided by a network of authorized Epson dealers and service centers throughout the United States. Epson America, Inc. provides product information and toll-free support to our dealers and service centers.

Therefore, we ask that you contact the business where you purchased your Epson product to request assistance. If they do not have the answer to your question, they can obtain it through our toll-free dealer support program.

We are confident that this policy will provide you with the assistance you need. If you need to find an Epson dealer or service center in your area, please call our toll-free number: 1-800-922-8911.

Chapter 1

MS-DOS and Your Computer

Loading MS-DOS 1-3
Setting the Date and Time 1-3
Changing the Default Drive 1-4
Resetting the Computer 1-5
Using Diskette Drives 1-6
  Drive and Diskette Incompatibilities 1-7
  Using Two Diskette Drives 1-7
  Using a Single Diskette Drive 1-8
Formatting Diskettes 1-9
  Formatting With Two Drives 1-9
  Formatting With One Drive 1-11
Using the Keyboard With MS-DOS 1-12
  Control Key Functions 1-12
  Function Keys 1-13
  Generating Special Characters 1-15
  Selecting International Keyboards 1-16

Loading MS-DOS

To use your computer, you must load MS-DOS and have it running in the computer's memory. Loading MS-DOS, also known as booting the system, lets you give instructions to the operating system in the form of MS-DOS commands.

If you have installed MS-DOS on a hard disk, follow these steps to load MS-DOS:

1. Turn on your monitor.
2. Make sure there is no diskette in drive A.
3. Turn on your computer.

If you do not have a hard disk, or if you have not installed MS-DOS on your hard disk, follow these steps to load MS-DOS:

1. Turn on your monitor.
2. Place your MS-DOS Startup diskette, or another diskette that has been formatted with the /S option (see Chapter 4), in drive A. Secure the diskette drive.
3. Turn on your computer.

When MS-DOS is loaded, you see the command prompt, which may be A> or C>. The command prompt tells you that MS-DOS is ready to use. See your User's Guide for a complete description of turning on your system and inserting diskettes.

Setting the Date and Time

Before you see the MS-DOS command prompt, you may be asked to enter or confirm the current date:

```
Current date is Mon Ø2-13-89
Enter new date (mm-dd-yy):
```

Press **Enter** to accept the date shown or type a new date and press **Enter**.

You may also be prompted to enter the current time:

```
Current time is 12:41:32.19p
Enter new time:
```

Press **Enter** to accept the time shown or type a new time and press **Enter**.

Changing the Default Drive

The symbol A> is the MS-DOS command prompt. It tells you that MS-DOS is operating from drive A and is ready to receive a command.

Whenever you start or reset your computer with an MS-DOS system diskette in drive A, MS-DOS automatically logs onto drive A and displays the A> prompt. If you have a hard disk and start your computer without a diskette in drive A, MS-DOS logs onto the C drive and displays the C> prompt.

The drive identified by the MS-DOS command prompt is called the *default*, or *current*, drive. It is the drive that MS-DOS is currently logged onto. If you type a command without specifying a drive letter, MS-DOS looks for that command on the default drive. For example, if MS-DOS is displaying the A> prompt and you type

```
CHKDSK
```

and press **Enter**, MS-DOS looks for the CHKDSK command on drive A.

To run a program or find a file that is not on the default drive, you may change the default drive. To do that, type the letter of the new default drive, followed by a colon. Then press **Enter**. For example, to change the default drive from A to B, type

```
B:
```

and press **Enter**.

Another way to run a program or find a file that is not on the default drive is to type the drive letter, followed by a colon, before the program name or filename. For example, if you are logged onto drive B and the file CHKDSK.COM is stored on the diskette in drive A, you can run CHKDSK by typing the following and pressing **Enter**:

```
A:CHKDSK
```

The combination of the drive letter and the colon character is sometimes called a *drive descriptor*.

Resetting the Computer

When you are running MS-DOS, you may want to reset your computer. This is sometimes necessary if you are using a program that stops in the middle of an operation. Or you may want to reset to start a program from the beginning.

You can perform three types, or levels, of resets. As you move from one level to the next, you cause more of the system to halt and start over. The three levels are: the MS-DOS BREAK, the operating system reboot, and the hardware level restart.

The first level, the MS-DOS BREAK, merely stops a program's operation and returns you to the command prompt. Any program you are running is stopped, and any data not stored on disk is lost. You can use the following command to interrupt most application programs: hold down the **Ctrl** key and press **Break**. (Not all programs can be terminated in this way.)

The next level is the operating system reboot. This stops the computer's CPU (central processing unit) and forces you to reload MS-DOS and start over. Any program you are running is stopped, and any data not stored on disk is lost. To reset at this level, hold down the **Ctrl** and **Alt** keys and press the **Del** or **Delete** key.

The final level is a hardware level restart. To do this, press the RESET button. This is the same as turning your computer's power switch OFF then ON again. Use the hardware level reset when no other reset can help with a software problem.

Using Diskette Drives

Your computer has one or more of the following types of drives:

- 360KB drive—With this drive, use 5¼-inch, double-sided, double-density, 360KB diskettes. (You can also use single-sided, 160KB or 180KB diskettes.) These diskettes contain 40 tracks per side, 8 or 9 sectors per track, and hold up to 360KB of information. (With 8 sectors per track, a diskette holds 320KB.) KB stands for kilobyte; each kilobyte equals 1024 bytes.
- 1.2MB drive—With this drive, use 5¼-inch, double-sided, high-density, 1.2MB diskettes. These diskettes contain 80 tracks per side, 15 sectors per track, and hold up to 1.2MB of information. MB stands for megabyte; each megabyte equals 1,048,576 bytes.
- 720KB drive—With this drive, use 3½-inch, double-sided, double-density, 720KB diskettes. These diskettes contain 80 tracks per side, 9 sectors per track, and hold up to 720KB of information.
- 1.44MB drive—With this drive, use 3½-inch, double-sided, high-density, 1.44MB diskettes. These diskettes contain 80 tracks per side, 18 sectors per track, and hold up to 1.44MB of information.

If your computer has more than one of these drive types, you need to be aware of certain incompatibilities between the drives and the diskettes they use. These incompatibilities also apply to the same type of diskette drives on other compatible computers.

Drive and Diskette Incompatibilities

Because of the difference in size, you cannot use 3½-inch diskettes in a 5¼-inch drive or vice versa. Additionally, there are certain limitations on using diskettes that are the same size as the drive but have different capacities. The following tables summarize the possibilities and limitations.

5¼-inch drive/diskette compatibility

Drive type	Diskette types it can read from and write to
360KB	160KB, 180KB, 320KB, or 360KB
1.2MB	160KB, 180KB, 320KB, 360KB, or 1.2MB

WARNING
If you write to a 360KB (or 160KB, 180KB, or 320KB) diskette while it is in a 1.2MB drive, you may not be able to read it or write to it in a 360KB drive later.

3½-inch drive/diskette compatibility

Drive type	Diskette types it can read from and write to
720KB	720KB
1.44MB	720KB, 1.44MB

If you have any combination of the above drives (360KB, 1.2MB, 720KB, and 1.44MB), you can copy files from one drive to another (using the COPY command) as long as the correct diskette type is in each drive. You can also copy files between a hard disk and any type of diskette. You cannot use the DISKCOPY command to copy from one drive to another if the two drives are not the same type.

Using Two Diskette Drives

Operating systems usually expect the computer to have at least two physical disk drives. Normally, MS-DOS recognizes drives A and B for two diskette drives, and C for the hard disk drive.

A typical way to use a two-drive system is to load MS-DOS from drive A, run your application software from the same drive, and use drive B to store your data files.

Note
You can load MS-DOS from an application software diskette if that diskette contains the operating sytem files, including COMMAND.COM. See Chapter 4 for information on the /S parameter of the FORMAT command and the SYS command. You use these commands to transfer the necessary system files to a diskette.

There are many advantages to having two drives. You can perform file copy operations without swapping diskettes. This saves a great deal of time and reduces the risk of damaging your data by accidentally inserting the wrong diskette for a copy or format operation. While you have an application program in one drive, you can log onto the other drive to access a file, perform diskette operations, or display a directory.

Using a Single Diskette Drive

If your system has only one diskette drive, you enter commands just as you would for a system with two diskette drives. The only difference is that all your diskette operations are performed on drive A.

For certain functions, such as copying diskettes, your single drive can act as two drives, drive A and drive B. Instead of A and B representing two physical drives, they represent two different diskettes in the same drive.

When you specify a file on a drive other than A, MS-DOS prompts you to insert the diskette for that drive. If you type `COPY LETTER B:` and press **Enter**, MS-DOS copies the file from the A diskette to the computer's memory and then the screen displays the following:

```
Insert diskette for drive B:
and press any key when ready
```

Whenever you need to change diskettes, this prompt reminds you which diskette to use. You simply remove your A diskette from the drive, insert your B diskette, and press any key to continue. MS-DOS copies the file from memory to the B diskette in the drive.

MS-DOS displays the letter of the default drive in the command prompt, not the letter of the last diskette you used. For example, if you just viewed the directory of the B diskette and your default drive is A, MS-DOS knows that the B diskette is still in the drive even though the command prompt is `A>`. If you now type `DIR` to view the directory of the A diskette, MS-DOS prompts you to:

```
Insert diskette for drive A:
and press any key when ready
```

Formatting Diskettes

Before you can store data on a new diskette, the diskette must be formatted. Formatting prepares the diskette so that MS-DOS can write to it. You need to do this only once, before you use the diskette for the first time.

You can also reformat previously used diskettes. Reformatting, however, erases all data on that diskette. Always be sure you do not want any data on a used diskette before you reformat it.

The formatting procedure you use depends on whether you have two diskette drives or one diskette drive. Follow the instructions below for your configuration.

Formatting With Two Drives

Follow this procedure to format a diskette in drive B. (If you want to format a diskette in a different drive, substitute the letter for that drive in the instructions below.)

1. Insert your working copy of the MS-DOS diskette that contains the FORMAT program in drive A (and press any key if necessary) or log onto drive C (and the appropriate directory) if you have stored the FORMAT program on the hard disk.

2. When you see the A> or C> prompt, type:

```
FORMAT B:
```

and press **Enter**. You see this prompt:

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

3. Insert the diskette you want to format into drive B, and press **Enter** to start formatting.

As the formatting takes place, MS-DOS displays a message telling you what percentage of the diskette has already been formatted.

4. When the diskette is completely formatted, you see the following message:

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

If you want, you can now assign a volume label to your diskette. Volume labels help you identify the contents of each diskette you use. A volume label consists of up to eleven characters. If you don't want to assign a volume label, just press **Enter**.

MS-DOS then displays messages about the capacity of the newly formatted diskette. For a 360KB diskette, the messages look like this:

```
362496 bytes total disk space
362496 bytes available on disk
1024 bytes in each allocation unit
354 allocation units available on disk

Volume Serial Number is xxxx-xxxx
```

Allocation units, also known as *clusters*, are the smallest unit of file storage on a hard disk or diskette. The size of the allocation unit varies, depending on the disk's format. Hard disks usually use larger allocation units than diskettes.

The volume serial number is an identification code assigned automatically to the diskette by the FORMAT command.

Following the capacity messages, you see the prompt:

```
Format another (Y/N)?
```

5. Press **Y** and **Enter** to format another diskette, or press **N** and **Enter** to return to the MS-DOS command prompt.

Formatting With One Drive

1. Insert your working copy of the MS-DOS diskette that contains the FORMAT program in drive A (and press any key if necessary) or log onto drive C (and the appropriate directory) if you have stored the FORMAT program on the hard disk.
2. When you see the `A>` or `C>` prompt, type:

   ```
   FORMAT A:
   ```

   and press **Enter**. You see this prompt:

   ```
   Insert new diskette for drive A:
   and press ENTER when ready
   ```

3. If the MS-DOS diskette is in drive A, remove it. Then insert the diskette you want to format in drive A and press **Enter** to start formatting. As the formatting takes place, MS-DOS displays a message telling you what percentage of the diskette has already been formatted.
4. When the diskette is completely formatted, you see the following message:

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

If you want, you can now assign a volume label to your diskette. Volume labels help you identify the contents of each diskette you use. A volume label consists of up to eleven characters. If you don't want to assign a volume label, just press **Enter**.

MS-DOS then displays messages about the capacity of the newly formatted diskette. For a 1.44MB diskette, the messages look like this:

```
1457664 bytes total disk space
1457664 bytes available on disk
512 bytes in each allocation unit
2847 allocation units available on disk

Volume Serial Number is xxxx-xxxx
```

Allocation units, also known as *clusters*, are the smallest unit of file storage on a hard disk or diskette. The size of the allocation unit varies, depending on the disk's format. Hard disks usually use larger allocation units than diskettes.

The volume serial number is an identification code assigned automatically to the diskette by the FORMAT command.

Following the capacity messages, you see the prompt:

```
Format another (Y/N)?
```

Using the Keyboard With MS-DOS

This section describes how to use the keyboard to take advantage of a variety of MS-DOS functions. (See your computer's User's Guide for a detailed description of the keyboard.)

Control Key Functions

The control key (**Ctrl**) performs a variety of functions when used with other keys. To activate these functions, you hold down the **Ctrl** key and then press the other key or keys. This manual shows these multiple key commands like this:

Ctrl C

The following table shows the control key functions available in MS-DOS.

Control key functions

Control Key Combination	Function
Ctrl C	Cancels the current command.
Ctrl H	Removes the last character from the command line and erases the character from the screen; same as pressing the backspace key.
Ctrl I	Inserts a tab in the command line; same as pressing the tab key.
Ctrl J	Inserts an end-of-line without emptying the command line; this allows you to enter a command on more than one line.
Ctrl M	Same as **Enter**.
Ctrl P	Echoes keyboard output to the printer; press **Ctrl P** again to turn off the echo.
Ctrl S	Pauses scrolling of the output display on the screen; press any key to resume.
Ctrl Z	Places an end-of-file character in a text file.
Ctrl Break	Halts command execution; same as **Ctrl C**.
Ctrl Alt Del	Performs a system reset; any data in memory is lost.

Function Keys

The function keys F1 through F6 serve as command line editing keys in MS-DOS. When you type a command on the command line and press **Enter**, MS-DOS executes the command. At the same time, MS-DOS automatically stores the characters in the command line in a buffer that acts as a temporary storage area. The function keys enable you to retrieve the characters stored in the buffer to do the following:

- Instantly repeat a command line by pressing a single key
- Edit the command and retry it without retyping the entire command

- Create a command line that is similar to the previous command line with a minimum of typing.

The following example shows how you can use the MS-DOS function keys to edit a command line:

1. Type `DIR PROG.COM` and press **Enter**. MS-DOS displays information about the file PROG.COM on your screen (`File not found`, for example) and saves the command line to the buffer.
2. Press **F3** to copy the contents of the buffer to the command line.
3. Press **Enter** to send the command line to MS-DOS for execution.
4. Press **F2** and **C** to display the characters in the buffer up to but not including the letter C. The command line displays the following:

   ```
   DIR PROG.
   ```

5. Type TXT and press **Enter** to display information about a file named PROG.TXT. The command line DIR PROG.TXT is now in the buffer.
6. Type TYPE, press **Insert** and a space, and then press **F3** and **Enter** to display the contents of PROG.TXT.

The command line TYPE PROG.TXT is now in the buffer.

When you are typing, MS-DOS enters the characters directly into the command line, overwriting the characters in the buffer. When you press **Insert**, as in the above example, you can insert characters into the buffer without overwriting characters.

If you had misspelled TYPE as BYTE, you would see:

```
BYTE PROG.TXT
```

Because this is not an acceptable command, you would receive an error message. Correct the command in one of two ways:

- Press the following keys one after the other:

  T, F1, P, F3

  The T replaces the B, F1 copies the Y, the P replaces the T, and F3 copies the rest of the command line.

- Press the following keys one after the other:

  Delete, Delete, F1, Insert, Y, P, F3

  Pressing Delete twice removes the first and second characters in the buffer. F1 displays the third character in the buffer. Insert, Y, and P insert the two characters, and F3 displays the rest of the buffer characters.

Either method produces the following command line:

```
TYPE PROG.TXT
```

Generating Special Characters

MS-DOS uses 256 characters, each of which is assigned a unique code. You can generate special characters that do not appear on the keyboard using the keys on the numeric keypad with the Alt key. To do this, hold down Alt and enter the decimal value of the character code for the desired character. The special characters and their character codes are shown in Appendix B.

> **Note**
> Remember, you must generate these characters from the numeric keypad. You cannot use the numbers on the top row of the standard keyboard. You cannot display characters in the range from 0 to 31.

When you release Alt, the specified character is displayed at the current cursor position. For example, to print the ½ character, which is code 171, hold down Alt and type 171. When you release Alt, ½ appears.

When you are in graphics mode, you can display the characters from 128 to 255 only after loading the MS-DOS GRAFTABL character generator. See Chapter 4 for information on the GRAFTABL command.

Selecting International Keyboards

MS-DOS allows you to select an international keyboard layout for your system using the KEYB command:

Country/Language	Keyboard Code	KEYB Command	Keyboard ID
Australia	US	KEYB US	103
Belgium	BE	KEYB BE	120
Canada (English)	US	KEYB US	103
Canada (French)	CF	KEYB CF	058
Denmark	DK	KEYB DK	159
Finland	SU	KEYB SU	153
France	FR	KEYB FR	189/120
Germany	GR	KEYB GR	129
Italy	IT	KEYB IT	141/142
Latin America	LA	KEYB LA	171
Netherlands	NL	KEYB NL	143
Norway	NO	KEYB NO	155
Portugal	PO	KEYB PO	163
Spain	SP	KEYB SP	172
Sweden	SV	KEYB SV	153
Switzerland (French)	SF	KEYB SF	150
Switzerland (German)	SG	KEYB SG	000
United Kingdom	UK	KEYB UK	166/168
United States (default)	US	KEYB US	103

See the KEYB command in Chapter 4 for more information.

After you select one of these special keyboard commands, use the appropriate international keyboard type to generate the characters appearing on the keys. Appendix B includes illustrations of these keyboards and complete details.

Chapter 2

Managing Files and Directories

Managing Files 2-3
  Naming Files 2-3
  Maintaining Files 2-6
  Protecting Files 2-6
Using Directories 2-7
  The Default Directory 2-9
  Using Pathnames 2-9
  Including Filenames With Pathnames 2-10
  Including Drive Letters With Pathnames and Filenames 2-10
  Listing the Contents of a Directory 2-11
  Creating Directories 2-12
  Removing Directories 2-12
  Changing the Default Directory 2-13
  Displaying a List of Directories 2-13

Managing Files

You store all your work and programs in files. Usually, you create these files with the application programs you are using with your computer.

This section explains how to name, maintain, and protect your files.

Naming Files

Each file must have a filename. A filename consists of up to eight characters, identifying the information stored in the file. Although it is not essential, you can add up to a three-character extension to further identify the file. (In this manual, the word *filename* represents the name of a file whether or not it has an extension.)

When you use an extension, you must separate it from the filename by a period. For example, a typical word processing filename looks like this:

```
LETTER.DOC
```

Application programs often determine the extensions for the files you create. For example, if you create a file called TEXT with Microsoft Word, Word automatically stores the file as TEXT.DOC and creates a backup copy called TEXT.BAK. Avoid giving your files names and extensions used by your application software.

It is best not to use the following filename extensions because MS-DOS uses them:

BAT, BIN, CLR, COM, CPI, CRF, DAT, EXE, HLP, LST, MAP, MEU, MOS, OBJ, PRO, PRT, SYS

When you name a file, you can use any combination of letters and numbers. But do not use uppercase and lowercase letters to distinguish between files. MS-DOS does not recognize a difference and always displays filenames in uppercase.

Also, do not use a space or any of the following symbols in your filename:

. " / \ { } : ¦ < > + = ; , ? *

Develop a logical method for naming your files to make your data easy to find. For example:

REPORT.TXT	A text file containing a report
MEMO321	A memo written on March 21
2003.INV	Your invoice number 2003

Device names

Certain names are reserved by MS-DOS and used as *device names*. Do not use device names for your filenames. MS-DOS recognizes the following device names:

- AUX (Auxiliary) is the first serial port.
- CLOCK$ is the system clock.
- CON (Console) is the keyboard and monitor.
- PRN (Printer) is your printer.
- NUL (Null) is a non-functional device for testing without real output or input. When you use NUL to test output, all data is sent to the null device, but it is not acted upon. For input, an immediate end-of-file code is sent.
- LPT1 (Line Printer 1) is your printer (the same as PRN).
- COM1 (Communication Port 1) is your serial port (the same as AUX).

LPT2, LPT3, and COM2, COM3, and COM4 are device names for additional printers and serial ports. Your system can recognize these devices, but cannot use them unless they are actually installed.

Using a disk drive descriptor

In addition to the name and extension, you can include a disk drive descriptor in your filename. A disk drive descriptor tells MS-DOS to look for the file on the disk in the drive you specify. If you do not include a drive descriptor, MS-DOS looks for the file on the default drive.

If the file you want is not on the default drive, type the letter of the drive and a colon before the filename:

```
B:NEWFILE.EXT
```

Using wildcards to group files

You can use two special characters called wildcards when referring to your files. A question mark (?) can take the place of any single character and an asterisk (*) can replace any group of characters, even the entire filename or extension.

Using these wildcards allows you to refer to a group of related files such as all the files with .TXT extensions. This enables you to execute a command affecting all of the files in the group without entering each filename separately. For example:

`MEMO?`	Includes MEMO, MEMO1, MEMO2, and MEMOS, but not MEMOS1 (because of the extra character).
`*.EXE`	Is the same as ????????.EXE and represents all files with an .EXE extension.
`PROG*.BAS`	Is the same as PROG????.BAS and includes PROG1.BAS and PROG2.BAS.
`*.*`	Includes all files in the current directory.
`B:*.*`	Includes all files on drive B.

Here are some specific examples of using wildcards in MS-DOS commands (remember to press **Enter** after typing a command):

- To list all files with the extension .BAS on drive B, type:

```
DIR B:*.BAS
```

- To copy all .TXT files from drive A to drive B and rename them with the .DOC extension, type:

```
COPY A:*.TXT B:*.DOC
```

- To delete all files from the current directory whose names begin with ME and are followed by a single character, type:

```
DEL ME?
```

  Some MS-DOS commands operate only on single files, not groups of files. With these commands, you cannot use wildcard characters. See the description of the specific command, as appropriate.

Maintaining Files

Here are some good file management techniques to follow:

- Delete files you no longer need.
- Give files new names if they contain new information that no longer matches their names.
- Make copies of important files.

Chapter 3 provides general descriptions of the MS-DOS commands you use to maintain your files, and Chapter 4 describes each command in detail.

Protecting Files

Follow these precautions to safeguard the information you store on disk:

- Copy new data to backup diskettes every day to ensure your backup copy is up-to-date.

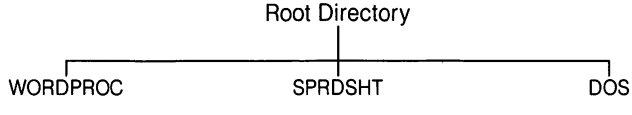
- Never remove a diskette when the drive's light is on. (This does not physically damage the diskette, but may prevent important data from being written to the diskette.)
- Write-protect important diskettes and backup copies. This prevents the disk drive from writing on the diskette. (Be aware that some application programs do not work properly if the program diskette is write-protected.)

Additionally, you can prevent any file from being accidentally altered by designating it as *read-only*. To do this, use the ATTRIB command as described in Chapter 4.

Using Directories

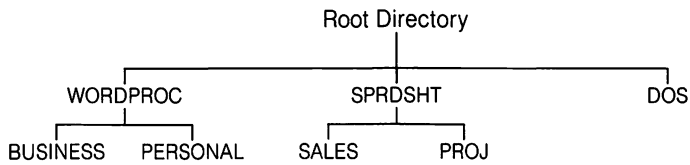
You can create many files on a diskette, and you can store thousands of files on a hard disk. To help you organize this much information, MS-DOS lets you subdivide a disk or a diskette into logical units called directories. Directories allow you to organize your work to keep files of similar type or purpose together.

When you format a hard disk or a diskette, MS-DOS creates a directory for you. This directory is called the *root* directory. Any additional directories you create stem from the root directory; that is, they are *subdirectories* of the root directory. A simple directory structure might look like this:



This structure would enable you to keep your word processing programs and text files in a directory called WORDPROC, your spreadsheet program and data files in a directory called SPRDSHT, and the MS-DOS files in a directory called DOS. The few files that MS-DOS needs to find as soon as you boot your system (COMMAND.COM, CONFIG.SYS, and AUTOEXEC.BAT) could remain at the top level of the structure, in the root directory.

As the number of files in your WORDPROC and SPRDSHT directories increases, you can create additional directories to support a more detailed organization, such as this:



This structure lets you separate your business text files from your personal text files and your sales spreadsheets from those you use for financial projections.

Your directory structure may be as simple as this example or much more complex. You can organize your disk(s) to suit your own needs, and you can modify the structure as your needs change.

Here are some points to note about directories:

- In the root directory, the total number of files and subdirectories must not exceed 512.
- All directories other than the root directory can have any number of files and subdirectories.
- You name subdirectories the same way you name files. You can use as many as eight characters for the name and as many as three characters for the extension.
- The root directory does not have a name. It is identified by a backslash (\) character.
- Diskettes have root directories, just as hard disks do, and you can create subdirectories on diskettes the same as on hard disks.
- You can create as many directories on a disk or diskette as you like. But be sure that the entire pathname, from the root directory to your deepest subdirectory, is not longer than 63 characters.

The following sections discuss the basics of creating, deleting, and using directories.

The Default Directory

MS-DOS always recognizes one directory as the *default* or *current* directory, just as it always recognizes one drive as the default drive. The default directory is the one in which MS-DOS performs your commands, unless you specify otherwise. If you want to run a program or access a data file that is not stored in the default directory, you can either change directories (making a different directory the default) or include a *pathname* in your command.

Using Pathnames

A pathname tells MS-DOS how to locate the directory you want to access. There are two basic types of pathnames: relative and absolute. A relative pathname tells MS-DOS how to find its way to the directory *from the current default directory*. An absolute pathname tells MS-DOS how to find its way to the desired directory *from the root directory*. You can use either relative or absolute pathnames at any time.

Here is an example of an absolute pathname:

```
\WORDPROC\PERSONAL
```

The backslash at the beginning of this pathname tells MS-DOS to start its journey at the root directory, proceed down the directory tree to WORDPROC, then continue down the tree to PERSONAL.

Here is an example of a relative pathname:

```
SALES
```

Because this pathname does not begin with a backslash, MS-DOS assumes that the starting point of the path is the current default directory. This pathname thus tells MS-DOS to find a directory named SALES, which is a subdirectory of the current default directory.

Relative pathnames can tell MS-DOS to move *upward* in the directory tree as well as downward. The symbol . . (two periods) in a pathname tells MS-DOS to move upward one level in the tree. For example, if the default directory were WORDPROC (in the previous example), the pathname . . \DOS would tell MS-DOS to move up one level from WORDPROC (to the root directory) and then find a subdirectory called DOS.

Including Filenames With Pathnames

You typically use a pathname when you want to access a file that is not stored in the current default directory. The name of the file you want to access is specified at the end of the pathname, like this:

```
TYPE \WORDPROC\PERSONAL\JEAN1204.DOC
```

This command tells MS-DOS to list on screen (TYPE) the contents of the file JEAN1204.DOC, which is stored in the directory \WORDPROC\PERSONAL. Note that the filename is connected to the pathname by a backslash character—the same character you use to separate the various directories in the pathname.

Including Drive Letters With Pathnames and Filenames

If you want to access a file stored on a drive other than the default drive, you must include a drive designator (for example, A:) with the filename. If the file you want is not stored in the default directory of that drive, you must also include a pathname as well as the drive designator.

For example, suppose you are logged onto the root directory of drive C, and you want to list the contents of the file JEAN1204.DOC, which is stored in the directory \WORDPROC\PERSONAL of drive D. The last time you logged onto drive D, the default directory on that drive was the root directory. If you issue the command

```
TYPE D:JEAN1204.DOC
```

MS-DOS attempts to find the file you want in the root directory of drive D. Because the file is not in the root directory, MS-DOS displays an error message. To correct the problem, add a pathname to the command, like this:

```
TYPE D:\WORDPROC\PERSONAL\JEAN1204.DOC
```

If you are not sure which directory is the default, include the full pathname.

MS-DOS provides some commands that make using pathnames easier. See Chapter 4 for information on these helpful commands.

- APPEND lets you set a search path for data files and programs.
- PATH lets you specify a search path for program files so you don't have to type a full pathname every time you want to run an application program or an MS-DOS command.
- SUBST lets you substitute a drive letter for a directory path, which is helpful if you frequently type long pathnames.

Listing the Contents of a Directory

To list the files in the current directory, type DIR and press **Enter**. MS-DOS lists the names of the files in the current directory on the current drive.

Directory listings include the following information about each file: the name and extension, the size in bytes, and the date and time the file was created or last modified. Subdirectories are listed along with files and are identified by the letters <DIR>. At the bottom of the listing, MS-DOS indicates the number of bytes on the disk that are still available for use.

If the listing is too long to fit on one screen, add the /P switch to the command, like this:

```
DIR /P
```

This switch instructs MS-DOS to pause after displaying one screen of information. To see the next screen, press any key. Another way to view a long directory is to use the /W switch:

```
DIR /W
```

This tells MS-DOS to display a wide-format directory listing, which includes only file and directory names.

To list the contents of a different drive or a different directory, include the appropriate drive designator and/or pathname in the command. For example, to list the files in the root directory of the diskette in drive A, type the following and press **Enter**:

```
DIR A:\
```

Creating Directories

The MKDIR command lets you create directories. For example, to create a LEDGER directory under your root directory, type the following and press **Enter**:

```
MKDIR \LEDGER
```

You can abbreviate the name of this command to MD. For example, to create a SALES directory under the LEDGER directory, type the following and press **Enter**:

```
MD \LEDGER\SALES
```

Removing Directories

To delete a directory from a disk, use the RMDIR command (or RD). For example, to remove the directory ACCOUNTS from the directory LEDGER, log onto the LEDGER directory, type the following, and press **Enter**:

```
RD ACCOUNTS
```

A directory must be empty before you can delete it. If it contains any files or subdirectories, attempting to delete it produces an error message. Delete the files or copy them to another directory. Once the directory is empty, you can delete it.

Changing the Default Directory

To change from one directory to another, use the CHDIR command (or CD). For example, to change to the root directory from anywhere in the directory tree, type the following and press **Enter**:

```
CD \
```

If you are in the LEDGER directory and you want to change to ACCOUNTS, a subdirectory of LEDGER, type the following and press **Enter**:

```
CD ACCOUNTS
```

To change from the ACCOUNTS to the LEDGER directory, you can use the special symbol . . or you can use an absolute pathname. (The . . symbol always designates the parent directory of the current default directory.) You can type:

```
CD ..
```

or

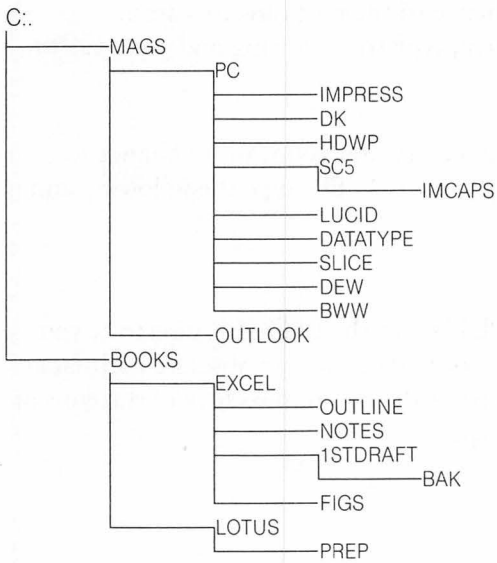
```
CD \ACCOUNTS
```

Displaying a List of Directories

With the TREE command, you can display a list of all the directories on a drive. For example, to see the names of all directories on drive C:, type the following and press **Enter**:

```
TREE C:\
```

The screen displays a diagram showing the directory hierarchy for drive C, like this:



If you want to see a list of all the files in all the directories, add the /F switch, like this:

```
TREE C:\ /F
```

Note
When using the TREE command, be sure you are logged onto the drive (and directory) where TREE.COM is stored or specify the drive location in the command.

Chapter 3

Using MS-DOS Commands

MS-DOS Command Overview 3-3
- Internal and External Commands 3-3
- Diskette Operations 3-4
- Hard Disk Operations 3-5
- File Operations 3-5
- Directory Operations 3-6
- MENU and SETPRINT Utilities 3-7
- System Configuration and Operation 3-7
- Batch Processing Operations 3-10
- Programming Tools 3-10

Entering Commands 3-11
- Command Formats 3-11
- Rules for Entering Commands 3-14
- Using Batch Files to Combine MS-DOS Commands 3-15
- Input and Output 3-18

MS-DOS Command Overview

To help you understand how MS-DOS works, this section presents an overall look at the different categories of MS-DOS commands.

Internal and External Commands

Each MS-DOS command is either internal or external. Internal commands are built into MS-DOS. Once you start your computer and load MS-DOS, you can execute these internal commands any time you see the MS-DOS command prompt. External commands are stored on your system diskettes as program files. Any file with an extension of .BAT, .COM, or .EXE is an external command. MS-DOS must read these commands from the disk in order to run them.

Your system diskettes contain the files you can access. You can view these files on your screen when you type the directory command, DIR, and press **Enter**.

Two files on the system diskettes do not appear in the directories. These files contain the instructions that make up the operating system. You cannot access these files.

The internal commands, which are always immediately available in memory, do not show on the directory either. These commands are stored in the COMMAND.COM file. COMMAND.COM is the command interpreter, which displays the command prompt, accepts your commands, and runs programs. These are the internal commands:

BREAK	DEL	MKDIR	SET
CALL	DIR	PATH	SHIFT
CHCP	ECHO	PAUSE	TIME
CHDIR	ERASE	PROMPT	TYPE
CLS	EXIT	REM	VER
COPY	FOR	RENAME	VERIFY
CTTY	GOTO	RMDIR	VOL
DATE	IF		

It's easy to understand the MS-DOS commands if you think of them in these functional categories:

- Diskette operations
- Hard disk operations
- File operations
- Directory operations
- System configuration and operation
- Batch processing operations
- Programming tools.

Every MS-DOS command falls into one of these categories. The remainder of this chapter examines the commands by category, name, and function.

Diskette Operations

The following external commands let you manage the information on your diskettes:

CHKDSK lets you check the status of a diskette to determine the available space and the size of your files. CHKDSK also checks and reports the condition of diskettes and files and, optionally, fixes any errors found in the directory or file allocation table. (CHKDSK functions in the same manner on a hard disk.)

DISKCOMP compares diskettes and identifies differences between them. This is useful if you need to determine whether two diskettes are exactly alike or if you want to verify a DISKCOPY operation.

DISKCOPY copies the complete contents of one diskette to another. (It formats the diskette if necessary.)

FORMAT formats diskettes and assigns volume serial numbers. Optionally, FORMAT installs system files and puts volume labels on diskettes. Before using new diskettes for the first time, you must format them.

LABEL lets you create, change, or delete a diskette volume label.

SYS transfers operating system files to a diskette or hard disk.

Hard Disk Operations

The following external commands let you prepare and maintain a hard disk:

BACKUP lets you back up files. Use this command to store your hard disk files on diskettes. (Restore them from diskettes to the hard disk with the RESTORE command.)

FDISK partitions the hard disk to establish a work area for the MS-DOS operating system on your hard disk. It also enables you to start MS-DOS automatically from your hard disk (rather than a diskette) every time you turn on or reset your computer.

RESTORE lets you restore hard disk files. Use this command to restore your files from diskettes (which you stored previously with the BACKUP command) to the hard disk.

File Operations

The following external commands help you manage your files:

ATTRIB lets you set or reset the read-only attribute of a file and set the archive bit on or off. You can also use this command to display the attributes of a file. By setting a file's attribute to read-only, you protect the file from being accidentally altered or deleted.

COMP lets you compare the contents of two files or two groups of files to see if they are different.

FASTOPEN speeds up file access time by storing directory information in the computer's memory.

FC compares two files and either verifies that they are identical or reports the differences.

FIND locates text strings within files.

MORE is a command filter you can use when displaying files. This command filters screen output so that a pause occurs each time the screen fills. (Filters are described below.)

PRINT is the file printing command that outputs your MS-DOS text files to a printer. While your file is printing, you can use the computer for other tasks.

RECOVER attempts to save damaged disks or disk files. Use this command if you continually receive disk error messages when accessing a particular disk or file.

SHARE is the command that sets up your computer to support file sharing through the network system. This command has no effect unless a network system has been installed.

REPLACE lets you replace files on the target disk with files of the same name from the source disk. You can also add new files, copying them from the source to the target.

SORT sorts strings in text files. This command arranges the records in a file in alphabetical or numerical order.

XCOPY lets you copy groups of files or entire subdirectories.

Internal commands

These internal commands also help you manage your files:

COPY copies your files to other disks, directories, or devices.

DEL, also called *ERASE*, deletes files from your disks.

RENAME, also called *REN*, renames files on your disks.

TYPE displays text files on your screen.

Directory Operations

The following external commands create and maintain directories:

APPEND displays or sets paths to directories containing program or data files.

JOIN connects a directory on one drive to a different drive. This allows certain applications to access a different drive without having to change the current drive.

SUBST lets you substitute a different drive letter for another drive or path. Using this command, you can shorten a long pathname to just one letter.

TREE lets you display graphically all the directories on a disk and, optionally, list all the files they contain. Use this command to display your disk file structure in one complete listing.

Internal commands

These internal commands also maintain your directories:

CHDIR, also called *CD*, shifts you to a different directory within the directory structure of a disk.

DIR displays the contents of a disk directory.

MKDIR, also called *MD*, creates (makes) a new directory on your disk.

RMDIR, also called *RD*, removes an empty directory from your disk.

MENU and SETPRINT Utilities

Epson offers two menu utilities: MENU and SETPRINT. MENU provides simple, easy-to-use menus for certain complex MS-DOS commands and also lets you access any MS-DOS command. SETPRINT lets you define the printing defaults for your printer. See Chapter 5 for instructions on these menu utilities.

System Configuration and Operation

The following external commands let you define your system's configuration and operation:

ASSIGN lets you assign a drive letter to a different drive. Because some application programs require that you place data diskettes in a particular drive, this command is useful in redirecting input/output to a drive of your choosing.

DOSSHELL invokes the MS-DOS Shell, which allows you to select MS-DOS commands and run programs from menus. The MS-DOS Shell is described in the MS-DOS 4.0 Shell User's Guide.

GRAFTABL loads a special character generator table for use when the display screen is set to graphics mode.

GRAPHICS (for 9-pin printers) and *GRAPH24.COM* (for 24-pin printers) let you make exact printed copies of your screen's image.

HELP displays information about all the MS-DOS commands and utility programs.

INSTALL lets you load any of the following commands by including them in your CONFIG.SYS file: FASTOPEN, KEYB, NLSFUNC, and SHARE. See Chapter 6 for more information about the CONFIG.SYS file.

KEYB loads a special program that redefines the keyboard country layout.

MEM displays information about available memory and current memory usage.

MODE allows you to change the configuration of printer output, monitor output, serial port operation, keyboard typematic rate, and code pages.

NLSFUNC enables code page switching.

SYS files

These files contain device drivers that you install with a CONFIG.SYS file. See Chapter 6 for more information on CONFIG.SYS.

ANSI.SYS is a file that contains a set of standard escape codes for console control. Some application programs you run may require this standard console driver interface. If so, the documentation for the application program specifies it. When you do need ANSI.SYS, enter it in the CONFIG.SYS file with the DEVICE=ANSI.SYS command. This automatically places it in memory at startup. See Appendix D for more details about ANSI codes.

DRIVER.SYS lets you copy files from one drive to the *same* drive when the system has more than one diskette drive. Without DRIVER.SYS, files are automatically copied to the second drive. When you need DRIVER.SYS, just list it in a CONFIG.SYS file with the command DEVICE=DRIVER.SYS. This automatically places it in memory at startup. See Chapter 6 for more information on CONFIG.SYS and DRIVER.SYS.

VDISK.SYS simulates a disk drive by using a portion of your computer's memory as the storage medium. When you need VDISK.SYS, just list it in a CONFIG.SYS file with the command DEVICE=VDISK.SYS. This automatically places it in memory at startup.

Internal commands

These internal commands also let you configure and operate your system:

BREAK enables or disables command aborting by **Ctrl C**.

CHCP displays or changes the current code page (character set).

CLS clears the screen.

CTTY changes the standard input/output device.

DATE displays and allows you to set the date.

EXIT returns to a previous level after an application program or shell has invoked the MS-DOS command interpreter. For example, after calling the MS-DOS command prompt from the MS-DOS Shell, you use EXIT to return to the MS-DOS Shell.

PATH displays or sets paths to directories containing command files.

PROMPT restores or changes the characters that make up the command prompt.

SET displays or sets your system parameters in the current environment.

TIME displays and allows you to set the time.

VER displays the number for the MS-DOS version you are using.

VERIFY enables or disables a verification of all files written to disks.

VOL displays the volume identification of your disks.

Batch Processing Operations

The commands you use for batch processing are all internal commands. These commands add flexibility and power to your batch programs. However, you do not need to use any of these commands in order to use your operating system or application programs.

CALL allows a batch file to call another batch file without loading another copy of COMMAND.COM.

ECHO enables or disables screen output during batch file operations.

FOR allows conditional repetitive processing in batch file operations.

GOTO jumps to a specified label in batch file operations.

IF specifies parameters for conditional processes in batch file operations.

PAUSE interrupts processing of batch files until you are ready to continue.

REM lets you add remark statements to batch files to explain their operation.

SHIFT allows you to access more parameters in batch files.

Programming Tools

Your system diskettes contain four standard MS-DOS utilities you can use in programming and designing assembly routines. You do not need to run any of these in order to use your operating system or application programs.

DEBUG is the debugging tool that examines and modifies code and data in memory. This command monitors and modifies your computer's memory.

EDLIN is the editing utility for creating and modifying text files.

EXE2BIN is the conversion utility for changing .EXE files into .BIN files.

LINK is the utility that combines your object code modules into one executable program. Use this command when constructing programs. (You need a language assembler or compiler to create object modules.)

Entering Commands

This section explains the format for MS-DOS commands and the various parameters you can use. It also describes how to group commands in a batch file and redirect input or output from one device to another.

Command Formats

To enter an MS-DOS command, you must understand the format of the command. The format specifies everything MS-DOS needs to perform a task. This includes the name of the command and any parameters.

In this manual, the command name appears in all uppercase letters. You can enter the name in upper- or lowercase letters. MS-DOS does not recognize the difference.

The parameters you specify for each command are determined by the command itself. Some parameters are required while others are optional.

This manual uses certain symbols and typographical conventions to illustrate what the command parameters are and how you enter them.

Square brackets ([ ]) indicate an optional entry. Ellipses (...) indicate that you may repeat a parameter as many times as needed or desired. Do not enter the brackets or the ellipses when you type the command.

You must enter the following items:

- All other punctuation, such as commas, colons, slash marks, and equal signs, *exactly* as shown.
- Any parameter that does not have square brackets around it. (This indicates that the command requires these parameters.)
- All items shown in uppercase letters. (While you can enter commands in upper- or lowercase, be sure to spell them correctly.)

Items shown in lowercase italic letters are parameters, as in this example:

```
TYPE filename
```

You must enter both TYPE and *filename* because they are shown without special symbols. TYPE is the name of the command and *filename* is a parameter. You can type any filename.

The chart below identifies the parameters you see in this book. The descriptions show both the format and an example for each parameter. The format shows what you see in the Format section for each command (in chapters 4, 6, 7, and 8) and the example shows one example of what you could type for that parameter.

Parameter	*Description*
d:	Disk drive descriptor such as A: or B:. Shows which drive contains the file or files you want. A colon always follows a disk drive descriptor. Format: *d:* Example: A:
path	The pathname showing how to locate the file on the disk, including any necessary backslashes and the name of each directory in the path. Format: *path* Example: \LEDGER\SALES\WEST The total number of characters in a path, together with the disk drive descriptor, must not be greater than 63.

Parameter	*Description*
filename	The name of a file with or without an extension. This parameter does not include the disk drive descriptor. When you do not use a disk drive descriptor, MS-DOS looks for the file on the disk in the current default drive. Filenames can be from one to eight characters. The optional extension can be up to three characters.

Note
Whenever a file is part of a path, the filename must be preceded by a backslash. The filename is always the last name in a path, separated from the last directory name by a backslash.

You can use wildcard characters in the filename unless noted in the command description. The wildcard characters are `*` and `?`. Chapter 2 describes how to use wildcard characters.

Format: *`filename`*
Example: `NEWS.TXT`

n or *nnn* — A number that varies according to the command.

Format: *`n`*
Example: `9`

x or *xxx* — A character (or characters) that varies according to the command.

Format: *`x`*
Example: `R`

Certain commands may allow or require other parameters not shown here. The individual descriptions for these commands (in Chapters 4 through 8) identify any special parameters.

For instance, some commands can include optional switches that affect the way the commands work. You always precede the switch name with a slash (/).

A typical command format using switches looks like this:

```
DIR [d:][path][filename][/P][/W]
```

You can enter this DIR command in a number of ways, each yielding different results. Some examples are:

```
DIR
DIR\LEDGER\SALES
DIR B:\
DIR A:ACCPAY/P
DIR C:/W
```

Rules for Entering Commands

Follow these rules when entering MS-DOS commands. Remember, one or more parameters usually follow command names:

- Enter command names and parameters in either uppercase or lowercase letters.
- Separate command names and parameters by *delimiters*. You usually use spaces or commas as delimiters. For example:

  ```
  TYPE TEXTFILE
  ```

  You can also use the semicolon (;), the equal sign (=), or the tab key as delimiters. This manual shows the space only.
- Do not separate a file specification with delimiters. The colon, backslash, and period already serve as delimiters:

  ```
  A:\SALES\LETTER.DOC
  ```
- Include the file extension when referring to a file with an extension.
- For an external command, type the command *without* its extension (.COM, .EXE, or .BAT). For an internal command, there is no extension; simply type the command name.

- For an external command, you may need to add the drive descriptor and directory if the file is not in the current directory. For example, if you are logged onto drive B and want to use the MENU program, which is on drive A in the \DOS directory, you would type:

  ```
  A:\DOS\MENU
  ```

- You can specify a path preceding the name of a command. This lets you store the MS-DOS commands in subdirectories. For example, if the file CHKDSK.COM is in the \DOS subdirectory of drive C, you can enter the command by typing:

  ```
  C:\DOS\CHKDSK
  ```

- You can use batch processing commands, such as IF and FOR, when specifying an MS-DOS command.
- Always press **Enter** to execute a command.

This version of MS-DOS provides network support. You can use most MS-DOS commands on a network disk, directory, or printer. The following commands, however, may not work in a network environment:

CHKDSK
DISKCOMP
DISKCOPY
FORMAT
JOIN
LABEL
RECOVER
SYS

Using Batch Files to Combine MS-DOS Commands

If you find yourself typing the same sequence of commands over and over, you can list the commands in a special file called a batch file. When you execute the batch file, MS-DOS processes the entire sequence of commands in the order you entered them.

You can store in a batch file any command that you normally run from the command prompt.

You can create batch files with any text editor, including the MS-DOS Line Editor, EDLIN (described in Chapter 8). You can also use the COPY command with the CON device. The CON device allows you to enter data from your keyboard and copy the data to a file.

The batch filename must include the extension .BAT so that MS-DOS recognizes it as a batch file. To execute a .BAT file, type its name without the extension. For example, if your batch filename is END.BAT, type END and press **Enter**.

As an example, follow the steps below to create a batch file called SAMPLE.BAT. It erases all files on drive A with an extension of .BAK, displays a message indicating the files have been erased, and displays the directory of drive A. (This example uses the COPY command with the CON device.)

Type each of the following lines exactly as shown. Press **Enter** after typing each line:

```
COPY CON SAMPLE.BAT
DEL A:*.BAK
REM All .BAK files have been erased.
DIR A:
```

Press **Ctrl Z** and then **Enter** to close the file.

Now, to use this batch file, simply type SAMPLE at the command prompt and press **Enter**.

You can specify another batch file as the last command in a batch file. In this way, you can link batch files. You can also use the CALL command to run one batch file within another. When MS-DOS finishes executing the second batch file, it continues where it left off in the first. For more information, see the description of CALL in Chapter 7.

You can terminate execution of any batch file by pressing **Ctrl Break**.

If you run a batch file from a diskette, and you change to a different diskette before MS-DOS can execute all the commands in the batch file, you see this message:

```
Insert diskette with batch file
Press any key to continue . . .
```

The AUTOEXEC.BAT file

When you boot MS-DOS, the command processor looks for a file called AUTOEXEC.BAT on your system diskette or hard disk. AUTOEXEC.BAT is a batch file you can create to automate your system boot-up routine. By grouping several commands in this file, you can tell your computer to perform them automatically, in sequence, every time you load MS-DOS.

You must store AUTOEXEC.BAT in the root directory of your MS-DOS startup diskette or hard disk. If the command processor finds AUTOEXEC.BAT, the batch file immediately executes, bypassing the date and time prompts. See Chapter 7, Batch Processing Commands, for more information.

You can create AUTOEXEC.BAT with a text editor such as EDLIN. If you use the COPY command with the CON device, remember to press **Ctrl Z** or **F6** and then press **Enter** to close the file.

Creating a batch file with replaceable parameters

You can create a batch file with replaceable parameters, enabling you to run the same batch file with different sets of data.

You can specify up to 10 *dummy* parameters (%0 through %9) within a batch file. When you execute the batch file, the values you supply sequentially replace the %1 through %9 parameters. The filename of the batch file and the disk drive descriptor, if specified, always replace the %0 parameter.

As an example, create a a batch file called MYFILE.BAT that consists of three lines:

```
COPY %1.MAC %2.MAC
TYPE %2.PRN
TYPE %Ø.BAT
```

To run the file, type the following and press **Enter**:

```
MYFILE A:PROG1 B:PROG2
```

The program substitutes MYFILE for %0, A:PROG1 for %1, and B:PROG2 for %2. The result is the same if you execute the following commands:

```
COPY A:PROG1.MAC B:PROG2.MAC
TYPE B:PROG2.PRN
TYPE MYFILE.BAT
```

If you use the percent sign as part of a filename within a batch file, you must type it twice. For example, to specify the file ABC%.EXE, you type `ABC%%.EXE` in the batch file. When the batch file executes, you see `ABC%.EXE`, not `ABC%%.EXE`.

Input and Output

MS-DOS assumes that the keyboard is your input device and the monitor is your output device. You can redirect the input or output to a file or another device (such as a printer). You can also use filters and pipes (described below) to manipulate your input and output.

Redirecting input and output

You can instruct a program to read input from a file rather than the keyboard, and you can direct output to a file or printer instead of the monitor screen.

To direct MS-DOS to read input from a file instead of the keyboard, use the less-than sign (<) before the filename. For example, this command sorts the text file NAMES.TXT:

```
SORT <NAMES.TXT
```

To direct output to a file instead of the monitor screen, use the greater-than sign (>) before the filename. For example, this command stores the directory of the default disk drive in a file called MYFILES.DIR:

```
DIR >MYFILES.DIR
```

Note
If the file MYFILES.DIR already exists, MS-DOS overwrites what is in the file with the new data.

If you want to append (add) data to a file, rather than overwrite the file, type two greater-than signs (>>) before the filename. (If the file does not exist, MS-DOS creates it.) For example, this command appends the directory listing of the default drive to the existing file named MYFILES.DIR:

```
DIR >>MYFILES.DIR
```

MS-DOS recognizes a parallel printer by two device names: PRN and LPT1. Use either of these device names to redirect data output to a printer. For example, to send the file ABCDIR to the printer, turn on the printer and use one of the following commands:

```
COPY ABCDIR PRN:

COPY ABCDIR LPT1:
```

You can send the MS-DOS file EPSON.TXT to the printer with either of the following commands:

```
TYPE EPSON.TXT >PRN:

TYPE EPSON.TXT >LPT1:
```

The RS-232C serial port can both send and receive data. Therefore, output can be sent to it, and input can be accepted from it. The device names for the RS-232C port are AUX, COM1, COM2, COM3, and COM4. (COM2, COM3, and COM4 are available only if you have installed additional serial ports in your computer.)

To communicate with another MS-DOS based computer via the RS-232C serial port and send the EPSON.TXT file to the other computer, follow these steps:

1. Match signal parameters with the SETUP or MODE command.
2. Type the following, but *do not press* **Enter**:

```
COPY EPSON.TXT AUX:
```

3. Wait for the receiving end to type `COPY AUX: EPSON.TXT`.
4. When the receiving end presses **Enter**, you must press **Enter**.

Using filters

A filter is a command that reads your input, changes it in some way, and then outputs it. In this process, the data is filtered by the program.

MS-DOS filters include the following:

- FIND searches for a string of text in a file.
- MORE displays text on your screen one screen page at a time.
- SORT sorts text in alphabetical or numerical order.

You can put these filters together in many different ways and create functions that take the place of a large number of specific commands. See Chapter 4 for more details.

Command piping

You can give more than one command to the system at a time by piping commands to MS-DOS. To do this, enter a pipe separator (¦) between command names. For example, this command gives you an alphabetically sorted list of your directory:

```
DIR ¦ SORT
```

The pipe separator sends all output for processing—generated by the command on the left side—to the command on the right side of the bar. In other words, the output from the command on the left becomes the input for the command on the right.

You can link more than two commands. For example, this command sequence sorts your directory and displays it one screen at a time:

```
DIR ¦ SORT ¦ MORE
```

You can link several commands together as long as the output of one is usable as the input of the next.

Chapter 4

Alphabetical List of MS-DOS Commands

Introduction 4-3
APPEND 4-5
ASSIGN 4-10
ATTRIB 4-12
BACKUP 4-15
BREAK 4-20
CHCP 4-21
CHDIR 4-23
CHKDSK 4-25
CLS 4-28
COMMAND 4-29
COMP 4-32
COPY 4-35
CTTY 4-39
DATE 4-41
DEL 4-43
DIR 4-45
DISKCOMP 4-47
DISKCOPY 4-51
DOSSHELL 4-54
ERASE 4-55
EXIT 4-57
FASTOPEN 4-59
FC 4-62
FDISK 4-71
FIND 4-85
FORMAT 4-87
GRAFTABL 4-94
GRAPH24 4-96
GRAPHICS 4-97
HELP 4-100
JOIN 4-101
KEYB 4-104

LABEL 4-108
MEM 4-110
MKDIR 4-112
MODE 4-114
MORE 4-129
NLSFUNC 4-130
PATH 4-132
PRINT 4-134
PROMPT 4-137
RECOVER 4-140
RENAME 4-142
REPLACE 4-143
RESTORE 4-146
RMDIR 4-150
SELECT 4-151
SET 4-152
SHARE 4-154
SORT 4-156
SUBST 4-158
SYS 4-160
TIME 4-161
TREE 4-163
TYPE 4-165
VER 4-166
VERIFY 4-167
VOL 4-168
XCOPY 4-169

Introduction

This chapter and the following four chapters contain descriptions of the MS-DOS commands in alphabetical order. The commands you use most often are covered in this chapter. Chapter 5 describes the menu utilities created by Epson: MENU and SETPRINT. Chapter 6 describes the commands you store in the CONFIG.SYS file, Chapter 7 describes batch processing commands, and Chapter 8 describes programming commands.

For a general description of the functional categories of commands, see "MS-DOS Command Overview" in Chapter 3. For instructions on how to enter commands, see "Entering Commands," also in Chapter 3.

The commands are presented in a standard format that provides the following information:

- Command name—appears at the top of the page in bold uppercase letters.
- *Internal* or *External*—appears across from the command name and identifies whether the command is internal or external.
- Purpose—briefly describes what the command does.
- Format—gives the correct syntax for entering the command. Be sure to read "Command Formats" in Chapter 3 for a complete explanation of the command syntax.
- Description—explains in greater detail how to use the command.
- Cautions—when applicable, identify any special conditions to avoid or be aware of when using the command.
- Examples—show one or more examples of the command and describe the result.

Although the Format section of each command may require more than one line to show all the possible variables, always type the complete command on one line, including all parameters, before pressing **Enter**.

To execute an external command, MS-DOS must be able to access the .BAT, .COM, or .EXE file that contains the command instructions. This means:

- You must be logged onto the drive and directory containing the command file, or
- You must include the drive descriptor and pathname before the command name, or
- You must previously have specified the drive and directory containing the command file in an APPEND or PATH.

See the APPEND and PATH descriptions in this chapter for information about executing a command when you are not logged onto the drive and directory containing the command file.

APPEND

External/Internal

Purpose

Sets a search path for data files. With the /X switch, also sets a path for executable files.

Format

First time:

```
[d:][path]APPEND [d:]path[;[d:]path. . .]
[/PATH:ON][/PATH:OFF][/X:OFF][/X:ON][/X]
```

or

```
[d:][path]APPEND [/E][/PATH:ON][/PATH:OFF]
[/X:OFF][/X:ON][/X]
```

After APPEND is loaded:

```
APPEND [d:]path[;[d:]path...][/PATH:ON]
[/PATH:OFF][/X:OFF][/X:ON][/X]
```

or

```
APPEND[;]
```

Description

Use APPEND to identify the directories that you want MS-DOS to search to locate files that are not in the current directory. Once you enter the APPEND command, you can access files that are in different directories without entering their pathnames.

APPEND is similar to PATH. The main difference is that PATH enables MS-DOS to locate executable (.BAT, .COM, or .EXE) files only. You can use APPEND for both executable files and data files.

Whenever you request a file from MS-DOS, it first searches the current directory, then the directories listed in the APPEND command, then any directories specified with the PATH command.

Because the APPEND command is an external command, you need to precede it with the drive descriptor and pathname the first time you use the command if it is not in the current directory. After you load APPEND, it becomes an internal command. Therefore, if you give the APPEND command again later, you do not need to specify its location.

Use the format APPEND *path;path;path* to specify several paths by separating each path with a semicolon. MS-DOS searches the paths in the order you specify.

Type APPEND; (with a semicolon) to set a null path. This instructs MS-DOS to look only in the current directory for the specified file or files.

To display the current search path, type only APPEND, with no pathnames or parameters.

Each time you use the APPEND command, the new path you specify replaces the path specified in the previous APPEND command.

You can use the following switches with APPEND:

Switch	*Function*
;	Cancels all APPEND paths if it is the only parameter
/E	Stores the path information in the MS-DOS environment (may be used only the first time you invoke APPEND)
/PATH:OFF	Cancels APPEND searching when a filename is preceded by a drive or path specification
/PATH:ON	Instructs APPEND to search even when the filename is preceded by a drive or path specification; this is the default condition
/X:OFF	Allows APPEND to search for nonexecutable (data) files only; this is the default condition

Switch	*Function*
/X:ON	Allows APPEND to search for executable as well as nonexecutable files
/X	Same as /X:ON

Normally, you use the APPEND command to locate data files and the PATH command to locate executable files—program files that have an extension of .COM, .EXE, or .BAT. However, you can add the /X switch to the APPEND command to include program files, as well as data files, in the search path.

Note
Certain MS-DOS commands, such as BACKUP and RESTORE, set their own search paths. If you include the /X switch the first time you use the APPEND command, you must set the search path to null with the command APPEND; before using either the BACKUP or RESTORE command.

Use the /E switch to store the APPEND paths in the MS-DOS environment. This enables you to display and change the APPEND MS-DOS environment values with either the APPEND or SET command. Without the /E switch, MS-DOS stores the paths only in APPEND, and you can use only the APPEND command to display or change paths. For the command to be effective, you must include the /E switch the first time you give the APPEND command after loading MS-DOS.

Cautions

Although APPEND can retrieve a file from a different directory, it does not automatically return the file to its proper location when storing the file. If you edit a file from another directory that was located by the APPEND command, be sure to specify the complete pathname when you store the file. Otherwise, the file is stored in the *current* directory, not the one it came from.

If you use the ASSIGN command to reassign drives, do so after you use the APPEND command the first time.

The maximum number of characters you can specify in an APPEND command is 128.

If you make any mistakes when entering the path information in the APPEND command, such as an invalid drive descriptor or pathname, MS-DOS does not detect the error until it searches for the specified paths.

Examples

To tell MS-DOS to search the \WORDPROC\BUSINESS directory on drive C for both data files and program files, type the following and press **Enter**:

```
APPEND C:\WORDPROC\BUSINESS /X
```

You may substitute `/X:ON` for `/X`.

To search this path and two others, type the following and press **Enter**

```
APPEND C:\WORDPROC\BUSINESS;C:\ELECMAIL\IN;
C:\LEDGER\SALES
```

To display the current set of APPEND paths, type the following and press **Enter**

```
APPEND
```

To ensure that MS-DOS does not search the APPEND paths when the target filename is preceded by a drive designator or pathname, type the following and press **Enter**:

```
APPEND /PATH:OFF
```

When a filename is preceded by a drive designator or pathname, MS-DOS always begins by searching the specified drive and directory. If it fails to find the file, it continues by searching the APPEND paths. If you issue the APPEND command with /PATH:OFF, MS-DOS does not search the APPEND paths. Instead, it issues a `File Not Found` error message.

To tell MS-DOS to cancel all APPEND paths, type the following and press **Enter**:

```
APPEND;
```

ASSIGN *External*

Purpose

Assigns a new letter to a drive specification.

Format

`[d:][path]ASSIGN [d1=d2...]`

Description

Use ASSIGN to assign different letters to your disk drives. This can be helpful when an application program recognizes only drives A and B and does not allow you to specify another drive such as C. Entering this command lets you use other drives.

In the ASSIGN format, *d1* is the disk drive to receive the new letter and *d2* is the new letter. Both drives must exist; that is, they must be a diskette drive, a hard disk drive, or a device driver such as VDISK.

To cancel the assignment, type ASSIGN without parameters. This resets all drive letters to their original drive assignments.

Cautions

Use ASSIGN only when necessary. When you are done, cancel the assignment. Do not use ASSIGN during normal use of MS-DOS.

Do not use the ASSIGN command with BACKUP, JOIN, LABEL, PRINT, SUBST, or RESTORE. If you do, you could destroy data because ASSIGN hides the true device type. DISKCOMP, DISKCOPY, and FORMAT ignore any reassignments made with ASSIGN.

Examples

To make all references to drive B act upon drive C, type the following and press **Enter** (do *not* type colons):

```
ASSIGN B=C
```

Now, if you type `DIR B:`, you see the list of files on drive C.

To make all references to drives A and B act upon drive C, type the following and press **Enter**:

```
ASSIGN A=C B=C
```

To reset all drives to their default assignments, type the following and press **Enter**:

```
ASSIGN
```

ATTRIB

External

Purpose

Sets, displays, or removes the read-only attribute or archive bit for a file or group of files.

Format

`[d:][path]ATTRIB [±R][±A] [d:][path]filename[/S]`

Description

Use the ATTRIB command to give you an extra level of protection for important data files that you store on disk. Using ATTRIB, you can set a file's attribute to read-only. This allows the file to be read but not altered or deleted.

You can also use ATTRIB to set a file's archive bit on or off. The archive bit tells MS-DOS which files have been altered so commands such as BACKUP and XCOPY know which files to copy.

Note
An easier way to use the ATTRIB command is through the MENU utility. See Chapter 5 for instructions.

`ATTRIB +R` sets the read attribute of the specified file to read-only. `ATTRIB –R` removes the read-only attribute, allowing you to alter or delete the file.

`ATTRIB +A` sets the archive bit of the specified file so XCOPY and BACKUP know to copy the file. `ATTRIB –A` turns off the archive bit, indicating the file has not been altered and does not need to be archived. After you copy a file with XCOPY or BACKUP, the file's archive bit is turned off.

You can use ATTRIB with any logical combination of +R/–R or +A/–A.

If you use ATTRIB without an R or A option, the screen displays the attribute that is currently set for the specified file.

You can use wildcards (* and ?) in the filename to specify groups of files.

Use the /S switch when you want ATTRIB to process every matching file in a specified directory and all of its subdirectories.

Caution

If you use REPLACE /R, you can overwrite a read-only file with the source file. If you use XCOPY /M to copy a file whose archive bit is set, XCOPY turns off the archive bit of the *source* file. See the REPLACE and XCOPY commands in this chapter.

Examples

To set the read attribute of a file named BETTY to read-only, type the following and press **Enter**:

```
ATTRIB +R BETTY
```

To remove the read-only attribute of all files on drive A, type the following and press **Enter**:

```
ATTRIB -R A:*.*
```

To set the archive bit of all files on drive C in the SALES directory, type the following and press **Enter**:

```
ATTRIB +A C:\SALES\*.*/S
```

The /S sets the archive bit of all files in any subdirectories of the SALES directory.

To turn off the archive bit of a file named MEMO.TXT, type the following and press **Enter**:

```
ATTRIB -A MEMO.TXT
```

In the following example, a file called EPSON.TXT on drive C has been set to read-only (+R) and has had its archive bit set (+A). To display the current attributes of this file, type the following and press **Enter**:

```
ATTRIB EPSON.TXT
```

The screen displays:

```
R A            C:\EPSON.TXT
```

The letters R and A appear only when the applicable attribute is set. If neither attribute is set, using the ATTRIB command with the EPSON.TXT file, the screen displays the following:

```
C:\EPSON.TXT
```

BACKUP *External*

Purpose

Copies (backs up) one or more files from one disk drive to another so you can restore them using the RESTORE command.

Format

```
[d:][path]BACKUP d1:[path][filename] d2: [/A]
                 [/D:date][/T:time][/F:size]
                 [/L[:[d:][path]filename]][/M][/S]
```

Description

Use BACKUP to back up one or more files from one disk drive to another. You can back up files in any of the following ways:

- Hard disk to diskette(s)
- Hard disk to hard disk
- Diskette to diskette
- Diskette to hard disk.

BACKUP is most commonly used to back up hard disk files.

> **Note**
> An easier way to use the BACKUP command is through the MENU utility. See Chapter 5 for instructions.

The disk that contains the files you want to back up is called the *source* disk. The *target* disk is the disk being used to store all the backed up files.

In the command format above, *d1:* represents the source and *d2:* represents the target.

XCOPY and BACKUP offer similar file-selection options. The main differences between the two commands are as follows:

- XCOPY creates files on the target diskette that all MS-DOS commands can access. To access files created by BACKUP, you must use the RESTORE command.
- BACKUP can copy source files to a series of target diskettes. If a source file is too large to fit in the remaining space on a target diskette, BACKUP copies as much as it can to the current target diskette, then prompts you to insert another diskette so you can continue the copy. When XCOPY runs out of room on the target diskette, it issues an error message and stops copying.
- BACKUP automatically formats unformatted diskettes.
- BACKUP destroys any pre-existing data in the root directory of the target diskettes, unless you include the /A switch.

Because of these differences, you may prefer to use BACKUP when you have many files to archive. XCOPY may be more convenient when you need to copy only a few files.

You can use the following switches with BACKUP:

Switch	*Function*
/A	Adds files to the target disk if it already contains backed-up files. Use /A to copy files from the source disk which are not already stored on the target.
/D:*date*	Backs up files that have been modified on or after the specified date. The date format depends on the COUNTRY code selected with the COUNTRY command. Use a period or a dash to separate the day, month, and year.
/F:*size*	Formats the target diskette if it is not already formatted. The value of *size* may be 160, 180, 320, 360, 720, 1.2, or 1.44.

Switch	*Function*
/T:*time*	Backs up files that have been modified on or after the specified time on the specified date. If you do not specify a date, all files after the specified time on any date are backed up. The time format depends on the COUNTRY code selected with the COUNTRY command. Use a period or dash to separate the hours, minutes, and seconds.
/L:[*d:*][*path*] *filename*	Makes a backup log file listing the filenames. If you do not specify a filename, MS-DOS creates the log file BACKUP.LOG on the source disk. MS-DOS cannot create the log file on the target disk.
/M	Backs up files that have been modified since the last BACKUP.
/S	Backs up all files in the subdirectories of the specified directory in addition to the files in the specified or current directory. Without the /S switch, only files in the current directory are backed up.

If you do not specify a drive and path, the log file is stored in the root directory of the source drive. If a log file already exists, the new log information is added to the end of that file.

The first record in the log file specifies the date and time of the backup. Each subsequent record contains the backup disk number and complete pathname and filename for each file backed up.

If you do not include the /A switch, BACKUP destroys all files stored in the root directory of the target disk.

When you use the /A switch to add files to diskettes, be sure to back up the additional files on the last diskette in the series. Otherwise, MS-DOS displays an error message.

You can use wildcards to back up groups of similarly named files including files from different subdirectories.

BACKUP displays the name of each file as it is backed up. After BACKUP fills up a diskette, it prompts you to insert another diskette into the drive.

Label and number each backup diskette in consecutive order. When you RESTORE these files, you are prompted to insert the diskettes in order.

If the target is a hard disk, backup files are stored in a subdirectory called \BACKUP. If the target is a diskette, backup files are stored in the root directory.

If you want to stop the BACKUP procedure after it has started, press **Ctrl Break**. MS-DOS cancels the command and displays the system prompt on the screen.

If you are sharing files, you can back up only those files that you normally can access. If you try to back up a file that you are not permitted to access, the following message appears:

```
PATHNAME\FILENAME.EXT
Not able to backup file
```

BACKUP automatically formats any unformatted target diskettes, provided the file FORMAT.COM is in the current directory or in any directory specified by the APPEND or command.

When backing up to a diskette drive, the target diskettes are formatted to the maximum capacity of the drive. Therefore, do not use 360KB diskettes in a 1.2MB drive or 720KB diskettes in a 1.44MB drive.

BACKUP sets the following exit codes, which you can test in a batch file using the IF ERRORLEVEL command:

0 Normal completion of BACKUP procedure.

1 No files were found to be backed up.

2 Some files were not backed up because of file-sharing conflicts.

3 Terminated by user (**Ctrl C** or **Ctrl Break**).

4 Terminated because of an error.

Cautions

Do not use BACKUP while the APPEND /X, ASSIGN, JOIN, or SUBST commands are in effect.

If the source is a diskette, do not write-protect it. BACKUP must be able to write to the source diskette, to clear the archive bit of files it backs up.

BACKUP does not copy the hidden MS-DOS system files or COMMAND.COM.

Examples

To back up all files on the hard disk (drive C) to the diskette(s) in drive A, type the following and press **Enter**:

```
BACKUP C:*.* A:/S
```

This command backs up all files in the current directory and all of its subdirectories on drive C.

To back up the file FILEIT.DOC from drive B to drive A, type the following and press **Enter**:

```
BACKUP B:FILEIT.DOC A:
```

To back up all files on drive C that have been modified since March 15, 1989, type the following and press **Enter**:

```
BACKUP C: A:/D:3-15-89/L
```

Using the /L switch creates a log file named BACKUP.LOG in the root directory of drive C; this log file lists all the backup files.

To *add* files from drive B in the WORDPROC directory (and any of its subdirectories) to files already backed up on the diskette in drive A, type the following and press **Enter**:

```
BACKUP B:\WORDPROC A:/A/S
```

BREAK

Internal

Purpose

Enables or disables the checking for the **Ctrl C** or **Ctrl Break** function.

Format

```
BREAK [ON]
```

or

```
BREAK [OFF]
```

Description

Use BREAK to turn on or off the MS-DOS function (**Ctrl C**) that terminates the current command. Normally, MS-DOS checks for **Ctrl C** only while it is reading from the keyboard, or writing to the monitor or printer. The standard setting is BREAK OFF.

When BREAK is ON, MS-DOS checks for **Ctrl C** whenever a program requests MS-DOS to perform any function, such as a disk read and/or write.

If you are running an application program that uses **Ctrl C** itself, type `BREAK OFF` to turn off the MS-DOS **Ctrl C** function. Then, when you press **Ctrl C**, you affect your application program and not the operating system.

When you have finished running your application program and are using MS-DOS, type `BREAK ON` to turn the **Ctrl C** function on again.

To display the current setting of BREAK, type `BREAK` only without ON or OFF. If BREAK is OFF, the screen displays:

```
BREAK is off
```

You can also turn on the abort current command function by including `BREAK=ON` in your CONFIG.SYS file. See Chapter 6.

CHCP

Internal

Purpose

Displays or changes the current code page (character set).

Format

`CHCP [nnn]`

Description

Use the CHCP command to display the current code page or to select a different one. Unlike the MODE command, which lets you change the code page for a particular device, CHCP changes the code page for the entire system.

The *nnn* represents the number of the code page you want to use:

Number	*Code Page*
437	United States—English
850	Multilingual
860	Portuguese
863	Canada—French
865	Nordic

If you do not specify a number, the screen displays the number of the current code page for the system.

Before giving the CHCP command, you must do the following:

- Load the NLSFUNC command to provide national language support.
- Use the DEVICE command in the CONFIG.SYS file to enable the device drivers affected by code page switching.
- Use the MODE command with the Codepage Prepare option to prepare the selected code page for the device or devices.
- Be sure that the COUNTRY.SYS file, which contains country information, is accessible to the CHCP command. If it is not in the current directory, you can use the COUNTRY= statement in the CONFIG.SYS file or the NLSFUNC command to tell MS-DOS where to find COUNTRY.SYS.

Be sure to read Chapter 9, Code Page Switching, for information on code page switching.

If you are running an application program and then give the CHCP command, the program continues to use the original code page. If you start a program after giving the command, it uses the new code page.

If the code page you select is not prepared for the system, the screen displays a message similar to this:

```
Code page 863 not prepared for system
```

If the device (a monitor or printer, for example) is not prepared for a code page, the screen displays a message similar to the following:

```
Code page 863 not prepared for all devices
```

The CHCP command changes the active code page even if the specified code page is not prepared for a device. If you want to return to the original code page, you need to reenter its number in the CHCP command.

Examples

To display the current code page, type the following and press **Enter**:

```
CHCP
```

If the current code page is the US, the screen displays the following:

```
Active code page: 437
```

To select the multilingual code page, type the following and press **Enter**:

```
CHCP 850
```

CHDIR

Internal

Purpose

Changes or displays the current directory.

Format

```
CHDIR [d:][path]
```

or

```
CD [d:][path]
```

Description

Use CHDIR to change the current directory to another, or to display the current directory path of a drive. You can type either CHDIR or CD.

The directory you are working in is called your current directory. MS-DOS remembers the current directory for each drive. Therefore, when you change drives without specifying a directory, MS-DOS goes to the directory you were using when you last used that drive.

Unless you set a path, MS-DOS looks in the current directory (on the current drive) to find files. You can set a path of up to 63 characters starting from the root directory. Indicate the root directory with a leading backslash (\).

A single period (.) represents the current directory. Two periods (..) represent the parent directory of the current directory.

Caution

The current directory may be hidden by ASSIGN, JOIN, and SUBST.

Examples

If your current directory is \LEDGER\SALES\WEST and you want to change to the \WORDPROC\BUSINESS directory, type the following and press **Enter**:

```
CD \WORDPROC\BUSINESS
```

To display the current directory, type only CHDIR and press **Enter**. The screen displays your current directory. For example, if your current directory is \WORDPROC\BUSINESS, the screen displays the following:

```
B:\WORDPROC\BUSINESS
```

To change to the parent directory of the current directory (the next directory *up* the tree), type the following and press **Enter**:

```
CHDIR ..
```

To change to the root directory, regardless of what your current directory is, type the following and press **Enter**:

```
CHDIR \
```

CHKDSK

External

Purpose

Checks the directories, files, and file allocation table on the specified disk and produces a disk and memory status report.

Format

```
[d:][path]CHKDSK [d:][path][filename][/F][/V]
```

Description

Use CHKDSK to inspect the status of a diskette or hard disk and report the number of files, the amount of total disk space and space still available for use, and the amount of internal memory available.

Run this command periodically to check for errors. If CHKDSK finds any errors in the disk's directory or file allocation table, it reports them and, optionally, corrects them.

> **Note**
> An easier way to use the CHKDSK command is through the MENU utility. See Chapter 5 for instructions.

If you specify a filename, CHKDSK reports the number of non-contiguous areas occupied by the file (or all the files if you specify *.*).

CHKDSK assumes that the disk to be checked is in the specified drive. You are not prompted to insert the disk.

You can use two switches with CHKDSK:

Switch	*Function*
/F	Selects fix mode; CHKDSK attempts to correct any errors
/V	Selects verify mode; CHKDSK displays filenames, directories, and messages while it is running.

CHKDSK *d*: > *filename* redirects the output from CHKDSK to a file and sends status information and errors to the specified filename. If you also include the /V switch, CHKDSK outputs the filenames and messages to the file.

Do not use the /F switch if you redirect the output to the same disk drive. The addition of a new filename to the directory automatically causes an error.

Cautions

Do not use CHKDSK on drives specified in an ASSIGN, JOIN, or SUBST command.

This command may not work in a network environment.

Examples

To check the diskette in drive B, type the following and press **Enter**:

```
CHKDSK B:
```

If CHKDSK does not find any errors, you see a status report similar to this:

```
Volume SALES created Ø5-14-1989 9:53a
Volume Serial Number is 16ØC-4Ø86

1457664 bytes total disk space
   6656 bytes in 12 directories
1349632 bytes in 38 user files
 1Ø1376 bytes available on disk

    512 bytes in each allocation unit
   2847 total allocation units on disk
    198 available allocation units on disk

 656384 total bytes memory
 489264 bytes free
```

The volume serial number is a number assigned by the MS-DOS 4.01 FORMAT command. Diskettes formatted using earlier versions of MS-DOS may not have a volume serial number.

Allocation units, or *clusters*, are the smallest increment of disk space MS-DOS uses in writing files. The information in this CHKDSK report tells you that MS-DOS uses a cluster size of 512 bytes for a 1.44MB diskette; this means that the smallest file you can create on such a diskette uses 512 bytes of disk space. (The directory listing may indicate that the file is smaller than 512 bytes, but it still uses 512 bytes of disk storage.)

To check the disk in drive A and display all the files and their paths, type the following and press **Enter**:

```
CHKDSK A:/V
```

You see a status report like the one above plus a display of all the files and their paths.

To determine the number of non-contiguous areas, if any, occupied by the file BUSINESS in your current directory, type the following and press **Enter**:

```
CHKDSK BUSINESS
```

If the file contains two non-contiguous areas, CHKDSK displays the following message:

```
B:\BUSINESS Contains 2 non-contiguous blocks
```

CLS

Internal

Purpose

Clears the monitor screen.

Format

```
CLS
```

Description

Use CLS to clear your monitor screen and display the command prompt in the upper left corner of the screen.

If you have loaded ANSI.SYS and have set screen attributes, the CLS command does not change the attributes.

Example

On a screen filled with text, type the following and press **Enter**:

```
CLS
```

Your screen clears and displays only the command prompt.

COMMAND *External*

Purpose

Loads a secondary command processor.

Format

```
[d:][path]COMMAND [d:][path][CTTY-dev]
                  [/C command][/E:nnnnn][/P]
```

Description

If you have loaded a top-level command processor in place of COMMAND.COM, use COMMAND to load a secondary command processor.

When you boot MS-DOS, COMMAND automatically loads in parts. One part remains in memory permanently. Another part is transient.

The space in memory occupied by the transient part is not reserved, and application programs can overwrite it. Therefore, programs sometimes have to reload this transient part.

If COMMAND.COM is not in the specified directory, MS-DOS searches the path in your environment. COMMAND.COM loads the transient portion from the file specified in COMSPEC= as part of its initialization.

If you enter COMMAND without any parameters, MS-DOS loads a new copy of the command processor. This new copy inherits the environment known to the previous level of the command processor.

If you use the SET command to change the environment of the secondary command processor, only the secondary copy recognizes the change. If you exit back to the primary command processor, it returns to the environment of the primary command processor that was in effect before the secondary copy existed.

If you specify a drive and/or path following the keyword COMMAND, a new COMSPEC= is created, and the secondary processor does not inherit the environment of the primary command processor.

You can use the CTTY-*dev* parameter to select a different device (such as AUX) for input and output. See the description of CTTY later in this chapter for more information.

You can use the following switches with COMMAND:

Switch	*Function*
/C *command*	Allows you to enter a command and then automatically exit back to the primary command processor after the command is completed. COMMAND ignores /P is if you use /C. You send *command* to the command processor. The command is interpreted and executed as if you had typed it at the system prompt.
/E:*nnnnn*	Specifies a base 10 integer and indicates the number of bytes to set the environment size. This number must be between 160 and 32768. It is rounded up to the nearest paragraph boundary (a multiple of 16 bytes).
/P	Causes the copy of the new command processor to become permanent in memory and remain active until you turn off or reboot the system. The second command processor does not return to the primary command processor. You must restart MS-DOS to remove the second command processor. If you use /C, COMMAND ignores /P. Without the /P switch, the secondary processor remains resident until you use the EXIT command. If you use the /P switch, the EXIT command does not work.

Examples

To load a secondary command processor and have it execute the DIR command (for drive B), and then exit back to the primary command processor, type the following and press **Enter**:

```
COMMAND /C DIR B:
```

To load a new command processor with an environment size of 100 paragraphs and cause it to become permanent in memory, type the following and press **Enter**:

```
COMMAND /E:16ØØ /P
```

COMP

External

Purpose

Compares the contents of one file or group of files to another.

Format

```
[d:][path]COMP [d:][path][filename]
               [d:][path][filename]
```

Description

The COMP command compares single files or groups of files by examining each byte of information. The files can be in the same directory or in different directories on the same or on different disks.

> **Note**
> Use COMP if you just want to know if the files are the same. If you want more detailed information about the files, you may prefer to use the FC command.

If you do not enter any parameters in the command, COMP displays prompts on the screen for information about the two files to be compared.

If the files you want to compare are not on the same disk as COMP, type the COMP command without any parameters. The program then prompts you for the filenames, and you can insert the proper diskette.

If the two files to be compared are not the same size, COMP displays the following prompt and stops the comparison:

```
Files are different sizes

Compare more files (Y/N)?
```

You cannot compare two files that are not the same size.

When the program finds bytes that do not match, it displays a message such as the following:

```
Compare error at OFFSET 1Ø
File 1=48

File 2=71
```

This error message gives the hexadecimal location (the offset) and the value of bytes that do not match. After finding 10 unequal comparisons, COMP quits and displays this message:

```
1Ø Mismatches - ending compare
```

If the files are identical, COMP displays this message:

```
Files compare ok
```

If you are comparing more than two files, COMP compares the next pair matching the filename specification. When COMP finishes comparing all the specified files, it displays this message:

```
Compare more files (Y/N)?
```

Press **Y** if you want to compare more files; the program prompts you to enter the filenames. Press **N** to exit the program.

For all comparisons, COMP looks at the last byte of the files to verify that they contain a valid end-of-file mark (Ctrl Z). If it does not find this mark for a file, COMP displays this message:

```
EOF mark not found
```

Examples

To compare all files with the extension .TXT on drive A with files on drive B with the same names, type the following and press **Enter**:

```
COMP A:*.TXT B:
```

To compare a file named JUNEMEMO with a file named JULYMEMO in the WORDPROC directory on drive C, type the following and press **Enter**:

```
COMP C:\WORDPROC\JUNEMEMO  C:\WORDPROC
     \JULYMEMO
```

COPY

Internal

Purpose

Copies one or more files to the same or to another disk and optionally renames or merges them.

Format

```
COPY [d:][path]filename [d:][path][filename]
     [/A][/B][/V]
```

or

```
COPY [d:][path]filename +
     [[d:][path]filename[...]]
     [d:][path]filename[/A][/B][/V]
```

or

```
COPY [d:][path]filename +,,
```

Description

There are several ways to use the COPY command to copy files. You can do any of the following:

- Copy individual files from one disk to another or to the same disk
- Copy a group of files using wildcard characters
- Copy one or more files under new names
- Copy files to any system device
- Combine or merge files into one file with each file starting at the end of the preceding file.

If you want to copy files including subdirectories, use either the BACKUP or XCOPY command. COPY only copies files from the current or specified directory.

Copying Files

In the first COPY format above, the first file specification is the source file and the second one is the destination, or target, file. If the source and target files are in the current directory, you need not specify a complete path; simply specify the filenames.

The source file and the target file must either be on different directories or have different names. If you try to copy a file without specifying a different directory or a different filename for the target file, the copy stops and the screen displays an error message.

`COPY [d:][path]filename d:` copies the original file to the drive you designate and retains the original filename.

`COPY [d:][path]filename1 filename2` copies the original file to a file on the default drive and directory, assigning the new filename you specify.

`COPY [d:][path]filename1 d:filename2` copies the original file to the drive and filename you specify.

`COPY [d:][path]filename device` copies the file to a device, such as the printer. For example, to print the file JUNEMEMO, which is stored in the \MEMOS subdirectory of drive C, you could type the following and press **Enter**:

```
COPY C:\MEMOS\JUNEMEMO PRN
```

You can use the wildcard characters to copy groups of similar files. The command `COPY *.TXT B:` copies all files with the TXT extension from the default drive to the diskette in drive B.

You can use three switches with COPY:

Switch	*Function*
/A	Selects ASCII (text) mode
/B	Selects Binary (program) mode
/V	Selects Verify mode

The /A and /B switches apply to the file specification immediately preceding them and to all remaining file specifications on the command line, until you enter another /A or /B switch. You can enter the /A and /B switches before the first filename.

Use /V to select Verify mode. MS-DOS then verifies that the sectors are recorded properly during the COPY process. This is the same check that results from the VERIFY ON command. However, /V is effective only during the specified COPY command.

Use /A to select ASCII mode, which indicates that you are processing ASCII, or text, files. When you use /A with a source file specification, the copy includes data but does not include the first end-of-file mark (Ctrl Z). The remainder of the file does not copy. When you specify a destination file, the /A switch adds an end-of-file (Ctrl Z) character as the last character of the file.

Use /B to select Binary mode, which indicates that you are processing binary, or program, files. When you use /B with a source file specification, the copy includes the entire length of the file, as indicated by the file-size entry in the directory. Any Ctrl Z characters are included in the copy. When you specify a destination file, the /B switch includes the entire file, including an ending Ctrl Z character.

Merging Files

Use the second COPY format to merge files. Type any number of filenames separated by plus signs (+). For example, the following command combines three files into a new file called *newfile*:

```
COPY file1+file2+file3 newfile
```

You can use wildcards to combine several files. The command `COPY *.ext newfile` combines all files with the specified extension into a file called *newfile*.

You can also use wildcards and file extensions to combine the files and create additional files.

Updating Without Copying

Use the third COPY format to update the date and time of the file without copying it anywhere:

```
COPY filename +,,
```

Caution

Do not try to merge files if one of the source filenames has the same extension as the target. For example, if ALL.LST already exists, do not use the command `COPY *.LST ALL.LST`. MS-DOS overwrites the original ALL.LST on the target diskette and detects this error only when the copy is complete.

Examples

To copy all files from drive A to drive B, type the following and press **Enter**:

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

To copy all .TXT files in the specified path to the root directory of the hard disk, type the following and press **Enter**:

```
COPY A:\WORDPROC\BUSINESS\INTERNAL\*.TXT C:\
```

To copy the file 620.TXT (in the \ELECMAIL\IN directory) to the \WORDPROC\PERSONAL directory, type the following and press **Enter**:

```
COPY \ELECMAIL\IN\62Ø.TXT \WORDPROC
     \PERSONAL
```

To print the textfile EPSON.TXT on your printer, type the following and press **Enter**:

```
COPY A:EPSON.TXT PRN
```

To update the date and time of the textfile STATUS.TXT without copying it, type the following and press **Enter**:

```
COPY STATUS.TXT +,,
```

CTTY

Internal

Purpose

Changes the input/output device.

Format

`CTTY` *`device`*

Description

Use CTTY to change the device you are using. You can direct input and output to the following devices:

Device	*Description*
AUX	An auxiliary device, such as a printer or disk drive, connected to the RS-232C serial port
CON	The console, which represents input from the keyboard and output to the monitor screen
PRN	A printer connected via the printer port
NUL	A nonfunctional device (useful during program development)
LPT1	Line printer 1, the same as PRN
COM1	Communications port 1, which is your serial port, the same as AUX

You can specify LPT2, LPT3, and COM2, COM3, and COM4 as long as that device is installed in or connected to the system.

You can use printers as output devices only. You cannot use a printer for input.

Caution

Once you use CTTY, input is no longer accepted from the keyboard; you have to reset the computer. Therefore, if you use the CTTY command in a batch file, remember to include CTTY CON at the end of your batch file to return control to the keyboard.

Examples

To direct all command output from the current device (the keyboard), to the AUX port, type the following and press **Enter**:

```
CTTY AUX
```

To direct all command input and output back to the keyboard and monitor, type the following and press **Enter**:

```
CTTY CON
```

Here is an example of a batch file you can create, directing output to the printer and then back to the screen:

```
REM This is printed on the screen.
CTTY PRN
REM This is output to the printer.
CTTY CON
REM This is output to the screen again.
```

With CTTY, you can use your computer as a terminal for another computer. Connect the two computers via the RS-232C serial port (device name AUX), and make sure the signal parameters match properly with the MODE command. Then use the command CTTY AUX to input and output all commands from the other computer. The command CTTY CON, entered from the other computer, moves all command output back to the keyboard of your computer.

DATE

Internal

Purpose

Displays and/or sets the system date.

Format

```
DATE [mm-dd-yy]
```

or

```
DATE [dd-mm-yy]
```

or

```
DATE [yy-mm-dd]
```

Description

You can use the DATE command to display the current date or set a new date for the system. This command updates the CMOS date if your computer contains a real-time clock.

Whenever you create a new file or alter an existing one, MS-DOS records the date and time with that file in its directory.

You can display the current system date by typing DATE and pressing **Enter**. If you are using the United States country code, you see a message similar to this:

```
Current date is Tue 3-15-89
Enter new date (mm-dd-yy):
```

For other countries, the month and day are reversed or the year may come first. (The way MS-DOS displays the current date depends on which COUNTRY command you include in the CONFIG.SYS file. United States is the default setting.)

Press **Enter** if you do not want to change the current date. Or, enter a new date and then press **Enter**. MS-DOS returns to the command prompt.

Enter the date using numbers only. You need not enter the day of week. The numbers must be in the following ranges:

- *mm* must be from 1 to 12
- *dd* must be from 1 to 31
- *yy* must be from 80 to 99 or from 1980 to 2099.

You can separate the month, day, and year entries with hyphens (-), slashes (/), or periods (.).

MS-DOS updates months and years correctly, whether the month has 31, 30, 29, or 28 days. MS-DOS also accommodates leap years.

Examples

To set the date for April 20, 1989, you can type any one of the following (using the U.S. format):

```
4/2Ø/89          Ø4/2Ø/89
4/2Ø/1989        Ø4/2Ø/1989
4-2Ø-89          Ø4-2Ø-89
4-2Ø-1989        Ø4-2Ø-1989
```

If you select the United Kingdom with the `COUNTRY=044` command in your CONFIG.SYS file, you type the date in one of these formats:

```
2Ø/4/89
2Ø-4-1989
2Ø.4.89
```

DEL

Internal

Purpose

Deletes a file or group of files.

Format

DEL [*d*:][*path*]*filename* [*/P*]

Description

Use DEL to delete one or more files. (You can type ERASE instead of DEL.)

DEL *filename* deletes the specified file from the current directory.

You can use the wildcard characters (? and *) to delete groups of similar files. The command DEL *.*ext* deletes all files with the specified extension from the current directory on the default drive.

You can delete all files from a directory or disk by using wildcard characters for the filename and extension. Type DEL *.* and you see this prompt:

```
All files in directory will be deleted!
Are you sure (Y/N)?
```

Press **Y** and **Enter** to delete all files in the current directory. Press **N** and **Enter** to cancel the DEL command.

If you enter a pathname but no filename in the command, MS-DOS assumes you want to delete all the files in the specified path. Since this is the same as entering *.*, you see the same prompt and have an opportunity to cancel the command.

If you include the /P switch, MS-DOS displays a confirmation prompt for each file you specify. For example, if you enter the command DEL C:\BACK.LOG /P, you see the following prompt:

```
C:\BACK.LOG, Delete (Y/N)?
```

Press **Y** and **Enter** if you want to delete the file, or **N** and **Enter** if you do not want to delete it.

DEL and ERASE delete files only. You cannot delete a directory with these commands. To delete a directory, use the RMDIR command.

You cannot delete a file that is marked as read-only.

Cautions

Be sure you no longer need a file before you delete it.

Be careful when using DEL after using ASSIGN, JOIN, or SUBST.

Examples

To delete the file SEEME from the current directory on the default drive, type the following and press **Enter**:

```
DEL SEEME
```

To delete all files on drive B with a .BAK extension, type the following and press **Enter**:

```
DEL B:*.BAK
```

To delete a file named JUNSALES from the subdirectory EXTERNAL (in the \WORDPROC\BUSINESS\EXTERNAL directory) on drive C, type the following and press **Enter**:

```
DEL C:\WORDPROC\BUSINESS\EXTERNAL\JUNSALES
```

To delete the files on drive A selectively (that is, to confirm before deleting each file), type the following and press **Enter**:

```
DEL A:*.* /P
```

DIR

Internal

Purpose

Lists the files in a directory.

Format

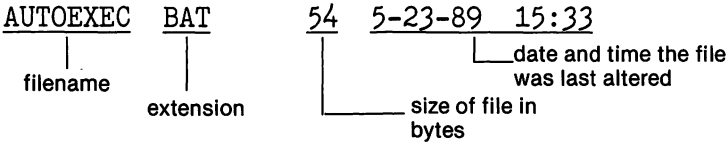
```
DIR [d:][path][filename][/P][/W]
```

Description

Use DIR to display a list of the files in a directory. You can display the entire directory or a portion of the directory.

The information DIR displays includes the filename and extension, the size (in bytes, in decimal notation), and the date and time the file was created or last modified. The directory listing also displays the volume label (if one exists), the volume serial number (if one exists), and the amount of free space left on the disk.

The following is an example of how a DIR command displays a file named AUTOEXEC.BAT:



Note that the format for date and time is dependent on the COUNTRY command set in your CONFIG.SYS file. See Chapter 6 for information on the CONFIG.SYS file.

Directory names are identified by <DIR> in the file size field.

You can use the wildcard characters ? and * in the filename. For example, the following DIR commands are equivalent:

Command	*Equivalent*
DIR	DIR *.*
DIR LETTER	DIR LETTER.*
DIR .INV	DIR *.INV

You can specify two switches with DIR:

Switch	*Function*
/P	Selects Page mode. The directory display pauses when the screen is full; press any key to display the next screen.
/W	Selects Wide display. The screen displays five filenames per line but no other file information. Use this switch for 80-column displays only.

Examples

To list all files and any subdirectories in the current directory on the default drive one screen at a time, type the following and press **Enter**:

```
DIR /P
```

To list all files and directories on drive B, type the following and press **Enter**:

```
DIR B:
```

To list all the files in the SALES directory on the default drive, type the following and press **Enter**:

```
DIR \SALES
```

To list a file named DEVST.MT on drive A, type the following and press **Enter**:

```
DIR A:DEVST.MT
```

To list all files with a .COM extension in wide display, type the following and press **Enter**:

```
DIR *.COM /W
```

To list all files in the current directory whose names begin with MEMO followed by a single character that varies, type the following and press **Enter**:

```
DIR MEMO?
```

DISKCOMP *External*

Purpose

Compares the contents of two diskettes.

Format

```
[d:][path]DISKCOMP [d1:[d2:]][/1][/8]
```

Description

Use DISKCOMP to compare sectors on one diskette with those on another diskette. An examination by sector tells whether the diskettes are exactly alike. Use DISKCOMP with diskettes only. You cannot compare a diskette to a hard disk.

> **Note**
> Use this command to compare entire diskettes. To compare files, use the COMP or FC command. You can also compare diskettes using the Epson MENU utility. See Chapter 5 for instructions.

The order in which you specify drives sets the order for reading and comparing. The diskette in the first drive you specify is read, then DISKCOMP compares the contents of the diskette in the second drive to those of the first. DISKCOMP then reports at which points the second diskette fails to match the first.

For example, `DISKCOMP A: B:` compares the diskette in drive A (the first diskette) with the diskette in drive B (the second diskette), displaying on the screen any difference on the second diskette.

If you are logged onto a hard disk drive, you must include both drive descriptors in the command, even if you have only one diskette drive.

If you are logged onto a diskette drive and specify only one drive, DISKCOMP reads the diskette in the specified drive first and then compares it to the diskette in the default drive. For example, if you

are logged onto drive A, type `DISKCOMP B:` to compare the diskette in drive B to the diskette in drive A and show possible differences between the two diskettes.

Note
The diskettes you are comparing must be of the same type. You cannot, for example, compare a 360KB diskette to a 1.2MB diskette, or a 720KB diskette to a 360KB diskette.

If you do not include any drive specifications, DISKCOMP performs its comparison using only the default drive, prompting you to insert diskettes one at a time in that drive. DISKCOMP compares the second diskette you insert to the image read from the first and reports the differences. Use this procedure if your system has only one diskette drive.

You can use two switches with DISKCOMP:

Switch	*Function*
/1	Compares only one side of each diskette; use for single-sided diskettes only.
/8	Compares only the first eight sectors per track on your diskettes; use for diskettes originally formatted in eight sectors per track.

Use DISKCOMP any time you want to know the similarity of two diskettes. You can also use it to check the success of a DISKCOPY operation.

DISKCOMP automatically determines the number of sides and sectors per track to compare. The first drive specified determines the type of comparison that is performed.

Batch files using DISKCOPY can also use DISKCOMP. You can make frequent use of this comparison command in other batch files dealing with diskette housekeeping. Your own operating needs determine just how you use DISKCOMP.

DISKCOMP ignores any differences in volume serial number.

Note
If you compare a diskette with a backup version you made using the COPY command, you may receive an error message even though the two diskettes contain the exact same files. This happens because even though COPY copies the files exactly, they may not be stored in the same sectors on the two diskettes. The diskettes are identical if you use the DISKCOPY command to make the backup copy.

Cautions

If you have used ASSIGN to reassign disk drive descriptors, DISKCOMP may not work properly. Cancel the assignments and then run DISKCOMP.

Do not use DISKCOMP on any drives that are being used in a JOIN or SUBST command.

This command may not work in a network environment.

Examples

To compare the diskette in drive B to the one in drive A, type the following and press **Enter**:

```
DISKCOMP A: B:
```

After DISKCOMP loads, it displays the following screen prompts:

```
Insert FIRST diskette in drive A:
Insert SECOND diskette in drive B:
Press any key to continue...
```

Put your first diskette in drive A and your second one in drive B. Press any key to continue.

DISKCOMP reads the format of the particular diskette being checked. For a 360KB diskette, the screen displays the following:

```
Comparing 40 tracks
9 sectors per track, 2 side(s)
```

If DISKCOMP detects any differences, the screen reports each of them by track and side. For example:

```
Compare error on
side 1, track 4
```

If DISKCOMP does not find any differences between your diskettes, it reports:

```
Compare OK
```

When the comparison is complete, your screen displays:

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

Press **Y** to repeat the procedure, or **N** to exit to the command prompt. If you do repeat the procedure, remember that the same drive order applies in your next comparison.

DISKCOPY *External*

Purpose

Copies the contents of one diskette to another.

Format

`[d:][path]DISKCOPY [d1:[d2:]][/1]`

Description

Use DISKCOPY to copy the contents of the diskette in the source drive to the one in the destination, or target, drive. If you enter `DISKCOPY A: B:`, the source drive is A, and the target drive is B.

Use DISKCOPY to copy diskettes only. You cannot use it to copy a diskette to a hard disk or any portion of the hard disk to a diskette.

Use this command to copy entire diskettes. To copy files, use the COPY command.

> **Note**
> An easier way to use the DISKCOPY command is through the MENU utility. See Chapter 5 for instructions.

You can specify the same drive as your source and target drive, or you can specify different drives. When the source and target drives are the same, MS-DOS performs a single-drive copy operation. DISKCOPY prompts you to insert the diskettes at the appropriate times and waits for you to press any key before continuing.

If you copy a 360KB diskette in a 1.2MB drive, you may have trouble reading that diskette in a 360KB drive later. Read "Drive and diskette incompatibilities" in Chapter 1 for more information.

When the copy operation begins, you see this message when copying a 1.2MB diskette:

```
Copying 8Ø tracks
9 Sectors/Track, 2 Side(s)
```

DISKCOPY then checks to see if the target diskette is formatted and in the same format as the source diskette. If it is not, DISKCOPY formats the diskette. You see this message:

```
Formatting while copying
```

When the copy is complete, you see this message:

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

Press **Y** to make another copy using the same drives. Press **N** to end the DISKCOPY command.

Notice that the disk drive descriptors are optional as long as the default drive is not the hard disk. If you omit both drives, DISKCOPY performs a single-drive copy operation. If you omit the second disk drive descriptor, the current drive is also the target drive. You cannot give the DISKCOPY command without drive descriptors if you are currently logged onto the hard disk drive.

Usually, DISKCOPY automatically determines the number of sides per diskette to copy, based on the source drive and diskette. For some applications, however, you must use /1 to perform a single-sided DISKCOPY.

If the source diskette has a volume serial number, DISKCOPY assigns a different volume serial number to the target diskette. If the source diskette has no volume serial number, DISKCOPY does not assign a volume serial number to the target diskette.

Cautions

Diskettes become fragmented when you create and delete a lot of files. This is because diskette space is not allocated sequentially. The first free sector found is the next sector allocated, regardless of its location.

A fragmented diskette can cause poor performance due to delays involved in finding, reading, or writing a file. To eliminate fragmentation, use the COPY or XCOPY command—instead of DISKCOPY—to copy your diskette.

If you have reassigned disk drive descriptors with ASSIGN, cancel the assignments before you run DISKCOPY. Otherwise, you could destroy data on your diskettes.

Do not use DISKCOPY on any drives that are being used in a JOIN or SUBST command.

This command may not work in a network environment.

If you copy to a 360KB diskette in 1.2MB drive, you may have trouble reading the diskette in a 360KB drive.

To copy the contents of a 360KB diskette to a 1.2MB high-density diskette or from a 720KB diskette to a 1.44MB diskette, use the COPY command.

Example

To copy two diskettes of the same type with a two-drive system, place the source diskette in drive A and the target diskette in drive B. Then type the following and press **Enter**:

```
DISKCOPY A: B:
```

DOSSHELL *External*

Purpose

Invokes the MS-DOS 4.01 Shell.

Format

`[d:][path]DOSSHELL`

Description

The MS-DOS Shell allows you to enter MS-DOS commands and run programs by selecting menu options. For a description of how to use the MS-DOS Shell, see the MS-DOS 4.01 Shell User's Guide.

If you want to use the Shell for most of your work, include the DOSSHELL command in an AUTOEXEC.BAT file. The Shell will be loaded each time you start or reset your computer.

ERASE

Internal

Purpose

Erases a file or group of files.

Format

ERASE [*d:*][*path*]*filename*[/P]

Description

Use ERASE to erase (delete) one or more files. (You can type DEL instead of ERASE.)

ERASE *filename* erases the specified file from the current directory.

You can use the wildcard characters (? and *) to erase groups of similar files. The command ERASE *.*ext* erases all files with the specified extension from the current directory on the default drive.

You can erase all files from a directory or disk by using wildcard characters for the filename and extension. Type ERASE *.* and you see the prompt:

```
All files in directory will be deleted!
Are you sure (Y/N)?
```

Press **Y** and **Enter** to erase all files in the current directory. Press **N** and **Enter** to cancel the ERASE command.

If you enter a pathname but no filename in the command, MS-DOS assumes you want to erase all the files in the specified path. Because this is the same as entering *.*, you see the same prompt and have an opportunity to cancel the command.

If you include the /P switch, MS-DOS displays a confirmation prompt for each file you specify. For example, if you enter the command ERASE C:\BACK.LOG /P, you see the following prompt:

```
C:\BACK.LOG, Delete (Y/N)?
```

Press **Y** and **Enter** if you want to delete the file, or **N** and **Enter** if you do not want to delete it.

ERASE and DEL erase files only. You cannot erase a directory with these commands. To erase a directory, use the RMDIR command.

You cannot erase a file that is marked as read-only.

Cautions

Be sure you no longer need a file before you erase it.

Be careful when using ERASE after using ASSIGN, JOIN, or SUBST.

Examples

To erase a file named MEMO in the current directory, type the following and press **Enter**:

```
ERASE MEMO
```

To erase all files on drive B with a .BAK extension, type the following and press **Enter**:

```
ERASE B:*.BAK
```

To erase all files from the BUSINESS subdirectory under the WORDPROC directory on drive C, type the following and press **Enter**:

```
ERASE C:\WORDPROC\BUSINESS
```

To erase the files on drive A selectively (that is, to confirm before deleting each file), type the following and press **Enter**:

```
ERASE A:*.* /P
```

EXIT

Internal

Purpose

Exits the command processor and returns to a previous level, if one exists.

Format

```
EXIT
```

Description

Use EXIT to exit the command processor, COMMAND.COM, and return to a previous level.

When you are running an application program, you may want to return to the command processor to perform some function. After performing the function, use the EXIT command to return to the program.

If you have loaded a secondary command processor (with COMMAND), use EXIT to return to the original command processor.

> **Note**
> If you used the /P switch when giving the COMMAND command, EXIT does not work.

You can use this command only if you are running an application program that supports it.

Example

To format a diskette in drive B while in a program that allows you to exit:

1. Leave the program using the appropriate command for the temporary exit.
2. Change the current drive and directory to the one that contains FORMAT.COM (unless you have set the appropriate path).

3. Type FORMAT B: and press **Enter**.

4. When the command prompt reappears, type EXIT and press **Enter** to return to the application program.

FASTOPEN *External*

Purpose

Speeds up access to frequently used files and directories on a hard disk.

Format

```
[d:][path]FASTOPEN d:[=n][...][/X]
```

or

```
[d:][path]FASTOPEN d:[=(n,m)][...][/X]
```

or

```
[d:][path]FASTOPEN d:[=([n],m)][...][/X]
```

Description

To access a file, MS-DOS needs to consult two tables: the directory and the FAT (File Allocation Table). Reading the information from these tables requires movement of the disk's read/write head, a time-consuming process. FASTOPEN speeds up the access time for hard disk files by storing selected directory and FAT information in memory.

The FASTOPEN command allocates two memory buffers, one for directory information and one for FAT information. The size of these buffers is governed by the *n* and *m* parameters, respectively.

By default, *n* and *m* are each set to 34, which means that 34 directory entries and 34 FAT entries can be stored in memory at any one time.

The minimum values for *n* and *m* are 10 and 1, respectively. The maximum values for each are a total of 999 for all drives named in the FASTOPEN command. Thus, the following command is not valid, because the total value of *n* for the three drives exceeds 999:

```
FASTOPEN C:=(400,64) D:=(400,64) E:=(400,64)
```

The optimum values for *n* and *m* depend on your system configuration. To set them properly requires experimentation. If you set values that are too low, you don't get the full benefit of directory and FAT buffering. If your values are too high, overhead is incurred as MS-DOS searches through buffers that are unnecessarily large.

If you specify one parameter but omit the other, no buffer is set aside for the omitted parameter. For example, the following command buffers only the directory entries:

```
FASTOPEN C:=(128,)
```

The /X switch allows FASTOPEN to store its information in expanded memory.

You can install FASTOPEN using an INSTALL command in your CONFIG.SYS file. See Chapter 6 for information about how to do this.

Cautions

You can use FASTOPEN only for hard disks. You cannot use it for diskettes or for disks created by ASSIGN, JOIN, or SUBST.

If you use INSTALL to activate FASTOPEN, you must include the extension in the INSTALL statement, like this:

```
INSTALL=FASTOPEN.EXE [parameters]
```

If you use INSTALL to activate FASTOPEN with the /X switch, the statement that activates your expanded memory driver must appear before the INSTALL statement in your CONFIG.SYS file. See Chapter 6 for details on the CONFIG.SYS file.

You can use FASTOPEN only once for each hard disk.

The value for *n* should be at least as great as the deepest subdirectory level on the hard disk. For example, if the path to your deepest subdirectory involves 12 directory names, be sure *n* is at least 12.

Examples

To set FASTOPEN to store 50 directory entries and 100 FAT entries on drive C, type the following and press **Enter**:

```
FASTOPEN C:=(50,100)
```

To have FASTOPEN speed access to files on drives C, D, and E, using the default buffer sizes and storing information in expanded memory, type the following and press **Enter**:

```
FASTOPEN C: D: E: /X
```

Purpose

Compares files on your disk.

Format

```
[d:][path]FC [/A][/B][/C][/L][/N][/T][/W][/LB n]
       [/n][d:][path]filename1 [d:][path]filename2
```

Description

Use FC to compare two files on your disk if you want to know exactly where they differ. You can output any differences between the two files to the monitor or to a third file.

Note
If you just want to know if two files are different, you may prefer to use the COMP (compare) command. It is simpler but does not provide the detailed information that FC produces. An easier way to use the FC command is through the MENU utility. See Chapter 5 for instructions.

You can compare either text files or binary files. Text files contain ASCII data. Binary files are output by an assembler, the MS-DOS LINK utility, or a high-level language compiler.

FC compares files in one of two ways: on a line-by-line or a character-by-character basis. A line-by-line comparison isolates blocks of lines that are different between the two files and prints those blocks. A character-by-character comparison shows the characters that are different between the two files, in hexadecimal value.

When comparing binary files, FC displays all bytes that differ. When comparing ASCII files, FC displays the lines that differ unless the number of consecutive lines that differ exceeds the size of the internal buffer.

You can use the following switches with FC:

Switch	*Function*
/A	Abbreviates the output of an ASCII comparison
/B	Forces a binary comparison
/C	Does not distinguish between uppercase and lowercase letters
/L	Forces an ASCII comparison
/N	Displays line numbers for an ASCII comparison
/T	Compresses tabs; normally tabs are expanded to 8 spaces
/W	Compresses tabs and spaces
/LB *n*	Sets the internal line buffer to *n* lines
/*n*	Allows you to specify the number of consecutive lines to match; the default is 2

When you use /A, FC displays only the beginning and ending lines of each set of differences. Intermediate lines are represented by ellipses (...).

Use /B to force a binary comparison of both files. FC uses this switch automatically when you compare .EXE, .COM, .SYS, .OBJ, .LIB, or .BIN files. The two files are compared byte-to-byte, with no attempt to resynchronize after a mismatch. The mismatches print as follows:

```
-ADDRS--F1--F2-
xxxxxxxx yy zz
```

In this example, *xxxxxxxx* is the relative address in hexadecimal of the pair of bytes from the beginning of the file. Addresses start at 00000000. The mismatched bytes are *yy* and *zz* from *file1* and *file2*, respectively. If one of the files contains less data than the other, FC displays a message. For example, if *file1* ends before *file2*, then the following message is displayed:

```
File2 longer than file1
```

Use /C to ignore the case of letters so FC treats uppercase and lowercase letters as equals. Only use /C in source comparisons. With /C set, these lines are a match:

```
Much_MORE_data_IS_NOT_FOUND
much_more_data_is_not_found
```

Use /L to compare ASCII files. FC uses this switch automatically when you compare any non-system files; that is, any files without the .EXE, .COM, .SYS, .OBJ, .LIB, or .BIN extensions.

Use /N to display line numbers on an ASCII comparison.

Use /T to compress tabs and use /W to compress both tabs and spaces during the comparison. Only use /W in source comparisons. With /W set, FC views multiple, contiguous blanks in any line as a single space. Note that although FC compresses blanks, it does not ignore them. The two exceptions are beginning and ending blanks in a line, which FC does ignore. For example (the underscores represent blanks), FC sees all but the fourth line as matches:

```
____More__data_to_be_____found_______
More_data_to_be_found
__________More_______data_to_be_____found_________
_____Moredata_to_be_found
```

If you specify both the /W and /C switches, then FC compresses blanks and ignores case. For example, these two lines match:

```
____DATA_was_found_____
data_was_found
```

Use /LB *n* to set the size of the internal line buffer. The default length is 100 lines. If the two files have more than this number of consecutive differing lines, FC ends.

Use */n* to specify the number of lines that must match for FC to consider the files a match after it finds a difference. If you don't specify this switch, it defaults to two. Use this switch only in source comparisons.

If the files are identical, FC displays the following message:

```
fc: no differences encountered
```

If the files are not identical, FC reports the differences between the two files you specify by displaying the first filename, the first line that matches, then the lines that differ between the files, and finally, the last line to match in both files. FC then displays the name of the second file along with the first and last line that match. The default for the number of lines to match between the files is two. (To change this default, specify the number of lines with the */n* switch.) FC continues to list each difference. FC then returns to the MS-DOS command prompt.

FC displays the differences and matches between the two files you specify on your screen unless you redirect the output to a file. Use the greater-than symbol (>) to redirect output to the printer or to a file.

The following command compares File1 and File2, and sends the output to DIFFER.TXT on the default drive:

```
FC File1 File2 >DIFFER.TXT
```

Examples

To understand how FC compares differences, you could store these two ASCII files on disk:

ALPHA.ASM	*BETA.ASM*
A	A
B	B
C	C
D	G
E	H
F	I
G	J
H	1
I	2
M	P
N	Q
O	R
P	S
Q	T
R	U
S	V
T	4
U	5
V	W
W	X
X	Y
Z	Z

To compare the two files and display the differences on the screen, type the following and press **Enter**:

```
FC ALPHA.ASM BETA.ASM
```

FC does not compress blanks and does not ignore case. Here, two lines must be the same to constitute a match.

The screen displays the following (the notes do not appear):

```
***** ALPHA.ASM
C
D
E
F
G
***** BETA.ASM
C
G
*****

***** ALPHA.ASM
I
M
N
O
P
***** BETA.ASM
I
J
1
2
P
*****

***** ALPHA.ASM
V
W
***** BETA.ASM
V
4
5
W
*****
```

Note: ALPHA file contains CDEFG, BETA contains CG.

Note: ALPHA file contains IMNOP where BETA contains IJ12P.

Note: ALPHA file contains VW where BETA contains V 45W.

You can print the differences on the printer using the same two source files. In this example, four successive lines must be the same to constitute a match. Type the following and press **Enter**:

```
FC /4 ALPHA.ASM BETA.ASM >PRN
```

The following is output to the printer:

```
***** ALPHA.ASM
C
D
E
F
G
H
I
M
N
O
P
***** BETA.ASM
C
G
H
I
J
1
2
P
*****

***** ALPHA.ASM
V
W
***** BETA.ASM
V
4
5
W
```

Note: P is the first of a string of 4 matches.

Note: W is the first of a string of 4 matches.

The /B switch in the next example forces a binary comparison. You must type this switch and any other before the filenames in the FC command line.

To force a binary comparison and display the differences on the monitor using the same two source files, type the following and press **Enter**:

```
FC /B ALPHA.ASM BETA.ASM
```

The screen displays the following:

```
00000009 44 47
0000000C 45 48
0000000F 46 49
00000012 47 4A
00000015 48 31
00000018 49 32
0000001B 4D 50
0000001E 4E 51
00000021 4F 52
00000024 50 53
00000027 51 54
0000002A 52 55
0000002D 53 56
00000030 54 34
00000033 55 35
00000036 56 57
00000039 57 58
0000003C 58 59
0000003F 59 5A

fc: ALPHA.ASM longer than BETA.ASM
```

You can specify a path to locate the files you want to compare. For example:

```
FC B:\WORDPROC\BUSINESS\BLUE.TXT
     \WORDPROC\BUSINESS\BLUE.BAK
```

FC compares BLUE.TXT in the \WORDPROC\BUSINESS directory of drive B with BLUE.BAK in the same directory.

FDISK

External

Purpose

Creates one or more partitions on a hard disk.

Format

`[d:][path]FDISK`

Description

Use FDISK to partition your hard disk. You can create as many as four separate partitions to run four operating systems. (FDISK stands for "fixed disk," which is another term for hard disk.)

If MS-DOS 4.01 is the only operating system you use, you may want to create only one partition to occupy the entire disk. Or, you can divide the disk into two or more MS-DOS partitions. If you create more than one partition, MS-DOS assigns each partition a separate drive letter (C, D, and E, for example).

Your Epson dealer may have partitioned your hard disk for you. In that case, use FDISK only to change the partitions.

If your hard disk has not been partitioned, you can follow the instructions in the MS-DOS 4.01 Installation Guide to install MS-DOS 4.01. During this process, the SELECT program determines whether your hard disk has been partitioned and partitions it for you. If you accept the partition choices offered by the SELECT program, you do not need to do anything further. If you do not accept the default partition choices, you see the same menus and prompts described in this section.

WARNING
Partitioning the hard disk destroys all data it contains. Use FDISK (or run the SELECT program) *before* storing files on a hard disk. If files are currently stored on the hard disk, make backup copies before running FDISK. Also, FDISK does not work on drives that have been reassigned by the JOIN or SUBST command.

You can divide your hard disk into as many as four partitions. Each partition is a separate working area. You may install a different operating system on each partition, or you may create more than one partition for MS-DOS. If you want to install a different operating system on a partition, see the instructions for that operating system.

You need to create at least one partition on your hard disk even if MS-DOS is the only operating system you use. There is no limit to the size of an MS-DOS 4.01 partition other than the size of the disk.

You can create two kinds of MS-DOS partitions: *primary* and *extended*. The first MS-DOS partition is the primary partition. All other MS-DOS partitions are extended partitions.

A primary partition may be *active* or *inactive*. You can use an active partition to start MS-DOS. If you want MS-DOS 4.01 to start automatically from the hard disk each time you turn on or reset your computer, you must make the primary partition active. Extended partitions cannot be active.

You must divide extended partitions into one or more *logical drives*. A logical drive is a subdivision of an extended partition that is assigned a separate drive letter. You cannot divide a primary partition into logical drives.

MS-DOS 4.01 identifies the primary partition of your first hard disk as drive C. It then assigns the primary partitions of any additional hard disks the next drive letters after C. For example, if you have a second hard disk, its primary partition is drive D. After MS-DOS names the primary partitions of all hard disks, it names the extended partitions, beginning with the first extended partition on the first hard disk.

MS-DOS names the logical drives in order within each extended partition. For example, if you have one hard disk with a primary partition and one extended partition, and if the extended partition has two logical drives, the primary partition is drive C and the two logical drives are D and E.

If you have two hard disks, each with a primary partition and an extended partition, and each extended partition has three logical drives, the names are as follows:

The first hard disk's primary partition is C and its logical drives are: E, F, and G.

The second hard disk's primary partition is D and its logical drives are: H, I, and J.

If you add a hard disk to your system, FDISK renames the extended partition(s) on your first hard disk as it partitions the second hard disk. Similarly, if you use FDISK to delete partitions or logical drives, the remaining partitions or logical drives may be renamed.

Using FDISK

You can use FDISK to perform the following tasks:

- Create a primary MS-DOS partition
- Create an extended MS-DOS partition
- Create logical drives within an extended MS-DOS partition
- Make a primary MS-DOS partition active
- Delete an MS-DOS partition
- Display partition information
- Partition additional hard disks.

Creating a Primary MS-DOS Partition

To create a primary MS-DOS partition, follow these steps:

1. At the MS-DOS command prompt, type FDISK and press **Enter**. You see the main FDISK menu:

```
Current fixed disk drive: 1

Choose one of the following:

1. Create DOS Partition or Logical DOS Drive
2. Set active partition
3. Delete DOS Partition or Logical DOS Drive
4. Display partition information

Enter choice: [1]

Press Esc to exit FDISK
```

2. Press **1**, then press **Enter**. You see the following menu:

```
1. Create Primary DOS Partition
2. Create Extended DOS Partition
3. Create Logical DOS Drive(s) in the
   Extended DOS Partition
```

3. Press **1**, then press **Enter**. You see the following:

```
Current fixed disk drive: 1

Do you wish to use the maximum available size
for a Primary DOS Partition and make the
partition active (Y/N).............? [Y]
```

4. If you want to create a single active primary partition using all available space on your hard disk, press **Y**, then press **Enter**. FDISK instructs you to restart your computer by pressing **Ctrl Alt Del**. You can now begin using MS-DOS.

5. If you want to use only part of the available space for the primary MS-DOS partition, press **N**, then press **Enter**. You see a screen similar to the following:

```
Current fixed disk drive : 1

Total disk space is 41 Mbytes
(1 Mbyte=1048576 bytes)

Maximum space available for partition is
41 Mbytes (100%)

Enter partition size in Mbytes or percent of
disk space (%) to create a Primary DOS
Partition.....: [41%]

No partitions defined
```

6. Specify the number of megabytes or the percentage of available space you want to use for the primary partition. To specify the size in megabytes, type a whole number and press **Enter**. To specify the size as a percentage of available space, type a whole number followed by a percent sign. Then press **Enter**.

7. Press **Esc** to return to the main FDISK menu.

FDISK creates the primary partition. However, before you can start MS-DOS from this partition, you must make the partition active. See "Setting the Active Partition."

Creating an Extended MS-DOS Partition and Logical Drives

To create an extended MS-DOS partition, follow these steps:

1. Create a primary MS-DOS partition (see "Creating a Primary MS-DOS Partition," above). You must have a primary partition to create an extended partition.

2. From the main FDISK menu, press **1**, then press **Enter**. You see the following menu:

```
1. Create Primary DOS Partition
2. Create Extended DOS Partition
3. Create Logical DOS Drive(s) in the
   Extended DOS Partition
```

3. Press **2**, then press **Enter**. You see a display of partition status information, similar to the following:

```
Current fixed disk drive: 1

Partition Status Type    Size in Percentage
                         Mbytes  of Disk Used

C: 1             PRI DOS 21      52%

Total disk space is 41 Mbytes
(1 Mbyte=1048576 bytes)

Maximum space available for partition is
20 Mbytes (40%)

Enter partition size in Mbytes or percent of
disk space (%) to create an Extended DOS
Partition.....................: [20]
```

Note
An A in the Status column indicates the partition is active.

4. Specify the number of megabytes or the percentage of available space that you want to use for the extended partition. To specify the size in megabytes, type a whole number and press **Enter**. To specify the size as a percentage of available space, type a whole number followed by a percent sign. Then press **Enter**. You see a new display of partition status information, similar to this:

```
Current fixed disk drive: 1

Partition Status Type    Size in Mbytes  Percentage of Disk Used

C: 1            PRI DOS 21              52%
C: 2            EXT DOS 20              48%

Extended DOS Partition created

Press Esc to continue
```

Note
An A in the Status column indicates the partition is active.

5. Press **Esc**. FDISK displays a screen similar to the following:

```
No logical drives defined

Total Extended DOS Partition size is 20 Mbytes (1 Mbyte=1048576 bytes)

Maximum space available for logical drives is 20 Mbytes (100%)

Enter logical drive size in Mbytes or percent of disk space (%)...: [20]

Press Esc to return to FDISK Options
```

6. Specify the number of megabytes or the percentage of available space you want to use for the first logical drive. To specify the size in megabytes, type a whole number and press **Enter**. To specify the size as a percentage of available space, type a whole number followed by a percent sign. Then press **Enter**.

   If you specify less than the total available space for the first logical drive, FDISK redisplays the partition status information so you can specify another logical drive. After you assign all available space to logical drives, FDISK displays this message:

   ```
   All available space in the Extended DOS
   Partition is assigned to logical drives.
   ```

7. Press **Esc** to return to the main FDISK menu.

Setting the Active Partition

Although each hard disk can have as many as four partitions, only the primary partition of the first hard disk can be active. The active partition takes control of your computer each time you turn it on or reset it. If you want MS-DOS to start automatically each time you start or reset your computer, you must make the primary partition on your first hard disk active. Follow these steps:

1. From the main FDISK menu, press **2**, then press **Enter**. FDISK displays a screen similar to the following:

   ```
   Current fixed disk drive: 1

   Partition Status Type    Size in Percentage
                            Mbytes  of Disk Used

   C: 1             PRI DOS 21      52%
      2             EXT DOS 2Ø      48%

   Total disk space is 41 Mbytes
   (1 Mbyte=1Ø48576 bytes)

   Enter the number of the partition you want to
   make active.....[ ]
   ```

Note
An A in the Status column indicates the partition is active.

2. Press **1**, then press **Enter**.
3. Press **Esc** to return to the main FDISK menu.

Deleting an MS-DOS Partition

To delete an MS-DOS partition, follow these steps:

1. From the main FDISK menu, press **3**, then press **Enter**. You see the following:

```
Current fixed disk drive: 1

Choose one of the following:

1. Delete Primary DOS Partition
2. Delete Extended DOS Partition
3. Delete Logical DOS Drive(s) in the
   Extended DOS Partition

Enter choice: [ ]
```

2. Press the number of the option you want, then press **Enter**. See the instructions below for the option you choose.

You must delete all logical drives in an extended partition before you can delete the extended partition. You must delete all extended partitions before you can delete a primary partition. To delete all partitions on a hard disk, follow the instructions in the order they are presented below.

WARNING
When you delete an MS-DOS partition, FDISK deletes any data stored in that partition. There is no way to recover that data. Be sure to back up any data you want to keep before deleting a partition.

Deleting the logical drive(s) in an extended MS-DOS partition

To delete a logical drive from an extended MS-DOS partition, follow these steps:

1. Press **3** at the Delete MS-DOS Partition menu. Then press **Enter**. You see a display similar to the following:

```
Drv  Volume Label  Mbytes   System  Usage

D:   COOLIDGE        8      FAT 12   40%
E:   HARDING         8      FAT 12   40%
F:   ROOSEVELT       4      FAT 12   20%

Total Extended DOS Partition size is 20
Mbytes (1 Mbyte=1048576 bytes)

Warning! Data in a deleted Logical DOS Drive
will be lost.

What drive do you want to delete? [ ]
```

The volume label is the name you assigned when you formatted each logical drive.

The System column displays the file system (file allocation table) in use for each logical drive. Drives larger than 16MB use the FAT 16 system. Drives up to 16MB use the FAT 12 system. If a drive has not been formatted, the System column displays UNKNOWN.

2. To delete a logical drive, type the drive letter and press **Enter**. FDISK displays this prompt:

```
Enter drive's volume label [           ]
```

3. Type the volume label and press **Enter**. FDISK displays this prompt:

```
Are you sure (Y/N).......? [N]
```

4. If you do not want to delete the logical drive, press **N** and **Enter**.

5. To delete the logical drive, press **Y** and **Enter**.

Deleting an extended MS-DOS partition

To delete an extended MS-DOS partition, follow these steps:

1. Press **2** at the Delete MS-DOS Partition menu. Then press **Enter**. You see a display similar to the following:

```
Partition Status Type    Size in Percentage
                         Mbytes  of Disk Used

C: 1             PRI DOS 21      52%
   2             EXT DOS 20      48%

Total disk space is 41 Mbytes
(1 Mbyte=1048576 bytes)

WARNING! Data in the deleted Extended DOS
Partition will be lost.

Do you wish to continue (Y/N).....[N]
```

2. If you do not want to delete the extended partition, press **N** and **Enter**.

3. If you want to delete the extended partition, press **Y** and **Enter**. FDISK displays this message:

```
Extended DOS partition deleted
```

Deleting a primary MS-DOS partition

To delete a primary MS-DOS partition, follow these steps:

1. Press **1** at the Delete MS-DOS Partition menu. Then press **Enter**. You see a display similar to the following:

```
Current fixed disk drive: 1

Partition Status Type    Size in Percentage
                         Mbytes  of Disk Used

C: 1      A      PRI DOS 41      1ØØ%

Total disk space is 41 Mbytes
(1 Mbyte=1Ø48576 bytes)

WARNING! Data in the deleted Primary DOS
Partition will be lost.

Do you wish to continue (Y/N).....? [N]
```

Note
An A in the Status column indicates the partition is active.

2. If you do not want to delete the primary MS-DOS partition, press **N**, then press **Enter**.
3. If you do want to delete the primary MS-DOS partition, press **Y**, then press **Enter**. You see the following message:

```
Primary DOS partition deleted
```

4 Press **Esc** to return to the main FDISK menu.

Displaying Partition Information

To display information about how the hard disk is partitioned, follow these steps:

1. Press **4** at the main FDISK menu. Then press **Enter**. You see a display similar to the following:

```
Current fixed disk drive: 1

Partition Status Type    Size in Percentage
                         Mbytes  of Disk Used

C: 1      A      PRI DOS 21      52%
   2             EXT DOS 20      48%

Total disk space is 41 Mbytes
(1 Mbyte=1048576 bytes)

The Extended DOS Partition contains Logical
DOS Drives.

Do you want to display the logical drive
information (Y/N)...? [Y]
```

Note
An A in the Status column indicates the partition is active.

2. To see the logical drive information, press **Y** and **Enter**. You see a display similar to the following:

```
Drv  Volume Label  Mbytes   System  Usage

D:   COOLIDGE      8        FAT 12  40%
E:   HARDING       8        FAT 12  40%
F:   ROOSEVELT     4        FAT 12  20%

Total Extended DOS Partition size is 20
Mbytes (1 Mbyte=1048576 bytes)
```

The volume label is the name you assigned when you formatted each logical drive.

The System column displays the file system (file allocation table) in use for each logical drive. Drives larger than 16MB use the FAT 16 system. Drives up to 16MB use the FAT 12 system. If a drive has not been formatted, the System column displays UNKNOWN.

3. Press **Esc** to return to the main FDISK menu.

Partitioning Additional Hard Disks

If you have more than one hard disk, the main FDISK menu provides five options. It looks like this:

```
Current fixed disk drive: 1

Choose one of the following:

1. Create DOS Partition or Logical DOS Drive
2. Set active partition
3. Delete DOS Partition or Logical DOS Drive
4. Display partition information
5. Select next fixed disk drive

Enter choice: [1]

Press Esc to exit FDISK
```

To partition an additional hard disk, press **5** and **Enter**. Repeat if necessary until the "Current fixed disk drive" line indicates the hard disk you want to partition. Then follow the appropriate instructions above.

FIND

External (filter)

Purpose

Searches for a specific string of text within one or more files.

Format

```
[d:][path]FIND [/C][/N][/V] "string"
               [[d:][path]filename...]
```

Description

Use FIND to locate all occurrences of a particular string in a directory or in each of the files you specify. FIND is called a filter because it reads data from a standard input device, filters out the string that matches your specification, and then writes the result to a standard output device. For example, it can display the matching line on your monitor screen.

FIND does not allow wildcards in the filename.

You can use three switches with FIND:

Switch	*Function*
/C	Prints the number of lines that match the string (or that don't match if you use the /V switch)
/N	Precedes each line with its relative line number in the file
/V	Displays all lines that do NOT contain the string

If you include both /C and /N in the command, the /C switch overrides /N.

If you do not enter any switches, FIND displays all lines in the specified file(s) that contain the string.

Type quotation marks around the string. If the text in your string contains quotation marks, type two quotation marks for each occurrence. For example, to find the following string:

She said "Goodbye" and left

type the following:

```
FIND "She said ""goodbye"" and left"
```

FIND locates the exact string of characters you specify, so be sure to type upper- or lowercase letters just as you want to find them. If you type uppercase Y, the program does not find lowercase y. The same rule applies to spaces and punctuation. If FIND does not locate the string you specify in the file or files you list, it displays just the filenames on the screen.

Examples

To display all lines from BOOK1.TXT and BOOK2.TXT (in that order) that contain the string A Day in the Life, type the following and press **Enter**:

```
FIND "A Day in the Life" BOOK1.TXT BOOK2.TXT
```

To display the filenames on drive B that do *not* contain the string DAT, type the following and press **Enter**:

```
DIR B: ¦ FIND /V "DAT"
```

To display all subdirectories of the current directory, type the following and press **Enter**:

```
DIR ¦ FIND "<DIR>"
```

You can include special characters in the string without redirecting input or output.

FORMAT

External

Purpose

Formats the disk in the specified drive to accept MS-DOS files.

Format

```
[d:][path]FORMAT d: [/1][/4][/8][/B][/F:size]
                 [/S][/V:label][/N:nn][/T:nn]
```

Description

Use FORMAT to format your disks for use with MS-DOS.

Note
An easier way to use the FORMAT command is through the MENU utility. See Chapter 5 for instructions.

Formatting initializes the directory and file allocation tables and analyzes the disk for any defective tracks. All new disks must be formatted before you can use them with MS-DOS.

WARNING
FORMAT erases any data stored on a disk, so be sure to back up any files you want to keep before you format a disk that contains data.

On a hard disk, formatting destroys the entire contents of any existing MS-DOS partition, including all subdirectories and files.

You must specify a drive descriptor when you use the FORMAT command. If you do not, you see this message:

```
Required parameter missing
```

You can use the following switches with this command. If you do not use any switches, the default parameters for FORMAT are no system files, the maximum available sectors, and two sides.

Switch	*Function*
/1	Formats only one side of a 5¼-inch diskette for single-sided use.
/4	Formats a 5¼-inch, 360KB double-density diskette in a 1.2MB high-density drive.
/8	Formats a diskette with 8 sectors per track.
/B	Reserves space for the system files (which you can copy later with the SYS command). This switch allows you to place any version of MS-DOS on the diskette.
/F:*size*	Specifies the capacity (in kilobytes) of the diskette to be formatted. This switch provides an easier way to specify the same information that you can express with the /T and /N switches.
/S	Copies the operating system files from the disk in the default drive to the newly formatted disk.
/V:*label*	Assigns *label* as the volume label for the newly formatted disk or diskette. If you omit this parameter (or if you omit *label*), MS-DOS prompts you for a volume label after it formats the disk.
/N:*nn*	Specifies the number of sectors per track if you want to format a diskette to use less of the maximum capacity allowed by the disk drive. For a 360KB diskette, *nn* is equal to 9 or 8. For a 1.2MB diskette, *nn* is equal to 15, 9, or 8.
/T:*nn*	Specifies the number of tracks on the diskette to be formatted. Use this switch when you want to format a diskette for less than the maximum capacity supported by the disk drive. For a 360KB diskette, *nn* is equal to 40. For a 720KB or 1.2MB diskette, *nn* is equal to 80 or 40.

You can use the following switches with the specified disk types:

Disk type	*Parameters allowed*
160/180KB	/1, /4, /8, /B, /F, /S, /V, /N, /T
320/360KB	/1, /4, /8, /B, /F, /S, /V, /N, /T
720KB	/F, /S, /V, /N, /T
1.2MB	/F, /S, /V, /N, /T
1.44MB	/F, /S, /V, /N, /T
Hard Disk	/S, /V

If you use the /S switch, FORMAT copies the operating system files from the disk in the default drive to the target disk. If the system does not have enough memory to store the files, it copies as many as it can to memory, formats the target disk, and then copies the files in memory to the disk. Then it reads the remaining files from the source disk to memory and copies them to the target disk. If you have removed the source diskette from the drive, the program prompts you to reinsert it before continuing.

The *label* parameter you specify with the /V switch can be up to eleven characters in length. It must comply with the rules that define a valid filename. Do not, however, include a period (.) between the eighth and ninth characters.

For compatibility with earlier versions of MS-DOS, you can include the /V switch with no *label* parameter. In this case, MS-DOS prompts for a volume label when it finishes formatting the disk or diskette. (The volume label prompt also appears if you omit /V altogether.)

The FORMAT program formats the diskette according to the format of the diskette drive and any switches you have entered in the command. If you format a single-sided diskette, be sure to include the /1 switch; otherwise, you will not be able to use the diskette in a single-sided drive.

The /F:*size* switch allows you to format a diskette to less than its maximum capacity without using /T and /N. For compatibility with earlier versions of MS-DOS, however, you may continue to use the /T and /N switches. The *size* parameter specifies the formatted capacity of the diskette in kilobytes; you may include the letters K and B with the number. For example, /F:720, /F:720K, and /F:720KB are all equivalent specifications.

The FORMAT program marks any defective tracks as reserved. These reserved tracks are not allocated to data files.

FORMAT also assigns a volume serial number to the target disk or diskette.

FORMAT produces a status report that lists the total disk space, defective disk space, space allocated to MS-DOS system files, volume serial numbers, and amount of space available for your files.

FORMAT sets the following exit codes, which may be tested in a batch file with the IF ERRORLEVEL command:

0	Most recent format successful
3	Terminated because user pressed **Ctrl Break**
4	Terminated because of error condition
5	Terminated because user pressed **N** in response to hard disk warning prompt.

Formatting a Hard Disk

If you want to use FORMAT to format a hard disk, be sure you have first used FDISK to create the MS-DOS partition. If you later change the MS-DOS partition with FDISK, you will need to run FORMAT again for the new partition.

When you format a hard disk, the FORMAT program prompts you for the volume label:

```
Enter current volume label for drive d:
```

If your hard disk does not have a volume label, press the **Enter** key.

> **Note**
> If the hard disk has never been formatted or if it has a bad boot sector, FORMAT does not prompt you for the volume label.

If the volume label you enter does not match the label for the disk, the screen displays the following message:

```
Invalid Volume ID
Format terminated
```

After you enter the proper volume label, FORMAT displays this message:

```
WARNING, ALL DATA ON NON-REMOVABLE DISK
DRIVE d WILL BE LOST!
Proceed with Format (Y/N)?
```

If you want to format the hard disk, press **Y** and **Enter**. If not, press **N** and **Enter**.

Cautions

If you have reassigned disk drive descriptors with ASSIGN, cancel the assignments before you run FORMAT. Otherwise, you could destroy data.

Do not use FORMAT with any drives that are currently used in an ASSIGN, JOIN, or SUBST command.

This command may not work in a network environment.

The /4 switch allows you to use a 360KB, double-density diskette in a 1.2MB, high-density drive. If you write to this diskette in a 1.2MB drive, however, you may have trouble reading or writing to it in a 360KB drive.

The /S and /B switches cannot be used together.

The /F:*size* and /8 switches cannot be used together.

Do not use /T, /N, /F:*size* when formatting a hard disk.

Do not use /N without /T or vice versa.

Do not use both /T and /N and /F:*size*.

Examples

To format a diskette in drive B using all default values, type the following and press **Enter**:

```
FORMAT B:
```

You see the prompt:

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

Insert the diskette you want to format in drive B and press **Enter** to start formatting. As MS-DOS formats the diskette, it displays the following information:

```
xx percent of disk formatted
```

When the diskette is formatted, you see this message:

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

After you enter a volume label (or press **Enter** to enter no volume label), you see a display like this:

```
xxxxxxx bytes total disk space
xxxxxxx bytes available on disk
    xxx bytes in each allocation unit
   xxxx allocation units available on disk
Volume Serial Number is xxxx-xxxx
Format another (Y/N)?
```

Press **Y** and **Enter** to format another diskette or press **N** and **Enter** to leave the program.

An *allocation unit*, or *cluster*, is the smallest increment of disk space used by MS-DOS in writing files. The size of the allocation unit depends on the format of the diskette.

To format a system diskette in drive A and specify the volume label SYSTEM_DISK, type the following and press **Enter**:

```
FORMAT A: /S /V:SYSTEM_DISK
```

To format the hard disk (after you have partitioned it with FDISK), type the following and press **Enter**:

```
FORMAT C:/S
```

The screen prompts you to enter the current volume label (if there is one) or enter a new one. Then the program formats the hard disk and copies the system files to it.

GRAFTABL *External*

Purpose

Loads a graphics table into memory to enable display of graphic characters in color/graphics mode.

Format

```
[d:][path]GRAFTABL [nnn][/STATUS][?]
```

Description

If your computer has a color/graphics adapter, you can use the GRAFTABL command to display ASCII characters with decimal values above 127 in graphics mode. GRAFTABL loads a special graphics character generator into memory so you can access these characters when you are using special application software or a programming language such as BASIC. You do not need to load this table if you are not in graphics mode.

If you also include one of the following code page numbers, you can access characters in that international language character set:

437	United States—English
850	Multilingual
860	Portugal
863	Canada—French
865	Nordic

The default is 437 (the United States). After you enter the GRAFTABL command, the screen displays a message such as the following:

```
Previous Code Page: None
Active Code Page: 437
```

To display the number of the code page currently being used, include the /STATUS switch. You may abbreviate /STATUS to /STA.

For a summary of the parameters supported by GRAFTABL, type the following and press **Enter**:

```
GRAFTABL ?
```

You see a display like this:

```
DOS command line parameters supported:

/STA - Request Status only
?    - Display this summary of parameters

Code Pages available:
437  - USA Graphic Character Set
850  - Multi-lingual Graphic Character Set
860  - Portuguese Graphic Character Set
863  - Canadian French Graphic Character Set
865  - Nordic Graphic Character Set
```

Once it is loaded, GRAFTABL remains in memory until you turn off or reset the computer. You can access any of the graphics or national language characters through the application software or programming language you are using.

Load the graphics table only once each session. If you use the command a second time, GRAFTABL displays the following message:

```
Previous code Page: 437
```

You may want to include GRAFTABL in your AUTOEXEC.BAT file so the table loads automatically every time you boot the system, or include it in a batch file for an application or programming diskette.

This command increases the size of MS-DOS in memory by 1KB.

Examples

To load the graphics character table, type the following and press **Enter**:

```
GRAFTABL
```

To load the French Canadian version of the graphics table, type the following and press **Enter**:

```
GRAFTABL 863
```

GRAPH24 *External*

Purpose

Enables printing of screen graphics on a 24-pin or laser printer when you are using a color/graphics adapter.

Format

`[d:][path]GRAPH 24 [/R]`

Description

The GRAPH24 command is exactly like the GRAPHICS command except that it supports 24-pin printers and you cannot use the /B switch. If you have a 9-pin printer, use the GRAPHICS command. If you have a 24-pin printer, use GRAPH24.

You can use GRAPH24 with a laser printer that supports 24-pin mode, such as the Epson GQ-3500.

For detailed information on GRAPH24, see the GRAPHICS command, next in this chapter.

GRAPHICS *External*

Purpose

Enables printing of screen graphics when using a color/graphics adapter.

Format

```
[d:][path]GRAPHICS [type][[d:][path]profile][/R]
                   [/B][/LCD][/printbox=id]
```

Description

Use GRAPHICS to make exact printed copies of the image on your screen on a graphics-compatible printer. After GRAPHICS loads, the command prompt reappears on the screen. GRAPHICS is now in memory, and you can use it whenever necessary. GRAPHICS stays in memory until you reset or turn off the system.

Note
If you have a 24-pin printer, use the GRAPH24 command instead of GRAPHICS.

The *type* parameter specifies the type of printer you are using. The GRAPHICS command supports the following printer types (the default type is GRAPHICS):

Type	*Printer(s)*
GRAPHICS	IBM Graphics Printer IBM Proprinter IBM Pageprinter IBM Quietwriter II IBM Quietwriter III
GRAPHICSWIDE	IBM Quietwriter II with 13½-inch wide paper IBM Proprinter XL
THERMAL	IBM PC Convertible Printer
COLOR1	IBM Color Printer with a black ribbon

Type	*Printer(s)*
COLOR4	IBM Color Printer with a red, green, blue, and black ribbon
COLOR8	IBM Color Printer with a cyan, magenta, yellow, and black ribbon

The *profile* parameter specifies a file containing information on all supported printers. The default file is PRINTER.PRO. This file should be stored in your MS-DOS directory.

You can use the following switches with GRAPHICS:

Switch	*Function*
/R	Prints exactly what is on the monitor: black as black and white as white. If you do not specify /R, black prints as white and white as black. (Other colors are not affected by this switch.)
/B	Prints the background color (COLOR4 and COLOR8 printers only).
/LCD	Prints the screen image from the LCD screen of a portable computer.
/printbox:*id*	Selects the print box size. *id* should match the first operand of a printbox statement in the printer profile file. The two options for *id* are STD and LCD. pb is the abbreviated form of printbox.

You need to enter the GRAPHICS command only once after booting the system. To print the image you see on your screen, just press **Shift Print Screen** (or **PrtSc**). If you have an enhanced keyboard, you need to press only **Print Screen**.

If the computer is in 320x200 color graphics mode, and if the printer type is COLOR1 or GRAPHICS, GRAPHICS prints the screen contents with up to four shades of gray. If the computer is in 640x200 color graphics mode, GRAPHICS prints the screen contents sideways.

To enable GRAPHICS printing, your system's monitor must be in graphics mode.

You can include the GRAPHICS command in a batch file, such as AUTOEXEC.BAT. GRAPHICS increases the size of MS-DOS in memory.

HELP

External

Purpose

Provides helpful information on MS-DOS commands.

Format

`[d:][path]HELP [command...]`

Description

Use the HELP command to ask for information on any MS-DOS command or utility progam. This help function provides useful information but is not intended to be a substitute for this manual.

The two programs that provide HELP information, HELP.COM and HELP.TXT, must be in the same directory.

Type HELP to see a menu of all the commands. Type HELP followed by the name of one or more MS-DOS commands, each separated by a space, to bypass the menu and display the messages for each command.

At the bottom of the menu, you see this prompt:

```
↑↓→←Select command name,
Enter to display command, Esc to Exit
```

Use the cursor keys to select a command. (You can select only one command name at a time from the menu.) When the correct command is highlighted, press **Enter**. The help screen for the command appears.

If there is more than one page of text, you see the prompt PgUp on the top of the screen. Press **Page Up** to display the rest of the text. After reading the help information, press **Esc** to return to the menu.

If you type HELP and one or more command names on the command line, the help information for the first command appears. Press **Esc** to see the help information for the next command. When you press **Esc** after viewing the information for the last command, MS-DOS redisplays the command prompt.

JOIN

External

Purpose

Joins a drive and all its files to a directory on another drive.

Format

```
[d:][path]JOIN [d1: d2:\directory]
```

or

```
[d:][path]JOIN d:/D
```

Description

This command makes an entire disk or diskette accessible as a subdirectory of another disk or diskette. If drive A, for example, is joined to drive C, then the directory structure of drive A becomes a first-level subdirectory on drive C.

When using application software that does not take advantage of the MS-DOS file handling features, you may have to change the current drive in order to access files from a different drive. For example, to use a separate hard disk or RAM disk for storing files, you must log on to these drives before they can be used. The JOIN command eliminates the need to change the current drive.

In the command format above, *d1* is the drive to be connected, or joined, to a directory on another drive.

d2:\directory is the directory on the second drive to be joined to the first drive. This directory must be empty and must be in the root directory. Also, this directory must not be the current directory and must have no subdirectories. All subdirectories of the source directory become sub-subdirectories in the new directory.

If *d2:\directory* does not already exist, it is created by JOIN on the specified drive.

Once you join the two drives, you cannot access the first drive directly. Use the /D switch to disconnect a join. For example, to restore the first drive, type the following and press **Enter**:

```
JOIN d: /D
```

You cannot disconnect a join if it is the current directory.

Any top-level subdirectory created by JOIN remains after you disconnect using JOIN *d*: /D. To remove such a subdirectory, use RMDIR.

To list all drives that are currently joined, type `JOIN` and press **Enter**.

Cautions

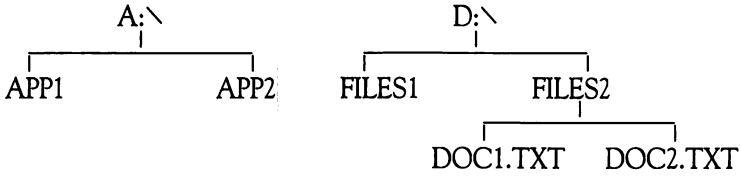
Do not use JOIN when the drive being joined is part of an ASSIGN or a SUBST command.

Do not use BACKUP, CHKDSK, DISKCOMP, DISKCOPY, FDISK, FORMAT, LABEL, RECOVER, RESTORE, or SYS while a JOIN is in effect.

This command may not work in a network environment.

Example

Assume these directory structures exist for drives A and D:



To use an application program located in APP1 of drive A while storing your files in FILES1 of drive D, you can use the JOIN command. First copy the JOIN command to your application diskette and insert it in drive A. Type the following and press **Enter**:

```
JOIN D: A:\DRIVED
```

Now, anytime you refer to the directory \DRIVED\FILES1 on drive A, all disk activity takes place in FILES1 on drive D without your having to change the current drive.

If you type DIR \DRIVED\FILES2, you see:

```
DOC1 TXT
DOC2 TXT
```

To cancel the effect of the JOIN command on drive D, type:

```
JOIN D: /D
```

KEYB

External

Purpose

Loads an alternate keyboard layout into memory.

Format

```
[d:][path]KEYB xx[,[nnn],[[d:][path]
              filename]]][/ID:zzz]
```

or

```
[d:][path]KEYB zzz
```

or

```
[d:][path]KEYB
```

Description

Use the KEYB command to replace the default keyboard layout in your computer's memory. This lets you type characters used by another language instead of the ones on the default keyboard layout.

Use the following parameters:

Parameter	*Function*
xx	A two-letter country code that specifies a keyboard layout (see table, below)
nnn	A three-digit value that specifies a code page for the selected keyboard layout. Each keyboard layout supports two code pages (see table, below). Code pages are described more fully in Chapter 9. If you omit this parameter, MS-DOS uses the currently active CON code page.
filename	A valid filename that specifies a keyboard definition file. If you omit this parameter, KEYB uses the KEYBOARD.SYS file and looks for this file in the root directory of the disk from which you boot MS-DOS.

Parameter	*Function*
zzz	A three-digit value that identifies a keyboard layout. Three countries (France, Italy, and the United Kingdom) have two enhanced keyboard layouts. You can use the *zzz* parameter to select the alternative layout for these countries (see table, below).

The following table lists the supported keyboard layouts, code pages, and keyboard IDs. Where two values are listed, the first is the primary value; the second is an alternative value that you can select.

Country	*Keyboard Layout*	*Supported Code Pages*	*Keyboard ID(s)*
Belgium	BE	437, 850	120
Canada (French-speaking)	CF	863, 850	058
Denmark	DK	865, 850	159
Finland	SU	437, 850	153
France	FR	437, 850	120, 189
Germany	GR	437, 850	129
Italy	IT	437, 850	141, 142
Latin America	LA	437, 850	171
Netherlands	NL	437, 850	143
Norway	NO	865, 850	155
Portugal	PO	860, 850	163
Spain	SP	437, 850	172
Sweden	SV	437, 850	153
Switzerland (French-speaking)	SF	437, 850	150
Switzerland (German-speaking)	SG	437, 850	000
United Kingdom	UK	437, 850	168, 166
United States	US	437, 850	103

Appendix B shows the supported keyboard layouts. For some keyboards, you need to press *dead* keys to add an accent or umlaut to a character. (See Appendix B.)

Before using the KEYB command to install a keyboard layout, you must use the MODE command to prepare the code page(s) you intend to use. The keyboard layout and code page must be compatible. If the keyboard and display do not match, MS-DOS cannot correctly translate the character keys into the correct code pages, and may display the wrong characters. (See the discussions of MODE and CHCP in this chapter.)

Once you load a particular keyboard layout into memory, it remains in effect until you reset or turn off the computer or until you enter the command to return to the default layout. You can easily switch back and forth between the default and alternate layouts. To return to the default layout, type **Ctrl Alt F1**. To return to the alternate layout, type **Ctrl Alt F2**.

You can change the keyboard layout without restarting your system. That is, you can use KEYB more than once in a working session.

If you type KEYB with no parameters, you see a message like this:

```
Current keyboard code: FR Code page: 437
Current CON code page: 437
```

If you routinely use an alternate keyboard layout, you may want to include the KEYB command in your AUTOEXEC.BAT file. You may also use an INSTALL statement in your CONFIG.SYS file to issue a KEYB command each time you start your system. (See the discussion of INSTALL in Chapter 6.)

If you use the SELECT command to format a system diskette, MS-DOS automatically initiates the keyboard layout you specify.

The KEYB command sets the following exit codes, which you may test with the batch processing command IF ERRORLEVEL:

0	Normal termination
1	Invalid code page, syntax, or language

2 Missing or invalid keyboard definition (KEYBOARD.SYS) file

3 KEYB could not create a keyboard table in resident memory

4 Error occurred while communicating with the CON device

5 Specified code page has not been prepared (use MODE and retry)

6 KEYB could not find the translation table for the selected code page in the resident keyboard table

7 Incorrect DOS version. (Do not use the KEYBOARD.SYS file provided with earlier versions of MS-DOS.)

Cautions

This command changes the keyboard layout, so some keys may be assigned to a new position or not exist at all after you have switched keyboard layouts. See Appendix B for the keyboard layouts.

Be careful how you use application software written in U.S. format. Your keys may send different signals (codes) to your operating system and application software.

Examples

To install the German keyboard layout with the multinational code page (850), type the following and press **Enter**:

```
KEYB GR,850
```

To use the United Kingdom layout number 166 with the multinational code page, type the following and press **Enter**:

```
KEYB UK,850,,166
```

You can omit the *filename* parameter in this example if the file KEYBOARD.SYS is stored in the root directory of your system disk. If the KEYBOARD.SYS file is stored in a subdirectory named C:\DOS4, type the following and press **Enter**:

```
KEYB UK,850,C:\DOS4\KEYBOARD.SYS,166
```

LABEL *External*

Purpose

Creates, displays, changes, or deletes a volume label on a disk.

Format

`[d:][path]LABEL [d:][volumelabel]`

Description

Use LABEL to identify your disks. The label can be up to 11 characters.

Do not use any of the following symbols as part of your disk label:

. " / \ [ ] : < > + = ; , ? * |

You can include space characters as part of a label.

If you do not specify a disk drive descriptor, MS-DOS assumes you are labelling the disk in the default drive.

If you type LABEL only and do not specify a volume label, you see this prompt:

```
Volume in drive X is xxxxxxxxxxx
Volume Serial Number is nnnn-nnnn
Volume label (11 characters, ENTER for none)?
```

or

```
Volume in drive X has no label
Volume Serial Number is nnnn-nnnn
Volume label (11 characters, ENTER for none)?
```

To give the disk a volume label or to change an existing volume label, type the new label and press **Enter**. The new label replaces the existing one.

To delete a volume label, press **Enter** only. You see the prompt:

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

Press **Y** to delete the label, or **N** to exit without deleting the label.

Cautions

Do not use LABEL with drives that have been assigned a path name with SUBST or another drive name with ASSIGN.

This command may not work in a network environment.

Example

To change the volume label of the diskette in drive B to AP DATA, type the following and press **Enter**:

```
LABEL B:AP DATA
```

To display or erase the volume label WS DATA from a diskette in drive B, type the following and press **Enter**:

```
LABEL B:
```

The screen displays the following:

```
Volume in drive B is WS DATA
Volume Serial Number is nnnn-nnnn
Volume label (11 characters, Enter for none)?
```

Press **Enter**. You see the following prompt:

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

Press **Y** and **Enter** if you want to delete the label, or press **N** and **Enter** to leave the label.

MEM

External

Purpose

Displays a summary of used and available memory, indicating the largest possible executable program size and, optionally, the locations and sizes of all device drivers and programs currently in memory.

Format

```
MEM [/PROGRAM]
```

or

```
MEM [/DEBUG]
```

Description

MEM presents a summary of current memory usage, showing how much conventional, extended, and expanded memory is available for application programs and data. You can choose from three levels of detail. For the most concise report, use MEM with no switches. For more information, use the /PROGRAM switch. For a complete breakdown of all objects in memory, use the /DEBUG switch. The following table describes the switches:

Switch	*Function*
/PROGRAM	Lists the starting addresses and sizes of all programs in memory.
/DEBUG	Lists the starting addresses and sizes of all objects in memory, including system device drivers, installed device drivers, and programs.

/PROGRAM and /DEBUG report addresses and sizes in hexadecimal notation. The figures for available memory are given in decimal.

Examples

To show the amount of memory currently available in your system, type the following and press **Enter**:

```
MEM
```

You see a report similar to this:

```
  650240 bytes total memory
  649216 bytes available
  191472 largest executable program size

 4030464 bytes total EMS memory
  524288 bytes free EMS memory

 3538944 bytes total extended memory
       0 bytes available extended memory
```

The first section of this report lists conventional memory (memory up to 640KB). The third line tells you how much memory is currently available for use by an application program and its data.

The second section lists expanded, or EMS, memory. The third section lists extended memory.

To show the starting addresses and sizes of all programs currently running, type the following and press **Enter**:

```
MEM /PROGRAM
```

To show the starting addresses and sizes of everything in memory, including all system device drivers and installed device drivers, type the following and press **Enter**:

```
MEM /DEBUG
```

MKDIR

Internal

Purpose

Creates (makes) a new subdirectory.

Format

```
MKDIR [d:]path
```

or

```
MD [d:]path
```

Description

Use MKDIR to create a subdirectory under the current directory or the specified directory. When you are in the root directory or one of its subdirectories, you can use MKDIR to create new subdirectories. If you do not specify a drive, the current drive is assumed. If your path starts with a backslash (\), MS-DOS starts its directory search with the root directory. If you omit the backslash, MS-DOS starts with the current directory.

Subdirectory names can be up to eight characters in length and can include all characters except a space or the following:

```
. " /\[ ] : | < > + = ; , ? *
```

You can create any number of subdirectories provided disk space is available. The maximum length of a path name is 63 characters, including backslashes.

A directory can include file and subdirectory names that also appear in other directories. Therefore, files having the same name can be stored in separate directories.

You can abbreviate MKDIR to MD.

Caution

Be careful when you create directories while an ASSIGN, JOIN, or SUBST command is in effect.

Examples

To create a subdirectory called PROGRAMS in your root directory (on the current drive), type the following and press **Enter**:

```
MKDIR \PROGRAMS
```

To create a subdirectory named BASIC, under a directory named PROGRAMS on drive B, type the following and press **Enter**:

```
MKDIR B:\PROGRAMS\BASIC
```

If your current directory is PROGRAMS, you only need to type:

```
MD BASIC
```

By leaving out the \, you tell MS-DOS to begin at the current directory.

MODE

External

Purpose

Sets the way a printer, the monitor, the keyboard, or a serial port operates, and prepares and controls code page switching.

Format

Parallel printer mode:

```
[d:][path]MODE LPTn[:] [cols][,[lines]
               [,[retry]]]
```

or

```
[d:][path]MODE LPTn[:] [COLS=cols] [LINES=lines]
               [RETRY=retry]
```

Serial port mode:

```
[d:][path]MODE COMn[:] baud[,[parity][,[data]
               [,[stop][,[retry]]]]]
```

or

```
[d:][path]MODE COMn[:] BAUD=baud [PARITY=parity]
               [DATA=data] [STOP=stop]
               [RETRY=retry]
```

Redirecting parallel printer output:

```
[d:][path]MODE LPTn[:]=COMn[:]
```

Display modes:

```
[d:][path]MODE displaymode[,L][,R][,T]
```

or

```
[d:][path]MODE displaymode,lines
```

or

```
[d:][path]MODE CON [COLS=width] [LINES=lines]
```

Setting keyboard repeat rates:

```
[d:][path]MODE CON: RATE=r DELAY=d
```

Device code page modes:

```
[d:][path]MODE device CODEPAGE PREPARE=((nnn)
                [d:][path]filename)
```

or

```
[d:][path]MODE device CODEPAGE PREPARE=
              ((nnn(,nnn...))[d:][path]filename)

[d:][path]MODE device CODEPAGE SELECT=nnn

[d:][path]MODE device CODEPAGE REFRESH

[d:][path]MODE device CODEPAGE [/STATUS]
```

You can use the following abbreviations in the code page commands:

CP instead of CODEPAGE
PREP instead of PREPARE
REF instead of REFRESH
SEL instead of SELECT
STA instead of STATUS

Description

Use this command to set or change the following aspects of your system's operation:

- Parallel printer selection, character width, and vertical spacing
- Serial port selection, baud rate, parity, data length, and stop bits
- Redirection of parallel printer output through serial ports for use by serial printers
- Monitor display format
- Keyboard repeat rates
- Preparation of one printer and monitor screen for code page switching.

Note
For a simpler method of using the MODE command, you can run the MENU program and select the Mode Settings option. See Chapter 5 for instructions.

To make these configuration changes to your printer, monitor, or serial port, enter the MODE command along with the appropriate parameters.

To select between printer, monitor, and serial port configurations, follow the MODE command with a device identifier. After the identifier, you give the parameters for the device itself.

Device	*Command/identifiers*
Printer	MODE LPT*n*: *(parameters)*
Serial port	MODE COM*n*: *(parameters)*
Printer redirect	MODE LPT*n*: *(parameter)*
Monitor	MODE *displaymode (parameters)*
Keyboard	MODE CON: *(parameters)*

If you use the parameters provided by MODE often, you may want to include the appropriate commands in your AUTOEXEC.BAT file or in batch files for other programs that require mode setting changes.

Changing Parallel Printer Settings With MODE

MODE offers these options for operating your parallel printer:

- Select between three logical printer devices: LPT1, LPT2, and LPT3
- Select the number of characters to print on each line, either 80 (standard) or 132 (condensed)
- Select the number of lines to print per vertical inch, either six or eight
- Select a retry mode when a printer error occurs.

To change parallel printer settings, use either of the following formats:

```
[d:][path]MODE LPTn[:] [cols][,[lines]
               [,[retry]]]

[d:][path]MODE LPTn[:] [COLS=cols] [LINES=lines]
               [RETRY=retry]
```

The device must be LPT1, LPT2, or LPT3. The colon after the device name is optional. Unless you add a special option card to your computer, the device selection is limited to LPT1; the other two options are not available.

The *cols* parameter specifies the number of characters the printer can print on a line and may be either 80 or 132. The default is 80.

The *lines* parameter specifies the vertical spacing of the printer, in lines per inch, and may be either 6 or 8. The default is 6.

The *retry* parameter tells MS-DOS what to do if the printer cannot accept characters sent from the computer (the printer is busy). The options are as follows:

Value	*Action*
B	Status check of printer returns busy.
E	Status check of printer returns error.
P	Same as B. (This option is provided for compatibility with earlier versions of MS-DOS.)
R	Status check of printer returns ready.
N or None	No retry action provided. This is the default condition.

If you specify B, E, P, or R for *retry*, you might create an endless loop while the computer tries to resend data to the printer. To terminate such a loop, press **Ctrl Break**.

In the first format above, you must specify the optional parameters in the order shown. In the second format, the order of the optional parameters does not matter. Parameters you omit retain their current values.

Caution

When using MODE to specify parallel printer settings, be sure the specified device is connected and on-line. If it is not connected or is off-line, your computer may "hang" for a few moments while MS-DOS attempts to send a command to the unavailable device.

Examples

To change the LPT1 printer's line spacing from 6 lines per inch to 8 lines per inch, type either of the following and press **Enter**:

```
MODE LPT1: ,8
MODE LPT1: LINES=8
```

You see a display similar to this:

```
LPT1: not rerouted
Printer lines per inch set
No retry on parallel printer time-out
```

Note that in the first example above, you must include the comma before the 8 because you omitted the preceding parameter (*cols*).

Changing Serial Port Operation With MODE

MODE offers these options in the operation of your serial port:

- Four logical port numbers: COM1, COM2, COM3, or COM4
- Nine baud rates, from 110 to 19200 (19200 baud is not supported on some computers)
- Odd, even, mark, space, or no parity (mark parity and space parity are not supported on some computers)

- A data length of five, six, seven, or eight bits
- One or two stop bits
- Various retry options.

To change serial port settings, use this format:

```
[d:][path]MODE COMn[:] baud[,[parity][,[data]
               [,[stop][,[retry]]]]]
```

or

```
[d:][path]MODE COMn[:] BAUD=baud [PARITY=parity]
               [DATA=data] [STOP=stop]
               [RETRY=retry]
```

The device must be COM1, COM2, COM3, or COM4. The colon after the device name is optional. Unless you add a special option card to your computer, the device selection is limited to COM1; the other three options are not available.

You must specify the *baud* parameter, which specifies the data transmission rate in characters per second. The options are 110, 150, 300, 600, 1200, 2400, 4800, 9600, and (on some computers) 19200. There is no default value.

The optional *parity* parameter specifies the error-checking method to use. The options are as follows:

O	Odd
E	Even
M	Mark (not available on some computers)
S	Space (not available on some computers)
N	None

The default is Even.

The *data* parameter specifies the number of bits per data word. The options are 5, 6, 7, and 8. The default is 7.

The *stop* parameter specifies the number of stop bits. The options are 1 and 2. The default is 1. (If *baud* is 110, the default is 2.)

The *retry* parameter tells MS-DOS what to do in case the serial port cannot accept data sent from the computer. The options are as follows:

Value	*Action*
B	Status check of port returns busy.
E	Status check of port returns error. This is the default condition.
P	Same as B. (This option is provided for compatibility with earlier versions of MS-DOS.)
R	Status check of port returns ready.
N or None	No retry action provided.

If you specify B, E, P, or R for *retry*, you might create an endless loop while the computer tries to resend data. To terminate such a loop, press **Ctrl Break**.

In the first format above, you must specify the optional parameters in the order shown. In the second format above, the order of the optional parameters does not matter. Parameters you omit retain their current values. If you omit a parameter using the first format, you must enter a comma in its place.

Examples

To change the baud rate for the modem attached to COM1 from 1200 to 2400 and the parity from even to none, type either of the following and press **Enter**:

```
MODE COM1 2400,N
MODE COM1 BAUD=2400 PARITY=N
```

You see a display similar to this:

```
COM1: 2400,n,7,1-
```

Changing Printer Output Direction

MODE permits you to redirect parallel printer output from one of three logical printer devices to one of four logical serial port devices. Instead of sending printer data out through the parallel printer port, you can direct it to your serial port for use by a serial printer or another serial device.

MS-DOS accepts the following device identifications (logical devices) for parallel and serial output:

Parallel Printer	*Serial Communications*
LPT1:	COM1:
LPT2:	COM2:
LPT3:	COM3:
	COM4:

Unless you install special option cards to enable additional output, devices LPT2:, LPT3:, COM2:, COM3:, and COM4: are not available. Although they exist logically in the operating system, the physical connections (to other printers or a second serial port) do not exist unless you add the option cards.

To redirect printer output, use this format:

```
[d:][path]MODE LPTn=COMn
```

Substitute device numbers to use other printer or serial port combinations.

Examples

To redirect printer output from LPT1: to COM1:, type the following and press **Enter**:

```
MODE LPT1:=COM1:
```

You see a display similar to this:

```
Resident portion of MODE loaded
LPT1: rerouted to COM1:
```

To cancel the redirection, so that output to LPT1: once again goes to LPT1:, type the following and press **Enter**:

```
MODE LPT1:
```

Changing Monitor Display Format With MODE

MODE offers these options for your monitor display:

- Choose 40- or 80-column width, in black and white, monochrome, or color display
- Align the screen image to the left [L] or right [R]
- Show a test pattern [T] to verify and adjust the screen image
- Change the number of text lines displayed.

To select a display mode and, optionally, change the alignment of the display, use the following format:

```
[d:][path]MODE displaymode[,L][,R][,T]
```

You must enter the *displaymode* parameter, but the L, R, and T parameters are optional. The L and R parameters are mutually exclusive.

The *displaymode* parameter offers various screen width and color options. You select one of the following:

Displaymode	*Description*
40	40-column screen width
80	80-column screen width
BW40	Black and white 40-column screen
BW80	Black and white 80-column screen (default)
CO40	Color 40-column screen
CO80	Color 80-column screen (used with color monitor)
MONO	Monochrome display adapter (which always has a display width of 80 characters per line)

The standard mode is BW80. If you have a color monitor, then CO80 is standard.

You can use the *alignment* parameters (L and R) to shift the screen image one column to the left or right. To move more than one column, repeat the command for each column you want to move. If you add the test parameter (T) after the L or R, a test line appears on the screen asking if you see the leftmost or rightmost character in the line. If you enter **N**, the line moves one column in the appropriate direction and the question appears again. When the image is aligned as you want it, type **Y**. Use the test parameter with a color monitor only. L, R, and T are valid only with a CGA or EGA adapter.

To select a display mode and change the number of screen lines, use the following format:

```
[d:][path]MODE displaymode,lines
```

The *displaymode* parameter may be 40, 80, BW40, BW80, CO40, CO80, or MONO (see the preceding table). The *lines* parameter may be 25, 43, or 50. Your monitor must be capable of displaying the number of lines specified in the *lines* parameter.

Another way to set the screen width and/or the number of screen lines is with the following format:

```
[d:][path]MODE CON [COLS=width] [LINES=lines]
```

In this format, the *width* parameter may be 40 or 80, and the *lines* parameter may be 25, 43, or 50. Your monitor must be capable of displaying the specified width and number of lines.

Examples

To see the effect of the MODE command on your color monitor, set the screen to display 40 columns instead of 80. Type the following and press **Enter**:

```
MODE 40
```

Your characters are now twice as wide. To restore the 80-character display, type the following and press **Enter**:

```
MODE 80
```

Setting the Keyboard Repeat Rates

You can use the MODE command to control the typematic action of your keyboard, that is, the rate at which keys repeat if you hold them down. Use the following command format:

```
[d:][path] MODE CON RATE=r DELAY=d
```

You must include both parameters.

The *r* parameter specifies the number of repetitions per second. The value of *r* may be from 1 to 32. The default value is 11 (approximately 4.6 repetitions per second).

The *d* parameter specifies the amount of time, in quarter-seconds, the keyboard waits before beginning to repeat. The value of *d* may be from 1 to 4. The default is 2.

Example

To double the default keyboard repetition rate from about 4.6 to about 9.2 repetitions per second, leaving the delay time unchanged, type the following and press **Enter**:

```
MODE CON RATE=19 DELAY=2
```

Setting the Code Page

You must use the MODE command to prepare and select a code page before you can use that keyboard layout with the device you select. Use the MODE command to:

- Prepare one or more code pages to be selected for use by your printer or monitor.
- Select which code page to use with which device. This activates the code page so you can use it on the designated device.
- Refresh (reselect) a code page that was deactivated because of an error. For example, turning off the printer may cause a code page to disappear.
- Display the current status of code page selections for a particular device.

Be sure to read Chapter 9 for complete information on code page switching.

Note
If you do not have an EGA adapter, you cannot display certain code pages on your monitor. If you do not have a printer that supports the code page you select, you cannot print all characters in that character set.

PREPARE—Before you can prepare a code page, you must have used the DEVICE command in the CONFIG.SYS file to activate the device driver(s) for which you want to use code page switching.

To prepare one or more code pages, use one of the following command formats:

```
[d:][path] MODE device CP PREP=((nnn)[d:]
                [path]filename)
```

or

```
[d:][path] MODE device CP PREP=((nnn[,nnn...])
                [d:][path]filename)
```

Type the MODE command, a space, the name of the device, a space, then CP PREP (or CODEPAGE PREPARE) followed by an equal sign. Use the first command format if you are preparing only one code page; use the second format if you are preparing more than one.

The *d:*, *path*, and *filename* identify the Code Page Information file for MS-DOS to use to prepare a code page for the device specified. These files have the extension .CPI.

You can enter one of the following devices in the command:

```
CON, LPT1, LPT2, LPT3
```

You can include only one device in a MODE command. If you want to use a code page or pages on another device, enter the MODE command again specifying the device.

You can enter one or more of the following code page numbers in the command:

437 United States—English
850 Multilingual
860 Portuguese
863 Canada—French
865 Nordic

To include more than one code page, list the numbers within parentheses and separate them with commas. For example, to prepare 437, 850, and 863, you would enter the code page part of the command as follows:

```
((437,850,863)[d:][path]filename)
```

If you later decide to change one of the code pages but want to keep the others, use a comma to represent the code pages you do not want to change. For example, to change 437 to 860, you would type the following:

```
((860,,)[d:][path]filename)
```

You must enter at least one code page number in the command.

Choose one of the following filenames for the code page information file:

File	*Code Page Information*
4201.CPI	IBM Proprinter
4208.CPI	IBM Proprinter X24 and XL24
5202.CPI	IBM Quietwriter III Printer
EGA.CPI	EGA display
LCD.CPI	IBM Convertible LCD

Be sure to include the proper pathname so MS-DOS can find the file.

SELECT—To select the code page to be used with a particular device, use the following command format:

```
[d:][path] MODE device CP SEL=nnn
```

Type MODE, a space, the name of the device (CON, LPT1, LPT2, or LPT3), a space, CP SEL (or CODEPAGE SELECT), an equal sign, and the number of the desired code page (437, 850, 860, 863, or 865). You can enter only one code page at time with the SELECT parameter.

Before you can select a code page, you must have done the following:

- Used the DEVICE command in the CONFIG.SYS file to activate the device driver(s) for which you want to use code page switching
- Used the PREPARE parameter with the MODE command to prepare the code page.

REFRESH—To refresh (reselect) a code page that has been deactivated, use this command format:

```
[d:][path] MODE device CP REF
```

Type MODE, a space, the name of the device (CON, LPT1, LPT2, or LPT3), a space, and CP REF (or CODEPAGE REFRESH). This command reactivates a code page you prepared and selected previously but was deactivated for some reason (such as if the printer was turned off).

/STATUS—Use the /STATUS switch if you want to display the code pages currently prepared and selected for a device:

```
[d:][path] MODE device CP /STATUS
```

/STATUS is optional; the following command does the same thing:

```
[d:][path] MODE device CP
```

Either command displays information similar to the following:

```
Active codepage for device LPT1 is 850 hardware
codepages:
   Codepage 437
prepared codepages:
   Codepage 437
   Codepage 850
   Codepage 860
MODE Status Codepage function completed
```

Examples

To prepare the Canada—French and Nordic code pages for the monitor (CON) using the code page information file EGA.CPI, type the following and press **Enter**:

```
MODE CON CP PREP=((863,865)EGA.CPI)
```

To select the United States code page for parallel printer LPT2, type the following and press **Enter**:

```
MODE LPT2 CP SEL=437
```

To refresh a code page for your printer, type the following and press **Enter**:

```
MODE LPT1 CP REFRESH
```

To display the current status of code pages for the monitor, type the following and press **Enter**:

```
MODE CON CODEPAGE /STATUS
```

MORE

External (filter)

Purpose

Displays long data listings one screen at a time.

Format

```
[d:][path]command ¦ [d:][path]MORE
```

or

```
MORE <filename
```

Description

When you specify the MORE command, MS-DOS reads the input and displays one screen of information at a time. The screen displays the following message on the last line:

```
-- MORE --
```

Press any key to display the next screen of information. Repeat this process until you have seen all the input data. MORE is useful for looking at a list of files in a large directory or viewing a long file.

Examples

To display the file MYFILES.TXT (on the default drive) one screen at a time, type the following and press **Enter**:

```
TYPE MYFILES.TXT ¦ MORE
```

To display the directory of files (on the default drive) one screen at a time, type the following and press **Enter**:

```
DIR ¦ MORE
```

To display the file LONGLIST on drive A one screen at a time, type the following and press **Enter**:

```
MORE <A:LONGLIST
```

NLSFUNC *External*

Purpose

Provides support for code page switching.

Format

`[d:][path]NLSFUNC [[d:][path]filename]`

Description

The NLSFUNC command enables support for extended country information and lets you use the CHCP command to select a code page (national language character set) for any device that supports code page switching. NLSFUNC also lets you select a country information file other than the one specified in your CONFIG.SYS file. Be sure to read Chapter 9 for complete information about code page switching.

If you do not specify a filename after NLSFUNC, the command uses the file defined by the COUNTRY command in your CONFIG.SYS file. If there is no COUNTRY command in your CONFIG.SYS file, COUNTRY is assumed to be 001 (United States), the default code page is 437, and the country information file is COUNTRY.SYS. NLSFUNC looks for the COUNTRY.SYS file in the root directory of the disk from which you boot MS-DOS. If you want to use a different country information file, include the drive, path, and name of the file in the command.

You use the CHCP command (described in this chapter) to select a code page for all devices defined with code page switching support. But you must first load the NLSFUNC command into memory before using the CHCP command. You may want to include the NLSFUNC command in your AUTOEXEC.BAT file.

You can install NLSFUNC using the INSTALL command in your CONFIG.SYS file. See Chapter 6 for information about how to do this.

Examples

To enable code page switching using the country information file in your CONFIG.SYS file, type the following and press **Enter**:

```
NLSFUNC
```

To enable code page switching using the country information file named COUNTRY2.SYS in the DOS directory on drive C, type the following and press **Enter**:

```
NLSFUNC C:\DOS\COUNTRY2.SYS
```

PATH

Internal

Purpose

Sets a search path for programs and batch files that are not in the current directory. Once you have used this command, you can run programs in other directories without specifying their pathnames.

Format

```
PATH [[d:]path1[;[d:]path2...]]

PATH;
```

Description

Use PATH to identify the directories that MS-DOS must search to locate external commands when they are not in the current directory. MS-DOS searches the directories in the path (in the specified order and on any drive) until it finds the command to be executed. PATH stays in effect until you reset the system or set another path.

This command is most useful when it is part of an AUTOEXEC.BAT file, which means that it is executed each time you boot MS-DOS. See Chapter 7 for further information on the AUTOEXEC.BAT file.

Use the format PATH *path1;path2;path3* to specify several paths by separating each path with a semicolon. MS-DOS searches the paths in the order you specify.

Type PATH; (with a semicolon only) to set a null path, instructing MS-DOS to search only the current directory for external commands.

Type PATH (with no pathnames) to display the current path.

MS-DOS searches the specified path only for command files (those having .COM, .EXE, and .BAT extensions) and then only when attempting to execute them. The current directory is not changed by this command.

PATH is similar to APPEND. The main difference is that PATH enables MS-DOS to locate executable (.BAT, .COM, or .EXE) files only. You can use APPEND for both executable files and data files.

Examples

To tell MS-DOS to search your \DOS\MENU directory (on drive C) for external commands, type the following and press **Enter**:

```
PATH C:\DOS\MENU
```

To search this path and two others, type the following and press **Enter**:

```
PATH C:\DOS\MENU;C:\SPRDSHT;C:\BASIC\NUMBERS
```

To display the current path, type the following and press **Enter**:

```
PATH
```

If the above paths have been entered, the screen displays the following:

```
PATH=C:\DOS\MENU;C:\SPRDSHT;C:\C:BASIC\NUMBERS
```

To have MS-DOS search only the current directory and *not* the previously specified path or paths, type the following and press **Enter**:

```
PATH;
```

PRINT *External*

Purpose

Prints a text file (or set of files) on a printer.

Format

```
[d:][path]PRINT [/D:device][/B:buffersize]
                [/U:busytick][/M:maxtick]
                [/S:timeslice][/Q:queuesize]
                [/T][/C][/P]
                [[d:][path][filename]...]
```

See the example section for further information on the placement of the switches.

Description

Use PRINT to output files to a printer while you continue other processing with your computer. The files to be printed are stored in a print queue.

You can use these switches with the PRINT command:

Switch	*Function*
/D:device	Specifies the print device; the default is PRN.
/B:*buffersize*	Sets the size in bytes of the print buffer (the default is 512). Increasing the value of *buffer size* may enhance the speed of the PRINT command. The maximum size of the print buffer depends on the amount of memory available.
/U:*busytick*	Specifies the number of computer clock ticks (busy ticks) PRINT will wait until the printer is available; the default is 1. If PRINT waits longer than this value, it gives up its timeslice (/S).
/M:*maxtick*	Specifies the number of clock ticks that PRINT can use to print a character. The value can be from 1 to 255 clock ticks; the default is 2.

Switch	*Function*
/S:timeslice	Specifies the number of clock ticks for a timeslice. The value can be from 1 to 255 timeslices; the default is 8.
/Q:queuesize	Specifies the number of files allowed in the print queue. The value can be from 4 to 32 files; the default is 10. To change the value, you must give the PRINT command without any filenames, for example: `PRINT /Q:32.`
/T	Selects terminate mode; deletes all files in the print queue waiting to be printed. A message alerts you to the deletion.
/C	Selects cancel mode; suspends printing of files in the print queue. The specified file and all following files on the command line are removed from the print queue until you type a /P switch.
/P	Selects print mode; enters files in the print queue. The specified file and all following files on the command line are entered in the print queue until you type a /C switch.

You can enter the /D, /B, /U, /M, /S, and /Q switches only the first time you give the PRINT command after starting MS-DOS. Once you set these switches, they remain in effect until you turn off or reset the computer.

If you do not specify the device name (/D), the first time you use the PRINT command, you see this prompt:

```
Name of list device [PRN]:
```

You can specify any output list device, such as LPT1, LPT2, LPT3, PRN, COM1, COM2, COM3, COM4, or AUX. Then press **Enter**. Press just **Enter** to select the default PRN device. The device you select must be attached to your system.

The files are queued for printing in the order you enter them. Any files in the print queue must remain unaltered on the disk in the specified drive until printing is complete.

You can use wildcard characters in the filenames. Also, you can specify more than one filename, each with the appropriate switches. Each print queue entry can contain a maximum of 64 characters, including the drive descriptor.

Type PRINT without a file specification to display the contents of the print queue on your screen without affecting the queue's contents.

After a file is entered in the print queue, you can change the current directory without affecting the print status of files in the queue.

Caution

If you have reassigned disk drive descriptors with ASSIGN, cancel the assignments before you run PRINT.

Examples

To set the size of the print buffer to 800 bytes, type the following and press **Enter**:

```
PRINT /B:8ØØ
```

To empty the print queue, type the following and press **Enter**:

```
PRINT /T
```

To remove these three files from the print queue, type the following and press **Enter**:

```
PRINT A:TEMP1.TST /C A:TEMP2.TST A:TEMP3.TST
```

To remove TEMP1.TST from the queue, and then add TEMP2.TST and TEMP3.TST to the queue, type the following and press **Enter**:

```
PRINT TEMP1.TST /C TEMP2.TST /P TEMP3.TST
```

Note

Each switch affects the file *before* it and all the files after it until the file *before* the next switch. In the last example, the /C switch affects TEMP1.TST and the /P switch affects both TEMP2.TST and TEMP3.TST.

PROMPT

Internal

Purpose

Changes or resets the MS-DOS command prompt.

Format

PROMPT [*string*]

Description

Use PROMPT to change the MS-DOS system prompt from A> to any string of characters. Whatever you specify for the string parameter becomes the command prompt until you specify a different prompt, or reset or turn off the computer.

Type PROMPT only (without a string of characters) to reset the prompt to the default drive identifier.

You can use the following characters in your PROMPT command to specify special prompt information. You must precede these characters with a dollar sign ($) in the string.

Specify this character:	*To get this in your prompt:*
_ *(underscore)*	Enters a carriage return/line feed sequence (goes to beginning of a new line)
b	Enters a | character
d	Enters the current date
e	Enters an ASCII code of hex 1B (used in ANSI character sets)
g	Enters the > character
h	Enters a backspace; the preceding character is erased
l	Enters the < character
n	Enters the default drive

Specify this character:	*To get this in your prompt:*
p	Enters the current directory of the default drive
q	Enters the = character
s	Enters a leading space only
t	Enters the current time
v	Enters the MS-DOS version number

If your system has an ANSI.SYS driver set in CONFIG.SYS, you can use escape sequences in your prompts.

Examples

To set the default drive letter prompt without any symbol, type the following and press **Enter**:

```
PROMPT $n
```

The screen displays A (if A is the default drive).

To set the prompt to the current drive and directory followed by the > symbol, type the following and press **Enter**:

```
PROMPT $p$g
```

If your current directory is WORD\WORK on the B drive, the screen displays the following:

```
B:\WORD\WORK>
```

To include a message such as READY as well as the > symbol, type the following and press **Enter**:

```
PROMPT READY$g
```

The screen displays:

```
READY>
```

To set a two-line prompt with the time and date, type the following and press **Enter**:

```
PROMPT Time=$t$_Date=$d
```

The screen displays the following prompt with the current time and date:

```
Time=13:05:23.88
Date=Thu  3-24-88
```

If you have an ANSI escape sequence driver, you can set the prompts in reverse video mode and return to normal video mode for other text. Type the following and press **Enter**:

```
PROMPT $e[7m$n:$e[m
```

To reset the prompt to the default drive descriptor, type the following and press **Enter**:

```
PROMPT
```

The screen displays the default command prompt, for example:

```
B>
```

RECOVER *External*

Purpose

Recovers a file or an entire disk containing bad sectors.

Format

```
[d:][path]RECOVER [d:][path]filename
```

or

```
[d:][path]RECOVER d:
```

Description

Use RECOVER to check individual files for damage and to reconstruct the directory of a damaged disk. If a sector is bad, you can recover either the file containing that sector or the entire disk. When you recover a file, any bad sectors are deleted from the file. Recover the entire disk if the bad sector is in the directory.

Type RECOVER *filename* to recover a particular file. MS-DOS reads the file sector by sector and skips the bad sector(s). When MS-DOS finds the bad sector, it marks it and no longer writes data to that sector.

Type RECOVER *d:* to recover a disk (*d* represents the descriptor of the drive containing the disk). When you recover an entire disk, all filenames change to FILE*nnnn*.REC (*nnnn* represents a sequential number starting with 0001). The date and time change to your current settings.

Cautions

If you have reassigned disk drive descriptors with ASSIGN, cancel the assignments before you run RECOVER. Otherwise, the program does not work properly.

This command may not work in a network environment.

Examples

To recover the file APFILE on drive A, type the following and press **Enter**:

```
RECOVER A:APFILE
```

The file APFILE is read sector-by-sector and bad sectors are skipped.

To recover the contents of the entire diskette in drive B, type the following and press **Enter**:

```
RECOVER B:
```

RENAME

Internal

Purpose

Renames the specified file or files.

Format

```
RENAME [d:][path]filename1 filename2
```

or

```
REN [d:][path]filename1 filename2
```

Description

Use RENAME to change the name of one or more files. You change the name of the old file to that of the new file. You can type the command name either as RENAME or REN.

You cannot specify a drive or path with the new filename. The file remains on the drive and in the directory where it currently resides.

You can use the wildcard characters * and ? to rename a group of files. When you use the ? wildcard character in the filename, the corresponding character positions do not change.

You cannot rename a file using the name of an existing file in the directory.

Examples

To change MYFILE.TXT to YOURFILE.DOC on the current drive, type the following and press **Enter**:

```
REN MYFILE.TXT YOURFILE.DOC
```

To change all files with an .LST extension to a .PRN extension, keeping the same filenames, type the following and press **Enter**:

```
REN *.LST *.PRN
```

To rename the file ABODE on drive B to ADOBE, type the following and press **Enter**:

```
REN B:ABODE ?D?B?
```

REPLACE *External*

Purpose

Selectively replaces or adds files including subdirectories.

Format

```
[d:][path]REPLACE [d1:][path][filename]
                  [d2:][path][/A][/P][/R][/S][/U]
                  [/W]
```

Description

Use REPLACE to replace files on the target disk with files of the same name from the source disk. You can also add files, copying them from the source to the target.

> **Note**
> An easier way to use the REPLACE command is through the MENU utility. See Chapter 5 for instructions.

In the format above, *d1* is the source and *d2* is the target. You must specify at least a drive, path, or filename for the source.

For the target, you can specify a drive or path, but you do not specify a filename. If you omit the target, the specified source files replace files in the current directory of the default drive.

Whether replacing or adding files, you can copy files to subdirectories on the target by using the /S switch. You can use wildcards to replace groups of similarly named files.

You cannot use REPLACE to update hidden files or system files.

You can use the following switches with REPLACE:

Switch	*Function*
/A	Copies all files that do not exist on the target. /A only *adds* files, but does not overwrite existing files. You cannot use /A together with /S.

Switch	*Function*
/P	Prompts you to reply Yes or No before replacing each file on the target.
/R	Replaces files on the target that are read-only. Without /R, read-only files cannot be overwritten.
/S	Replaces files in all subdirectories on the target. You cannot use /S and /A together.
/U	Replaces only those files on the target that have a later date and time on the source. You cannot use /U and /A together.
/W	Waits for you to insert a diskette before beginning to search for source files.

REPLACE sets the following exit codes, which may help you perform error-detection batch processing:

0 Normal termination of command

2 File not found

3 Path not found

5 Access denied

8 Insufficient memory

11 Command line error

15 Invalid drive.

Examples

To replace existing files in all subdirectories on drive C with files from drive A, type the following and press **Enter**:

```
REPLACE A:*.* C:/S
```

To *add* files from drive A to the \WORK subdirectory on drive C, type the following and press **Enter**:

```
REPLACE A: C:\WORK /A
```

To replace the file LONG.LST on drive B with the file on the source diskette in drive A and have the system wait for you to insert a diskette, type the following and press **Enter**:

```
REPLACE A:LONG.LST B: /W
```

The screen displays the following:

```
Press any key to continue . . .
```

RESTORE *External*

Purpose

Restores files copied with the BACKUP command.

Format

```
[d:][path]RESTORE d1: [d2:][path]filename[/M]
                  [/N][/P][/S][/A:date]
                  [/B:date][/E:time][L:time]
```

Description

Use RESTORE to restore files that have been copied with the BACKUP command. (BACKUP creates specially-formatted files that are not usable until you restore them with the RESTORE command.)

Note
An easier way to use the RESTORE command is through the MENU utility. See Chapter 5 for instructions.

In the command format above, *d1* is the source drive, the one that contains the BACKUP files, and *d2* is the target drive to which you want to restore the files.

For the source, specify only the drive that contains the backup files. For the target, you can specify a drive, path, and/or filename. If you omit the target, the backup files are restored to the current directory of the default drive. If you specify a path for the target, you must restore the files to the same directory they were in when BACKUP copied them.

You can use wildcards to restore groups of similarly named files.

You can use the following switches with RESTORE:

Switch	*Function*
/M	Restores only those files that have been modified since the last BACKUP. If you use /M with /N, /M is ignored.

Switch	*Function*
/N	Restores only those files that no longer exist on the target disk.
/P	Prompts you to reply Yes or No before restoring files that have been changed since the last backup, or that are marked read-only (see ATTRIB).
/S	Restores files in all subdirectories beyond the specified directory in addition to the files in the specified or current directory. (If the specified or current directory is the root directory, /S restores files in all subdirectories.)
/A:*date*	Restores only those files that were modified on or *after* the specified date.
/B:*date*	Restores only those files that were modified on or *before* the specified date.
/E:*time*	Restores only those files that were modified on or *earlier* than the specified time.
/L:*time*	Restores only those files that were modified on or *later* than the specified time.

Note
The date and time formats depend on the COUNTRY code selected.

RESTORE prompts you to insert the source diskettes in the order they were backed up with BACKUP. If you insert the wrong diskette, RESTORE prompts you to insert the next diskette until you insert the correct one.

If you use wildcards in the filename or extension, RESTORE restores all the files that match the wildcard files; then it prompts you to insert the next diskette.

If you are sharing files, you can restore only those files that you normally can access. If you try to restore a file that you are not permitted to access, the screen displays the following message:

```
PATHNAME\FILENAME.EXT
Not able to restore at this time
```

RESTORE sets the following exit codes, which may help you perform error-detection batch processing:

0 Normal termination of command

1 No files were found to restore

3 Terminated by user

4 Terminated because of an error.

Caution

Do not use RESTORE while the ASSIGN, JOIN, or SUBST command is in effect.

Examples

To restore all files including files in subdirectories from backup diskettes on drive A to the hard disk (drive C), type the following and press **Enter**:

```
RESTORE A: C:*.* /S
```

To restore FILEIT.DOC from a backup diskette in drive A to drive B, type the following and press **Enter**:

```
RESTORE A: B:FILEIT.DOC
```

To restore files that have a .DOC extension from backup diskettes in drive A to drive C, type the following and press **Enter**:

```
RESTORE A: C:*.DOC
```

To restore all files from backup diskettes in drive A to drive C, and have the system prompt you if any files on drive C have changed since the last BACKUP or if any files are marked read-only, type the following and press **Enter**:

```
RESTORE A: C:/P
```

RMDIR *Internal*

Purpose

Removes (deletes) an empty subdirectory.

Format

```
RMDIR [d:]path
```

or

```
RD [d:]path
```

Description

Use RMDIR to remove an empty directory from a directory tree. The directory must be empty except for the . and .. shorthand symbols. Therefore, you must delete all files before you delete the directory. (You can do this by typing DEL `*.*` when you are in the directory you want to delete.)

When you specify a path, RMDIR removes the last directory name in the path. If you do not specify a drive, the current drive is assumed.

You cannot remove the current directory or the root (\) directory. Also, you cannot remove subdirectories that contain hidden files.

Cautions

Be careful when removing a directory while an ASSIGN or a JOIN is in effect. You cannot remove a directory if it has been substituted (SUBST).

Example

To remove the \OUT subdirectory located under the ELECMAIL directory, which is in the root directory (on the current drive), type the following and press **Enter**:

```
RMDIR \ELECMAIL\OUT
```

SELECT *External*

Purpose

Installs MS-DOS on a hard disk or a series of diskettes.

Description

SELECT installs MS-DOS on a hard disk or a series of diskettes. It partitions and formats a hard disk if necessary, and it creates default AUTOEXEC.BAT and CONFIG.SYS files.

SELECT should not be used from the MS-DOS command line. See the MS-DOS 4.01 Installation Guide.

SET

Internal

Purpose

Sets or displays the environment string values in memory.

Format

SET [*name*=[*parameter*]]

Description

Use SET to insert a string into the command processor's environment or to set one string equal to another. A copy of the entire series of strings in the environment is made available to all commands and applications. SET also displays the current setting of environment strings.

SET inserts the entire string, beginning with *name*, into a block of memory reserved for environment strings. Any lowercase letters in the *name* are converted to uppercase letters when added to the environment (including foreign language characters). The remainder of the line is inserted as typed. If the name already exists in the environment, it is replaced with the new parameter.

If you enter SET only with no *name*, the current setting of environment strings is displayed.

If you specify a *name* but no *parameter*, SET removes the current setting of *name*=*parameter* from the environment.

Do *not* use SET to add the PROMPT and PATH commands to the environment. MS-DOS automatically does this when these commands are entered.

One string in the environment is always COMSPEC=*parameter*. (MS-DOS places it in the environment when you start or reset your system.) This parameter describes the path that MS-DOS uses to reload the command processor when necessary.

You can use the SET command in batch processing and define your replaceable parameters with names instead of numbers. This eliminates the need to change replaceable parameter names in each batch file.

Examples

To add the string PROG=\LEVEL1 to the environment, type the following and press **Enter**:

```
SET PROG=\LEVEL1
```

In the above example, when an application program receives control, it could search the environment for the name PROG and use the supplied parameter as the directory name to use for its files.

To remove PROG=\LEVEL1 from the environment, type the following and press **Enter**:

```
SET PROG=
```

To add the string ABC=xyz to other strings already in the environment, type the following and press **Enter**:

```
SET ABC=xyz
```

In the above example, you can enter keywords and parameters that have no meaning in MS-DOS, but can be found and interpreted by applications designed to examine the environment.

SHARE *External*

Purpose

Installs file sharing and locking protection in network situations.

Format

`[d:][path]SHARE [/F:space][/L:locks]`

Description

You can use the SHARE command only when networking is active. You normally include the command in an AUTOEXEC.BAT file to install shared files. See the manual for your networking software to learn about shared files.

You can use these switches with SHARE:

Switch	*Function*
/F:*space*	Allocates file space
/L:*locks*	Allocates the number of locks

Use the /F switch to allocate file space (in bytes) for the area MS-DOS uses to record file sharing information. The space required for each open file is the length of the full filename plus 11 bytes. The default value is 2048 bytes.

The /L switch allocates the number of file locks. The default is 20 locks.

You can install SHARE with an INSTALL statement in your CONFIG.SYS file. See Chapter 6 for information about how to do this.

You must install SHARE if you use disk partitions larger than 32MB. A copy of SHARE.EXE must reside either in the root directory of the disk from which you boot MS-DOS or in the directory specified by the SHELL= statement in your CONFIG.SYS file.

If your CONFIG.SYS does not override the default value for FCBS (4,0), SHARE adjusts FCBS to 16,8.

Once you use the SHARE command in an MS-DOS session, all read and write requests are checked by MS-DOS against the allowed number of FCBs specified in your CONFIG.SYS file. If you try to load SHARE again, the following message is displayed:

```
SHARE already installed
```

Example

To reserve 1500 bytes for file sharing and allow for 10 locks, type the following and press **Enter**:

```
SHARE /F:15ØØ/L:1Ø
```

SORT

External (filter)

Purpose

Sorts data from an input device and then writes it to the output device, such as a screen, file, or printer.

Format

`[d:][path]SORT [/R][<filename1 [>filename2]]`

Description

SORT is a filter program that reads input from your keyboard (or standard input device), sorts the data, and then writes it back to your monitor (or standard output device).

SORT arranges data in ASCII order (with numbers appearing before letters), except that lowercase letters are treated as equivalent to uppercase letters. Sort organizes characters with ASCII values greater than 127 using rules defined by the COUNTRY command currently in effect.

You can redirect your input and output to sort the data in a file and then write the sorted data to a new output file or system device. If you redirect input and output to files, the input and output files must have different names.

You can use two switches with SORT:

Switch	*Function*
/R	Reverses the sort order; output sorts in order from Z to A
/+*n*	Starts sorting in column *n*; SORT normally begins sorting in column 1

Examples

To read the file UNSORT.TXT, reverse the sort, and then write the output to a file named SORT.TXT, type the following and press **Enter**:

```
SORT /R <UNSORT.TXT >SORT.TXT
```

You can combine SORT with other commands. For example, to sort the output of the DIR command starting in column 14, and display output on your screen, type the following and press **Enter**:

```
DIR ¦ SORT /+14
```

The result of this command is a directory sorted by file size (represented in column 14).

You can add the MORE command to SORT. To read your sorted directory one screen at a time, type the following and press **Enter**:

```
DIR ¦ SORT /+14 ¦ MORE
```

SUBST *External*

Purpose

Substitutes a drive letter for a pathname.

Format

```
[d:][path]SUBST
```

or

```
[d:][path]SUBST d1: [d2:]path
```

or

```
[d:][path]SUBST d: /D
```

Description

Use SUBST to substitute a single drive letter for an entire pathname. This is useful if you use application programs that do not allow pathnames. With these programs, all disk activity must take place in the current directory. To get around this limitation, use the SUBST command.

The drive letters you can use with SUBST depend on the value of LASTDRIVE in the CONFIG.SYS file. If a CONFIG.SYS file does not exist or does not contain the LASTDRIVE parameter, the default value is E. In this case, any letter from A to E can be used by SUBST. If you use a lot of substitutions, increase the value of LASTDRIVE. (See Chapter 6 for a complete description of the CONFIG.SYS file.)

Even though you can use any letter in the range, do not use the letter assigned to an existing drive. If you do use an existing drive, the specifier will no longer refer to the drive but to the path instead.

In the command format above, *d1:* represents the letter assigned to the pathname. *d2:* must not be the same as the default drive. Specify *d2:* only if the pathname is on a different drive. The pathname must start from the root directory.

Type SUBST without any parameters to display the current substitutions.

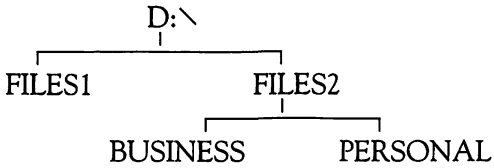
Use the /D switch to cancel a substitution. If more than one substitution has been made, use SUBST *d*:/D to cancel each value of *d*:.

Caution

Do not use ASSIGN, BACKUP, CHKDSK, DISKCOMP, DISKCOPY, FDISK, FORMAT, JOIN, LABEL, RECOVER, RESTORE, SELECT, or SYS while a substitution is in effect.

Examples

Assume the following directory structure for drive D:



If you are running the application program from drive C (a hard disk), to access the path D:\FILES1, substituting the letter E, type the following and press **Enter**:

```
SUBST E: D:\FILES1
```

If you log on or refer to drive E from your application program, all activity takes place through path D:\FILES1.

To cancel this substitution, type the following and press **Enter**:

```
SUBST E: \D
```

To access the path D:\FILES2\PERSONAL, substituting the letter F, type the following and press **Enter**:

```
SUBST F: D:\FILES2\PERSONAL
```

SYS

External

Purpose

Copies the system files from the boot-up drive to the specified disk.

Format

`[d:][path]SYS d:`

Description

Use SYS to transfer MS-DOS system files from the disk in the boot-up drive (the one from which you loaded MS-DOS) to the diskette or hard disk you specify. The system files are hidden and do not appear when you give the DIR command. The SYS command can either update an earlier version of MS-DOS on a disk or copy the system to a formatted disk containing no files.

SYS does not transfer COMMAND.COM, the command processor. You must use the COPY command to transfer COMMAND.COM into the destination disk's root directory.

You must include the drive descriptor of the target disk. The target disk must be formatted.

You may optionally include a source drive and path; if you do not, SYS assumes the source is the current drive and directory.

This command may not work in a network environment.

Example

To copy the system files from the disk in the boot-up drive to the disk in drive B, type SYS B: and press **Enter**. You see the message

```
System transferred
```

TIME

Internal

Purpose

Displays or sets the system time.

Format

```
TIME [hh:mm[:ss[.cc]]]
```

Description

Use TIME to set a new system time. The file directory then records the time on all files when you create or modify them.

The TIME command updates the CMOS time if your computer has a real-time clock.

To display the current system time, type TIME and press **Enter**. A message similar to this appears:

```
Current time is Ø9:11:19.7Ø
Enter new time:
```

Press **Enter** if you do not want to change the current time. Or, enter a different time and then press **Enter**. MS-DOS redisplays the command prompt.

You can enter the time on a 24-hour or 12-hour basis, depending on the current COUNTRY setting. If you use the 12-hour basis, add the letter *a* or *p* after the time to indicate a.m. or p.m.

Any numeric component of the time that is omitted is considered to be 0. For example, the following commands are equivalent:

```
TIME 7p
TIME 7:ØØ:ØØ.ØØp
```

You must separate the hour and minute entries with a colon. If you enter seconds, precede the number with a colon. If you enter hundredths of seconds, precede the number with a period.

You can change the format of the time display with the COUNTRY command in your CONFIG.SYS file.

Example

To set the time for 9:30 in the morning, you can type any one of the following:

```
9:3Ø:ØØ
9:3Ø
Ø9:3Ø:ØØ
Ø9:3Ø
```

TREE

External

Purpose

Diagrams all or part of a disk's subdirectory structure and optionally lists the files in all directories.

Format

`[d:][path]TREE [d:][path][/A][/F]`

Description

TREE displays a directory-structure diagram, using block graphic characters and indenting each subdirectory level.

If you do not include a drive designator or pathname, the diagram begins at the current directory and includes the current directory's subdirectories.

TREE also lists the volume label (if any) and serial number (if any) of the disk.

The following switches are available:

Switch	*Function*
/A	Generates the diagram without using block graphic characters. Use this switch when redirecting output to a printer that does not support the block graphic characters.
/F	Includes the filenames of each subdirectory.

Examples

To obtain a directory tree diagram listing all directories on the diskette in drive A, type the following and press **Enter**:

```
TREE A:\
```

To list all directories and files of the current directory, type the following and press **Enter**:

```
TREE /F
```

To list all files and directories on drive C and pause the display after each page, type the following and press **Enter**:

```
TREE C: /F | MORE
```

To print a directory diagram of the diskette in drive B, type the following and press **Enter**:

```
TREE B:\ >PRN
```

TYPE

Internal

Purpose

Displays the contents of a file on your screen.

Format

`TYPE [d:][path]filename`

Description

Use TYPE to display the contents of a file without modifying it. You must specify a filename and you cannot use wildcard characters. The specified file is displayed on the screen or the standard output device.

When you use TYPE, data is unformatted. Normal tabs expand and are shown at every eighth column. Some word processors embed tabs instead of spaces to save disk space.

The TYPE command is meaningful only with text files. If you give the TYPE command for a binary file, you may see strange characters that represent control characters such as bells, form feeds, and escape sequences.

You can use the MORE command to display the file one screen at a time, and you can press **Ctrl Print Screen** or **Ctrl P** to print the file as it appears on the screen.

Examples

To display a file on drive B called EPSON.TXT, type the following and press **Enter**:

```
TYPE B:EPSON.TXT
```

To print a file named TICKET in the current directory while it displays on the screen, press **Ctrl P**. Then type the following and press **Enter**:

```
TYPE TICKET
```

VER

Internal

Purpose

Displays the MS-DOS version number.

Format

```
VER
```

Description

Use the VER command to find out which version of MS-DOS you are using. MS-DOS displays the version number on your screen.

Example

To display the version of MS-DOS you are using, type VER and press **Enter**. You see a message similar to this:

```
MS-DOS Version 4.Ø1
```

VERIFY

Internal

Purpose

Turns on or off the verify function, which verifies each disk write.

Format

```
VERIFY [ON]
```

or

```
VERIFY [OFF]
```

Description

Use VERIFY to verify that all your files are written correctly to disk. When you specify `VERIFY ON`, MS-DOS checks that the disk controller did not make any errors during the disk writing procedure. This command, however, does not guarantee that the data is correct.

The /V switch with the COPY or XCOPY command performs the same function as VERIFY ON.

When you set `VERIFY ON`, it remains in effect until you turn it off or reset or turn off the computer. When VERIFY is on, MS-DOS takes a little longer when writing to disk.

The default setting of VERIFY is off. To display the current VERIFY setting, type `VERIFY`, with no parameters.

Examples

To verify that data written to disk is correctly recorded, type the following and press **Enter**:

```
VERIFY ON
```

To display the current setting, type the following and press **Enter**:

```
VERIFY
```

If you had set VERIFY ON, the screen displays the following:

```
VERIFY is on
```

VOL

Internal

Purpose

Displays a disk volume label.

Format

```
VOL [d:]
```

Description

Use VOL to display the volume label and serial number (if any) of the disk in drive *d*. If you do not specify a disk drive descriptor, VOL displays the volume label of the disk in the default drive.

If the disk has no label, the screen displays this message:

```
Volume in drive d has no label
Volume Serial Number is nnnn-nnnn
```

You can use LABEL to create or rename the volume label of a disk.

Example

To see the volume label of a diskette in drive B, type the following and press **Enter**:

```
VOL B:
```

The screen displays a message similar to this:

```
Volume in drive B is DATAFILESØØ1
Volume Serial Number is 1234-ABCD
```

XCOPY

External

Purpose

Selectively copies files and, optionally, subdirectories.

Format

```
[d:][path]XCOPY d1:[path][filename]
                [d2:][path][filename]
                [/A][/E][/M][/P][/S][/V][/W]
                [/D:date]
```

Description

Use the XCOPY command to copy:

- Individual files from one disk to another or to the same disk
- A group of files by indicating a path or using wildcard characters
- One or more files under new names.

Note
An easier way to use the XCOPY command is through the MENU utility. See Chapter 5 for instructions.

XCOPY differs from COPY in that you can copy subdirectories and, if necessary, create them as you copy.

With the /S and /M switches, you can use XCOPY for backup operations. Unlike BACKUP, however, XCOPY cannot divide a source file between two or more target diskettes.

Use XCOPY instead of DISKCOPY if you are copying all the files from one disk to another that is not the same format (from a 360KB diskette to a 1.2MB diskette, for example). Because DISKCOPY copies all the data by tracks, it cannot copy between two disks that are not the same format.

In the command format above, *d1* is the source and *d2* is the target. You must include the source drive designator. If you specify the target, you may omit the source file specification; in that case, XCOPY assumes the source is all files (*.*) in the source drive's current directory.

You can use wildcards when specifying the source file.

If you do not specify a path for the target copies, XCOPY copies the files to the current directory. If you do not include a filename, or if you enter *.*, the target copies have the same name as the source files.

You can use the following switches with XCOPY to select exactly which files to copy and how they should be copied:

Switch	*Function*
/A	Copies files whose archive bit is set, but does not change the attribute of the *source* file.
/E	When used with /S, creates subdirectories on the target even if no files are stored in them.
/M	Copies files whose archive bit is set but, unlike /A, turns off the archive bit of the *source* file.
/P	Prompts you to reply Yes or No before copying each file.
/S	Copies files from the source directory and from all the subdirectories below the source directory. /S creates subdirectories on the target as it copies files, but does not create a subdirectory that contains no files. To do this, use /E as well as /S.
/V	Verifies that the sectors written on the target diskette are recorded properly: /V causes XCOPY to run slower.

Switch	*Function*
/W	Tells XCOPY to wait for you to insert diskettes before beginning to search for source files.
/D:*date*	Copies files whose date is the same as or later than the specified date. (Date format depends on the COUNTRY code selected.)

You can give a file a new name with XCOPY by specifying a target filename that is different from the source filename.

You cannot use XCOPY to copy from or to the reserved device names CON and LPT1.

XCOPY sets the following exit codes (which you can use in error-detection batch processing):

0 Normal completion of command

1 No files were found to copy

2 Terminated by user (**Ctrl C** or **Ctrl Break**)

3 General error

4 INT 24 error

Examples

To copy all the files in all the subdirectories of drive C to drive A, type the following and press **Enter**:

```
XCOPY C:\ A:\ /S
```

XCOPY copies all the files, creating subdirectories as required.

If drive C contains some subdirectories with no files in them, and you want those subdirectories to be created on drive A along with the other subdirectories, type the following and press **Enter**:

```
XCOPY C:\ A:\ /S/E
```

To copy all files in the \WORK subdirectory modified on or after March 15, 1989 to drive A, type the following and press **Enter**:

```
XCOPY C:\WORK A:/D:Ø3-15-89
```

Chapter 5

Epson Menu Utilities

Introduction 5-3
MENU 5-3
- Selecting Options From Menus 5-6
- File Utilities 5-8
- Disk Utilities 5-25
- Mode Settings 5-31
- HELP 5-43
- Enter DOS Command 5-44

SETPRINT 5-45
- Options 5-46
- Cautions 5-48

Introduction

Epson has included two utilities to make MS-DOS easier to use:

- MENU provides simplified menu screens that present the options of certain complex MS-DOS commands.
- SETPRINT displays a menu you can use to control your printer's operation and select typefaces and spacing values.

This chapter describes the use of MENU and SETPRINT.

MENU

The MENU program displays a menu from which you can select any MS-DOS command, and it provides simple menus for running certain complex commands. To run the program, log onto the directory containing the MENU program or enter the correct pathname and then type `MENU` and press **Enter**. The screen displays this main menu:

```
      EXIT
 File Utilities
 Disk Utilities
 Mode Settings
      Help
Enter DOS Command
```

Each option in the main menu (except EXIT and Enter DOS command) displays a submenu of MS-DOS functions. Use ↓ or ↑ to move the highlighted cursor block to the desired option. Then press **Enter** to display that submenu. Select the EXIT option when you want to leave the menu utility.

Each option in the submenus corresponds to an Epson program and/or an MS-DOS command. The following table shows which commands are activated by the options in the menus.

MENU commands

Main Menu	Submenu Option	Epson Program	MS-DOS Command
File Utilities	Backup Files	SETBCKUP	BACKUP
	Restore Files	SETRSTOR	RESTORE
	Replace Files	SETRPLCE	REPLACE
	Compare Files	SETFC	FC
	Change File Attributes	SETATTR	ATTRIB
	Copy Files/ Directories	SETXCOPY	XCOPY
Disk Utilities	Check Disk	DU	CHKDSK
	Disk Copy	DU	DISKCOPY
	Disk Compare	DU	DISKCOMP
	Format Disk	DU	FORMAT
Mode Settings		SETMODE	MODE
Help			HELP
Enter DOS Command			any MS-DOS command

For example, if you select Backup Files from the File Utilities submenu, MENU calls up an Epson-created program named SETBCKUP. This program then calls up and runs the MS-DOS BACKUP command, but it does so by displaying a series of menus on the screen from which you select options to control the BACKUP command.

This structure means you can run the BACKUP command in one of three ways:

- Type BACKUP at the MS-DOS command prompt or place the command in a batch file, with all the appropriate parameters and switches. For instructions, see the BACKUP command in Chapter 4.

- Run the MENU program, select the File Utilities option, and then select the Backup Files option. This lets you run the BACKUP program by selecting options from the menus provided by SETBCKUP.
- Type SETBCKUP at the MS-DOS command prompt. This lets you run BACKUP through the SETBCKUP program (selecting options from menus) without going through the MENU program.

Because of this structure, when you run the MENU program you need to be sure it can find the appropriate Epson and MS-DOS commands (programs). If you have a hard disk, it is best to store MENU and all the necessary programs in one directory. If you do not have a hard disk, do one of the following:

- If your computer has a 1.2MB, 1.44MB, or a 720KB diskette drive, store MENU and all the necessary programs on one diskette. Then use this diskette whenever you run MENU.
- If your computer has only one or two 360KB diskette drives, use the following PATH command before you run the MENU program:

  ```
  PATH B:
  ```

  Then, when you run the MENU program and request an option that requires an MS-DOS command, the screen prompts you to insert the diskette for drive B. At this point, insert the diskette containing the program you want to use—in drive A if you have only one drive or drive B if you have two—and press any key to continue. Be sure to insert the correct diskette; otherwise, MENU displays an error message and you will not be able to complete the operation.

  When MENU finishes using the program, the screen prompts you to insert the diskette for drive A. If you have one drive, replace the diskette in drive A with the diskette that contains MENU. Press any key to continue.

With this method, you need to know exactly which programs are on which diskettes. You may want to keep a printout of the disks' directories for reference. (You can print the directory of a disk by pressing **Ctrl P** before giving the DIR command.)

If your work sessions involve a lot of disk handling, or you just want to keep things simple, it's a good idea to use this convenient menu program—alone or within a batch file. You can include MENU in a batch file just like any other MS-DOS command.

The instructions below first provide general rules for selecting options from the menus and then describe each of the MENU options separately. If you want to learn more about a particular menu option, see the description of the corresponding MS-DOS command in Chapter 4.

Selecting Options From Menus

Although each menu offers different options, the procedure for selecting options is the same, and each screen in the MENU program tells you what you need to do.

The key functions for controlling a menu appear in the bottom part of every menu screen. For example, the main menu displays the following prompt:

```
↓ Selects Function Enter Exits Program
```

This tells you to press ↓ to move the cursor block to the option you want to select or press **Enter** to activate the selected option (Exit, in this case).

The following message also appears on the main menu:

```
Use the PgUp and PgDn keys to move this Help Text up
and down one line.
```

Every menu screen provides help information in this bottom part of the screen about the option currently selected by the cursor block. If there is more text than can fit in these five lines, press **PgUp** to move the help information up to display more text; then press **PgDn** if you want to read the previous text or to go back to the top of the help text. You can use **PgUp** and **PgDn** on any of the submenus to read about the selected option.

Check this part of the screen whenever you need information about the command or what key to press.

When a submenu for a utility (such as BACKUP) is on the screen, the name of the command is in the upper left corner, such as this:

```
BACKUP Utility
```

When a utility menu is on the screen, you often need to choose one of two possible parameters for an option by pressing the **Enter** key. This switches the selection back and forth between the first choice and the second—YES and NO, for example. When the desired choice appears, press ↑ or ↓ to move the cursor block to another option.

If a particular option requires you to type in some information—the name of a file, for example—you must type something before you can move on to the next option. This prevents you from leaving out information that the command needs to work properly. Press the **Insert** key to turn on or off TYPEOVER mode when typing in this information. You can use the backspace key and the **Delete** key to correct text you type.

As you select options from a utility menu, the MS-DOS command string appears below the menu to show you how this command would appear if you were typing it at the command prompt. For example, the command string for a BACKUP command might look like this:

```
COMMAND STRING > BACKUP C:\*.* A:/S
```

When you want to leave the displayed menu, just keep pressing ↑ until the cursor block is on the EXIT option; then press **Enter**. You can also press the **Esc** key to return to the previous menu.

Select the option you want from the main menu, then go to the description of that function below for instructions on using it.

File Utilities

When you select the File Utilities option from the main menu, the screen displays these options:

```
          EXIT
     Backup  Files
     Restore Files
     Replace Files
     Compare Files
Change File Attributes
Copy Files/Directories
```

Move the cursor block to the option you want and press **Enter**. Then follow the appropriate instructions below for that option.

Backup files

The Backup Files option lets you back up (copy) files from one disk to another—most typically, from a hard disk to diskettes—to protect your data. You cannot access files that have been copied with the Backup Files option, but you can copy them back to the original disk with the Restore Files option and then access them.

This option runs the MS-DOS BACKUP program.

When you select the Backup Files option, the screen displays this menu:

```
                    EXIT
  Source        =
  Target Drive =    :
   NO       Include Subdirectory Files
   NO       Include Only Modified Files
   NO       Add Files Without Replacing
   NO       Select A Starting Date
   NO       Select A Starting Time
   NO       Format The Target Disk Before Backup
   NO       Specify A Log File =
************ S T A R T   B A C K U P ************
```

You must enter the appropriate information for the first two items in this menu. The rest of the options offer a choice between NO or YES; you can either accept the default (NO) or press **Enter** to change the parameter to YES. Use the BACKUP options as follows:

Source—Type the drive letter, pathname, and filename of the file or files you want to back up. You can use wildcards to select groups of files. For example, to back up all files on drive C in the \LETTERS directory, you would type:

```
C:\LETTERS\*.*
```

Target Drive—Type the letter of the drive containing the target disk (which is usually A).

Include Subdirectory Files—If you want to include all the files in any subdirectories that exist in the specified pathname, select YES.

Include Only Modified Files—If you want to include only those files that have been modified since the last time BACKUP was used on this disk, select YES.

Add Files Without Replacing—If you want to add only new files, and not replace those already existing on the target disk, select YES.

Select A Starting Date—If you want to back up all files that have been created or modified beginning from a certain date, select YES. A small box appears on the right side of the menu with this prompt (for the United States):

```
MM-DD-YYYY
```

Enter the date in this format (two digits each for the month and day and four digits for the year). The format depends on the COUNTRY command in your CONFIG.SYS file. You do not enter the dashes. For example, for June 22, 1989 you would type:

```
Ø6221989
```

Then press **Enter** to move the cursor back to the large menu. If you type an invalid date (more than 12 for the month or 31 for the day), you must reenter the date.

Select A Starting Time—If you want to back up all files that have been created or modified beginning from a certain time, select YES. A small box appears on the right side of the menu with this prompt:

```
HH:MM:SS
```

Enter the time in this format (two digits each for the hour, minutes, and seconds) using the 24-hour clock, but do not type the colons. For example, to select 9:30 a.m., you would type:

```
Ø93ØØØ
```

To select 4:15 p.m., you would type:

```
161ØØ
```

Then press **Enter** to move the cursor back to the large menu. If you type an invalid time (more than 23 for the hour or 59 for the minutes or seconds), the cursor does not go back to the large menu and you must reenter the time.

Format the Target Disk Before Backup—If the disk you are copying the files to has not been formatted, select YES; otherwise, leave the default to NO for this option.

If you select YES, you must specify the size of the target diskette. The options are 360KB, 720KB 1.2MB, and 1.44MB.

Specify A Log File—If you want to create a file on the source disk that lists the names of all the backup files, select YES. This creates a file you can read and print like any text file. The log file is given the default name of BACKUP.LOG unless you enter a different one. To change it, enter a name consisting of up to eight characters for the name (and up to three characters for the extension, if desired). You can store the log file on another disk, as long as it is not the target disk. Just specify the desired drive before the filename.

When you have made your selections, move the cursor block down to START BACKUP and press **Enter** to begin the backup procedure. If you have entered any invalid information in the menu, an error message appears stating what is wrong. Correct the error and select the START option again.

The screen prompts you to insert backup diskette 01 (the first diskette) in the target drive and press any key when ready. The program copies the files from the source disk to the target disk and displays the filenames on the screen as they are copied.

If the first disk fills up, the screen prompts you to insert backup diskette 02. Follow the prompts on the screen until all the files have been backed up. Then the screen prompts you to press any key to continue, and the Backup Files menu reappears. Move the cursor block up to EXIT and press **Enter** to return to the File Utilities menu.

Restore files

Use this option when you want to restore files that have been backed up with the Backup Files option (or the BACKUP command).

This option runs the MS-DOS RESTORE program.

When you select the Restore Files option, the screen displays this menu:

```
                    EXIT
  Source Drive =  :
  Target       =
  NO        Include Subdirectory Files
  NO        Prompt Before Restoring
  NO        Restore Files Before A Date
  NO        Restore Files After A Date
  NO        Include Only Modified and Deleted Files
  NO        Include Only Files Not On The Target
  NO        Restore Files Later Than A Time
  NO        Restore Files Earlier Than A Time
********** S T A R T   R E S T O R E **********
```

You must enter the appropriate information for the first two items in this menu. The rest of the options offer a choice between NO or YES; you can either accept the default (NO) or press **Enter** to change the parameter to YES. Use the RESTORE options as follows:

Source Drive—Type the letter of the drive containing the source disk (the disk containing the files you want to restore to the target disk).

Target—Type the drive letter and pathname on the target disk to which you want to restore the files from the source disk. You can use wildcards to identify a group of files. For example, to restore the files to drive C in the \SALES directory, you would type:

```
C:\SALES\*.*
```

Include Subdirectory Files—If you want to include all the files in any subdirectories that exist in the specified pathname, select YES.

Prompt Before Restoring—If you want the program to ask you whether you want to restore modified or read-only files, select YES. With this option selected, a prompt appears before each file is copied so you can choose whether or not to copy it.

Restore Files Before A Date—If you want to restore only those files that were modified on or before a specified date, select YES. A small box appears on the right side of the menu with this prompt (for the United States):

```
MM-DD-YYYY
```

Enter the date in this format (two digits each for the month and day and four digits for the year). The format depends on the COUNTRY command in your CONFIG.SYS file. You do not enter the dashes. For example, for May 16, 1989 you would type:

```
Ø5161989
```

Then press **Enter** to move the cursor back to the large menu. If you type an invalid date (more than 12 for the month or 31 for the day), you must reenter the date.

Restore Files After A Date—If you want to restore only those files that were modified on or after a specified date, select YES. A small box appears on the right side of the menu with this prompt (for the United States):

```
MM-DD-YYYY
```

Enter the date in this format (two digits each for the month and day and four digits for the year). The format depends on the COUNTRY command in your CONFIG.SYS file. You do not enter the dashes. For example, for June 22, 1989 you would type:

```
Ø6221989
```

Then press **Enter** to move the cursor back to the large menu. If you type an invalid date (more than 12 for the month or 31 for the day), you must reenter the date.

Include Only Modified and Deleted Files—If you want to include only those files on the target disk that have been modified or deleted since the last time BACKUP was used on this disk, select YES.

Include Only Files Not On The Target—If you want to restore only those files that exist on the source disk but not on the target, select YES.

Restore Files Later Than A Time—If you want to restore only those files that have been modified at or after a certain time, select YES. A small box appears on the right side of the menu with this prompt:

```
HH:MM:SS
```

Enter the time in this format (two digits each for the hour, minutes, and seconds) using the 24-hour clock, but do not type the colons. For example, to select 11:00 a.m., you would type:

```
11ØØØØ
```

To select 8:30 p.m., you would type:

```
2Ø3ØØØ
```

Then press **Enter** to move the cursor back to the large menu. If you type an invalid time (more than 23 for the hour or 59 for the minutes or seconds), you must reenter it.

Restore Files Earlier Than A Time—If you want to restore only those files that have been modified at or before a certain time, select YES. A small box appears on the right side of the menu with this prompt:

```
HH:MM:SS
```

Enter the time in this format (two digits each for the hour, minutes, and seconds) using the 24-hour clock, but do not type the colons. For example, to select 7:30 a.m., you would type:

```
Ø73ØØØ
```

To select 6:50 p.m., you would type:

```
185000
```

Then press **Enter** to move the cursor back to the large menu. If you type an invalid time (more than 23 for the hour or 59 for the minutes or seconds), you must reenter it.

When you have made your selections, move the cursor block down to START RESTORE and press **Enter** to begin. If you have entered any invalid information in the menu, an error message appears stating what is wrong. Correct the error and select the START option again.

The screen prompts you to insert backup diskette 01 (the first diskette) in the source drive and press any key when ready. The program copies the files from the source disk to the target disk and displays the filenames on the screen as they are copied. The screen also displays the date the files were backed up.

After all the files on the first disk have been copied, the screen prompts you to insert backup diskette 02, if necessary. Follow the screen prompts until all the files have been restored. Then press any key to return to the Restore Files menu. Move the cursor block up to EXIT and press **Enter** to return to the File Utilities menu.

Replace files

The Replace Files option replaces a file or group of files on the target disk with those from the source disk that have the same name, and displays the filenames on the screen as they are replaced. You can also add files from the source disk that do not exist on the target disk. You cannot replace hidden files or system files.

This option runs the MS-DOS REPLACE program.

When you select the Replace Files option, the screen displays this menu:

```
                    EXIT
   Source =
   Target =
    NO      Add Files Not On The Target
    NO      Prompt Before Replacing or Adding Files
    NO      Overwrite Read-Only Files
    NO      Search All Directories
    NO      Overwrite Older Files
    NO      Pause Before Beginning
********** S T A R T   R E P L A C E **********
```

You must enter the appropriate information for the first two items in this menu. The rest of the options offer a choice between NO or YES; you can either accept the default (NO) or press **Enter** to change the parameter to YES. Use the Replace Files options as follows:

Source—Type the drive letter, pathname, and filename of the file or files on the source disk. These will replace matching files on the target disk. You can use wildcards to identify a group of files. For example, to select the files from drive A in the \LETTERS directory that have the extension .NEW, you would type:

```
A:\LETTERS\*.NEW
```

Target—Type the drive letter and pathname for the files you want to be replaced on the target disk. For example, to replace the files on drive C in the \LETTERS directory you would type:

```
C:\LETTERS
```

Add Files Not On The Target—If you only want to add files from the source disk that do not exist on the target disk, select YES. This option adds new files but does not replace existing ones. You cannot select both this option and the Search All Directories option; choose one or the other.

Prompt Before Replacing or Adding Files—If you want the program to prompt you to respond Yes or No for each file before replacing (or not replacing) it, select YES.

Overwrite Read-Only Files—If you want files on the source diskette to replace matching files on the target disk that are read-only, select YES. (You can set a file as read-only with the Change File Attributes option—see below—or the ATTRIB command.)

Search All Directories—If you want the files on the source diskette to replace matching files in all subdirectories of the target directory, select YES. You cannot select both this option and the Add Files Not On The Target option; choose one or the other.

Overwrite Older Files—If you want REPLACE to overwrite only those files on the target disk that have a newer date on the source disk, select YES. You cannot use this option in combination with the Add Files Not On The Target option.

Pause Before Beginning—If you want the program to pause before it starts so you can insert the source diskette, select YES.

When you have made your selections, move the cursor block down to START REPLACE and press **Enter** to begin replacing files. If you have entered any invalid information in the menu, an error message appears stating what is wrong. Correct the error and select the START option again.

The program replaces the files on the target disk with those from the source disk that have the same name, and displays the filenames on the screen as they are replaced. Then the screen prompts you to press any key to continue, and the Replace Files menu reappears. Move the cursor block up to EXIT and press **Enter** to return to the File Utilities menu.

Compare files

Use the Compare Files option to compare the contents of two files. This option shows you exactly what the differences are, and you can output the results to either the screen or to another file. The files can be either text files or binary files.

This option runs the MS-DOS FC program.

Note
If you just want to know if two files are different and are not concerned with how they are different, you may want to use the COMP (compare) command instead. See COMP in Chapter 4 for instructions.

When you select the Compare Files option, this menu appears:

```
                    EXIT
Filename1 =
Filename2 =
NO        Abbreviate the Output
NO        Force a Binary Comparison
NO        Ignore the Case of Letters
NO        Force a DOS Text File Comparison
NO        Display Line Numbers
NO        Compress Tabs Only
NO        Compress Tabs and Spaces
NO        Reset the Internal Line Buffer
NO        Specify Number of Matching Lines
YES       Output to the Screen :
************ S T A R T     F C ************
```

You must enter the exact filenames for the first two items in this menu. The rest of the options offer a choice between NO or YES; you can either accept the default or press **Enter** to change the parameter. Use the Compare Files options as follows:

Filename1—Type the drive letter, pathname, and filename for the first file.

Filename2—Type the drive letter, pathname, and filename for the second file.

Abbreviate the Output—If you want to see only the first and last lines of each section of text that is different, select YES.

Force a Binary Comparison—Any file with the extension .EXE, COM, .SYS, .OBJ, .LIB, or .BIN automatically receives a binary comparison. If you want to force a binary comparison on another type of file, select YES for this option.

Ignore the Case of Letters—If you want the comparison to ignore whether letters are uppercase or lowercase, select YES.

Force a DOS Text File Comparison—Any standard DOS text file automatically receives an ASCII comparison. If you want to perform an ASCII comparison on a system file (one with the extension .EXE, .COM, .SYS, .OBJ, .LIB, or .BIN), select YES for this option.

Display Line Numbers—This option assigns a line number for each line of text in the two files. If you want to see the line numbers for the files as they are compared, select YES. You cannot use this option if you are comparing binary files.

Compress Tabs Only—Normally, the comparison expands tabs to eight spaces. If you want to compress each tab to only one space, select YES.

Compress Tabs and Spaces—If you want to compress both tabs and contiguous blank spaces to a single space, select YES.

Reset the Internal Line Buffer—If the program finds more than 100 lines in the two files that are different, it terminates the comparison. You can use this option to define a different value for the internal line buffer. First select YES to reset the buffer. A small box appears to the right of the menu. Type the number of lines you want to be the maximum amount of differing lines before the comparison is ended, from 1 through 999. Press **Enter**.

Specify Number of Matching Lines—Normally, the program expects to find at least two lines that are the same after finding a difference to continue the comparison. If you want to specify a different number, press **Enter**. A small box appears to the right of the menu. Type the number (from 1 through 9) you want to be the minimum amount of matching lines for the comparison and press **Enter**.

Output to the Screen—If you want the comparison to be recorded to a file or a printer instead of on the screen, select NO. Then type a name for the file to be created or the device name for the printer, including the drive letter and pathname if necessary.

When you have made your selections, move the cursor block down to START FC and press **Enter** to begin comparing files. If you have entered any invalid information in the menu, an error message appears stating what is wrong. Correct the error and select the START option again.

The program compares the two files and displays information about them on the screen (or stores the information in a file if you chose to output to a file). Then the screen prompts you to press any key to continue, and the Compare Files menu reappears. Move the cursor block up to EXIT and press **Enter** to return to the File Utilities menu.

Change file attributes

Use the Change File Attributes option to change the read-only attribute or archive bit of a file or group of files. When you set a file to read-only, it can be read, but cannot be altered or deleted. The archive bit tells MS-DOS if a file has been altered so commands such as BACKUP and XCOPY know whether to copy it.

This option runs the MS-DOS ATTRIB program.

When you select the Change File Attributes option, the screen displays this menu:

```
                    EXIT
Filename  =
No Change   Read-Only Attribute
No Change   Archive Attribute
  NO        Include All Subdirectory Files
 ******* S T A R T   A T T R I B ******
```

Use the Change File Attributes options as follows:

Filename—Type the drive letter, pathname, and filename of the file or files whose attributes you want to change. You can use wildcards to identify a group of files. For example, to select all files on drive B in the \SALES directory that have the extension .88, you would type:

```
B:\SALES\*.88
```

Read-Only Attribute—If you want to change the read-only attribute of the specified file or files, move the cursor to this item and press **Enter**.

A small box appears to the right of the menu with these options:

```
No Change
  Set
Cancel
```

To turn the read-only attribute on, move the cursor to Set. To turn the attribute off, move the cursor to Cancel. Select No Change if you don't want to change anything. Press **Enter**.

Archive Attribute—If you want to change the archive attribute of the specified file or files, move the cursor to this item and press **Enter**. A small box appears to the right of the menu with these options:

```
No Change
  Set
Cancel
```

To turn the archive attribute on, move the cursor to Set. To turn the attribute off, move the cursor to Cancel. Select No Change if you don't want to change anything. Press **Enter**.

Include All Subdirectory Files—If you want to include all files in any subdirectories of the specified filename, select YES.

When you have made your selections, move the cursor block down to START ATTRIB and press **Enter** to begin. If you have entered any invalid information in the menu, an error message appears stating what is wrong. Correct the error and select the START option again.

The program changes the attributes as specified, or if no change is selected for attributes it displays the current attributes. The screen then prompts you to press any key to continue, and the Change File Attributes menu reappears. Move the cursor block up to EXIT and press **Enter** to return to the File Utilities menu.

Copy files/directories

Use the Copy Files/Directories option to copy files and directories. This is a useful way to copy groups of files within their directories, which you cannot do with the COPY command.

This option runs the MS-DOS XCOPY program.

When you select the Copy Files/Directories option, the screen displays this menu:

```
                    EXIT
  Source =
  Target =
   NO       Copy Only Modified Files
   NO       Duplicate Empty Subdirectories
   NO       Copy Files and Turn Off Archive Attribute
   NO       Prompt Before Copying Each File
   NO       Include Files In All Subdirectories
   NO       Check the Copy Against the Original
   NO       Pause Before Starting
   NO       Copy Files After a Date
**************** S T A R T    X C O P Y **************
```

You must enter the exact file information for the first two items in this menu. The rest of the options offer a choice between NO or YES; you can either accept the default (NO) or press **Enter** to change the parameter to YES. Use the Copy Files/Directories options as follows:

Source—Type the drive letter, pathname, and filename of the file or files on the source disk you want to copy to the target disk. You can use wildcards to identify a group of files. For example, to copy all the files from drive C in the \SALES directory you would type:

```
C:\SALES\*.*
```

Target—Type the drive letter, pathname, and filename where you want to store the copies. For example, to copy the files to the \SALES directory on drive B, you would type:

```
B:\SALES
```

Copy Only Modified Files—If you want to copy only those files that have been modified, select YES. This option copies files whose archive bit has been set, but does not turn off the archive bit of the source file.

Duplicate Empty Subdirectories—If you want to copy empty subdirectories (those that do not contain any files) in addition to those that contain files, select YES. You must also select the Include Files In All Subdirectories option when you select this option.

Copy Files and Turn Off Archive Attribute—Like the Copy Only Modified Files option, this option copies only those files that have been modified, but it also turns off the archive bit on the source files. If you want the archive bit on the source files turned off after the files have been copied, select YES.

Prompt Before Copying Each File—If you want the program to prompt you to respond Yes or No before it copies (or does not copy) each file, select YES.

Include Files In All Subdirectories—If you want the program to copy files in all subdirectories of the specified source directory (in addition to the files in the source directory), select YES.

Check the Copy Against the Original—If you want the program to check the copy against the source file after it copies each file, select YES. If anything is wrong, the program copies the file again.

Pause Before Starting—If you want the program to pause before starting the copy operation so you can swap diskettes if necessary, select YES.

Copy Files After a Date—If you want to copy only those files that were modified on or after a certain date, select YES. A small box appears to the right of the menu with this prompt (for the United States):

```
MM-DD-YYYY
```

Enter the date in this format (two digits each for the month and day and four digits for the year). The format depends on the COUNTRY command in your CONFIG.SYS file. You do not enter the dashes. For example, for May 28, 1989 you would type:

```
Ø5281989
```

Then press **Enter** to move the cursor back to the large menu. If you type an invalid date (more than 12 for the month or 31 for the day), you must reenter the date.

When you have made your selections, move the cursor block down to START XCOPY and press **Enter** to begin. If you have entered any invalid information in the menu, an error message appears stating what is wrong. Correct the error and select the START option again.

The program copies the files as specified. The screen then prompts you to press any key to continue, and the Copy Files/Directories menu reappears. Move the cursor block up to EXIT and press **Enter** to return to the File Utilities menu.

Disk Utilities

When you select the Disk Utilities option from the main menu, the screen displays these options:

```
   EXIT
Check Disk
Disk  Copy
Disk  Compare
Format Disk
```

Move the cursor block to the option you want and press **Enter**. Then follow the appropriate instructions below for that option.

The Disk Utilities option calls the Epson program DU. When you select one of the options from this menu, DU runs the corresponding MS-DOS command.

Check disk

Use the Check Disk option to check the directories, files, and file allocation tables on the specified disk. The program displays a disk and memory status report. This option runs the MS-DOS CHKDSK program.

When you select the Check Disk option, the screen displays this menu:

```
Destination = A:
    NO        Correct Errors
    NO        Display Messages
  ******** START CHECK DISK ********
```

Use the Check Disk options as follows:

Destination Drive—Enter the letter of the drive containing the disk you want to check. The default is A. If you want to check a particular file or group of files to see if they are contiguous, you can specify a filename as well. For example, to check all files in drive A with the extension .TXT, you would type:

```
A:*.TXT
```

Correct Errors—If you want the program to correct any errors it finds in the directory or File Allocation Table and write the corrections on the disk, select YES.

Display messages—If you want the program to display filenames and directories while it is running, select YES.

When you have made your selections, move the cursor block down to START CHECK DISK and press **Enter** to begin. If you have entered any invalid information in the menu, an error message appears stating what is wrong. Correct the error and select the START option again.

The program checks the disk and files as specified. The screen then prompts you to press any key to continue, and the Check Disk menu reappears. Press ↑ to return to the Disk Utilities menu.

Disk copy

Use the Disk Copy option to make an exact copy of a diskette. This option runs the MS-DOS DISKCOPY program. It is a good idea to write protect your source disk before copying it so you do not accidentally destroy any of its data.

When you select Disk Copy, the screen displays this menu:

```
    :       Source Drive
    :       Destination Drive
   NO       Copy a Single Side
******** START DISK COPY ********
```

Use the Disk Copy options as follows:

Source Drive—Enter the letter of the drive containing the source disk (the disk you want to copy).

Destination Drive—Enter the letter of the drive containing the target disk. (If you have only one diskette drive, enter that letter for both the source and destination drives.)

Copy a Single Side—If you are copying to a single-sided diskette, select YES.

When you have made your selections, move the cursor block down to START DISK COPY and press **Enter** to begin. If you have entered any invalid information in the menu, an error message appears stating what is wrong. Correct the error and select the START option again.

Follow the screen prompts carefully to insert and swap diskettes as necessary. Be careful not to mix up the source and target diskettes. When the diskette has been copied, this prompt appears:

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

If you want to copy another diskette, press **Y** and **Enter** and then follow the screen prompts. If not, press **N** and **Enter**. The Disk Copy menu reappears. Press ↑ to return to the Disk Utilities menu.

Disk compare

Use the Disk Compare option to compare the contents of one diskette with another. This option runs the MS-DOS DISKCOMP program.

When you select the Disk Compare option, the screen displays this menu:

```
  :       Source Drive
  :       Destination Drive
DOUBLE    Single/Dbl Sided
 9        8/9 Sectors/Trk
*** START DISK COMPARE ***
```

Use the Disk Compare options as follows:

Source Drive—Enter the letter of the drive containing the first of the two diskettes you want to compare.

Destination Drive—Enter the letter of the drive containing the second diskette. (If you have only one diskette drive, enter that drive letter for both the source and destination drives.)

Single/Dbl Sided—If you are comparing two 5¼-inch single-sided diskettes, select SINGLE.

8/9 Sectors/Track—If you are comparing 5¼-inch diskettes that were originally formatted with eight sectors per track, select 8.

When you have made your selections, move the cursor block down to START DISK COMPARE and press **Enter** to begin. If you have entered any invalid information in the menu, an error message appears stating what is wrong. Correct the error and select the START option again.

Follow the screen prompts to insert and swap diskettes as necessary. When the diskettes have been compared, this prompt appears:

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

If you want to compare more diskettes, press **Y** and **Enter** and then follow the screen prompts. If not, press **N** and **Enter**. The Disk Compare menu reappears. Press ↑ to return to the Disk Utilities menu.

Format disk

Use the Format Disk option to format a diskette or the hard disk so MS-DOS can write to it and read it. The diskette can be a new, blank diskette, or one that contains data you do not want to keep.

This option runs the MS-DOS FORMAT program.

WARNING
The FORMAT program erases all data on the diskette. Be sure you do not want to keep any files on a used diskette before you reformat it.

When you select the Format Disk option, you see this prompt:

```
Destination Drive
```

Enter the letter of the drive containing the diskette you want to format. When you press **Enter** or ↓, the screen displays a menu such as this:

```
   A:     Destination Drive
DOUBLE    Single/Double Sided
   NO     8 Sectors / Track
   NO     Create System Disk
   NO     Create Volume Label
   NO     Create MS-DOS Area
1.2MB     Media in 1.2MB Drive
1.44MB    Media in 1.44MB Drive
***** START DISK FORMAT *****
```

This menu shows all possible options; if your system configuration is different, you may not see all of these.

Use the Format Disk options as follows:

Single/Double Sided—If you are formatting a 5¼-inch, single-sided diskette, select SINGLE.

8 Sectors/Track—If you want to format a 5¼-inch diskette with eight sectors per track, select YES.

Create System Disk—If you want to copy the MS-DOS system files to the diskette so it will be bootable, select YES.

Create Volume Label—If you want the disk to have a volume label, select YES. The program prompts you for the name when it is running.

Create MS-DOS Area—This option reserves space on an 8-sectored 5¼-inch diskette for the MS-DOS system files without creating a system disk. If you select this option (YES), you can later copy any version of MS-DOS to the disk with the SYS command; without the option, you can place only version 4.01 of MS-DOS on the disk.

Media in 1.2MB Drive—If you want to format a 360KB diskette in a 1.2MB drive, select 360KB.

Media in 1.44MB Drive—If you want to format a 720 diskette in a 1.44MB drive, select 720KB.

When you have made your selections, move the cursor block down to START DISK FORMAT and press **Enter** to begin. If you have entered any invalid information in the menu, an error message appears stating what is wrong. Correct the error and select the START option again.

Follow the screen prompts to insert the diskette or enter a volume label, as necessary. When the disk has been formatted, this prompt appears:

```
Format another (Y/N)?
```

If you want to format another diskette, press **Y** and **Enter** and then follow the screen prompts. If not, press **N** and **Enter**. When the Format Disk menu reappears, press ↑ to return to the Disk Utilities menu.

Mode Settings

When you select the Mode Settings option from the main menu, the screen displays these options:

```
     EXIT
Video Display
   Printer
 Serial Port
Redirect LPTn
  Code Page
Device Status
  Keyboard
```

These options let you perform the following procedures:

- Define the operation of your monitor, printer, or serial port.
- Control code page switching.
- Display the status of your devices.
- Set the keyboard typematic rate.

Move the cursor block to the option you want and press **Enter**. Then follow the appropriate instructions below for that option.

The Mode Settings option runs the MS-DOS MODE command. Any changes you make remain in effect until you change them, or turn off or reset the computer.

Video display

Use the Video Display option to change the way your screen displays text. When you select this option, the following menu appears:

```
Display and Shift
Display and Lines
Columns and Lines
```

These options let you use the MODE command to change the way your monitor displays text.

Display and Shift lets you select a display mode (color or black and white, 40 or 80 columns) and shift the display slightly to the left or right. Use this option if you are losing characters on the left or right edge of your screen.

The Display and Lines option lets you choose a display mode (color or black and white, 40 or 80 columns) and select the number of lines to display per screen (25, 43, or 50).

The Columns and Lines option lets you choose the number of characters per line (40 or 80) and the number of lines per screen (25, 43, or 50).

To change display mode but not shift the display left or right, use the first or second option. To change the number of lines displayed but not the width or display mode, use option 2 or option 3.

Display and Shift—When you select the Display and Shift option and press **Enter**, the following menu appears:

```
   BW8Ø    Attributes
   None    Alignment
   No      Test Pattern
*** START VIDEO SETUP ****
```

The following table shows the default and alternative values you can select for these parameters.

Video display parameters

Option	Default	Alternative
Attributes (black & white or color, 40 or 80 columns of screen image width)	BW80	40 80 BW40 CO40 CO80 MONO
Alignment (column shift to offset screen image on monitor)	None	Right Left
Test Pattern (display number pattern to see if screen image is centered)	No	Yes

When you have made your selections, move the cursor block down to START VIDEO SETUP and press **Enter** to begin. If you have entered any invalid information in the menu, an error message appears stating what is wrong. Correct the error and select the START option again.

Follow the screen prompts until the Video Display menu reappears. Move the cursor block back up to return to the Mode Settings menu.

Display and Lines—When you select the Display and Lines option and press **Enter**, the following menu appears:

```
      BW8Ø Attributes
        25 Lines
*** START VIDEO SETUP ***
```

The default Attributes setting is BW80 (black and white, 80 columns per line). You can select from the options shown in the top section of the table above. The default Lines setting is 25. You may change this value to 43 or 50, provided your monitor supports those modes.

When you have made your selections, move the cursor block to START VIDEO SETUP and press **Enter** to begin. If you have entered any incorrect information, an error message appears. Correct the error and select the START option again.

Follow the screen prompts until the Video Display menu reappears. Move the cursor block back up to return to the Mode Settings menu.

Columns and Lines—When you select the Columns and Lines option and press **Enter**, the following menu appears:

```
        40 Columns
         25 Lines
*** START VIDEO SETUP ***
```

The default Columns setting is 40. Press Enter with the cursor block on Columns to change this to 80. The default Lines setting is 25. You may change this to 43 or 50, provided your monitor supports those modes.

When you have made your selections, move the cursor block to START VIDEO SETUP and press **Enter** to begin. If you have entered any incorrect information, an error message appears. Correct the error and select the START option again.

Follow the screen prompts until the Video Display menu reappears. Move the cursor block back up to return to the Mode Settings menu.

Printer

Use the Printer option to select the printer device and to change the way it prints. When you select the Printer option, this menu appears:

```
   LPT1    Printer #
   132     Chars Per Line
   6       Lines Per Inch
   Error   Retry on errors
** START PRINTER SETUP ***
```

The following table shows the default and alternative values you can select for these parameters.

Printer parameters

Option	Default	Alternative
Printer # (printer device number)	LPT1	LPT2 LPT3
Chars Per Line (number of characters to print)	132	80
Lines Per Inch (number of lines to print per vertical inch)	6	8
Retry on error (entry print when errors occur)	ERROR	NONE BUSY READY

Note: You can use only printer device LPT1 unless you install additional option cards for multiple printer ports.

When you have made your selections, move the cursor block down to START PRINTER SETUP and press **Enter** to begin. If you have entered any invalid information in the menu, an error message appears stating what is wrong. Correct the error and select the START option again. Follow the screen prompts until the Printer menu reappears. Then move the cursor block back up to return to the Mode Settings menu.

Serial port

Use the Serial Port option to select the serial port you want to use and to define the communication parameters. When you select the Serial Port option, this menu appears:

```
COM1   Port Number
96ØØ   Baud Rate
Even   Parity
7      Data Length
1      # of Stop Bits
Error  Retry on error
* START SERIAL PORT SETUP *
```

The following table shows the default and alternative values you can select for these parameters.

Serial port parameters

Option	Default	Alternative
Port Number (communications port number)	COM1	COM2 COM3 COM4
Baud Rate (bits transmitted per second)	9600	110 1200 150 2400 300 4800 600 19200
Parity (checking method for bits transmitted)	NONE	ODD EVEN
Data Length (number of bits in each character transmitted)	7	5 6 8
# of Stop Bits (number of bits used to flag transmitted characters)	1	1.5 2
Retry on error (retry when errors occur)	ERROR	NONE BUSY READY

Note: You can use only communication port COM1 unless you install additional option cards for multiple serial ports.

When you have made your selections, move the cursor block down to START SERIAL PORT SETUP and press **Enter** to begin. If you have entered any invalid information in the menu, an error message appears stating what is wrong. Correct the error and select the START option again.

Follow the screen prompts until the Serial Port menu reappears. Then move the cursor block back up to return to the Mode Settings menu.

Redirect LPTn

Use the Redirect LPTn option to redirect data from the printer port to the serial port. When you select this option, the following menu appears:

```
   LPT1   Printer #
   COM1   Port Number
* START PRINT REDIRECTION *
```

The following table shows the default and alternative values you can select for these parameters.

Redirect LPTn parameters

Option	Default	Alternative
Printer # (printer device number to receive output)	LPT1	LPT2 LPT3
Port Number (communications port directing printer output)	COM1	COM2 COM3 COM4

Note: You can use only COM1 unless you install special option cards with additional hardware input/output ports.

When you have made your selections, move the cursor block down to START PRINT REDIRECTION and press **Enter** to begin. If you have entered any invalid information in the menu, an error message appears stating what is wrong. Correct the error and select the START option again.

Follow the screen prompts until the Redirect LPTn menu reappears. Then move the cursor block back up to return to the Mode Settings menu.

Code page

MS-DOS lets you choose from five different code pages (international character sets) so you can produce characters not available in the default character set.

> **Note**
> Be sure to read Chapter 9 for information on code page switching before you use the Code Page option.

Use the Code Page option to do the following:

- Prepare one or more code pages to be selected for use by your monitor or printer.
- Select (activate) which code page to use with which device.
- Display the current status of code page selections for a particular device.
- Refresh (reselect) a code page that was deactivated because of an error, such as a printer malfunction.

When you select the Code Page option, this menu appears:

```
Prepare
Select
Status
Refresh
```

Move the cursor block to the option you want and press **Enter**. Then follow the appropriate instructions below for that option.

Prepare—Before you can prepare a code page, you must have used the DEVICE command in the CONFIG.SYS file to activate the device driver(s) you want to use code page switching. (See Chapter 6 for instructions.)

When you select the Prepare option, this menu appears:

```
  CON        Device
             Code Page List
 EGA.CPI     File Name
** START PREPARE SETUP **
```

The following table shows the default and alternative values you can select for these parameters.

Prepare code page parameters

Option	Default	Alternative
Device	CON	PRN LPT1 LPT2 LPT3
Code Page List	None	437 850 860 863 865
File Name (code page information file)	EGA.CPI	4208.CPI 5202.CPI LCD.CPI

When you have made your selections, move the cursor block down to START PREPARE SETUP and press **Enter** to begin. If you have entered any invalid information in the menu, an error message appears stating what is wrong. Correct the error and select the START option again.

Follow the screen prompts until the Prepare menu reappears. Press ↑ to return to the Code Page menu.

Select—Once you have prepared a code page, you use the Select option to activate the code page for a particular device. When you choose the Select option, this menu appears:

```
  CON      Device
  850      Code Page List
** START SELECT SETUP ***
```

The following table shows the default and alternative values you can select for these parameters.

Select code page parameters

Option	Default	Alternative
Device	CON	PRN LPT1 LPT2 LPT3
Code Page List	NONE	437 850 860 863 865

You can select only one code page per device.

When you have made your selections, move the cursor block down to START SELECT SETUP and press **Enter** to begin. If you have entered any invalid information in the menu, an error message appears stating what is wrong. Correct the error and select the START option again.

Follow the screen prompts until the Select menu reappears. Press ↑ to return to the Code Page menu.

Status—Use the Status option to display the code page status for a particular device. When you choose the Status option, this menu appears:

```
    CON        Device
*** CODE PAGE STATUS ****
```

Press **Enter** to display the device list:

```
CON
PRN
LPT1
LPT2
LPT3
```

Choose the device for which you want to display the code page status. Then move the cursor block down to CODE PAGE STATUS and press **Enter** to begin. If you have selected an invalid device, an error message appears. Correct the error and try again.

The program displays the number of the active code page currently selected and any other prepared code pages for the device. Press any key to return to the Status menu. Press ↑ to return to the Code Page menu.

Refresh—Use the Refresh option to reselect a code page that has been deactivated because of an error. When you choose the Refresh option, this menu appears:

```
    CON      Device
*** REFRESH CODE PAGE ***
```

Press **Enter** to display the device list:

```
CON
PRN
LPT1
LPT2
LPT3
```

Choose the device for which you want to refresh the code page. Then move the cursor block down to REFRESH CODE PAGE and press **Enter** to begin. If you have selected an invalid device, an error message appears. Correct the error and try again.

The program refreshes the code page previously selected for the device. Press any key to return to the Refresh menu. Press ↑ to return to the Code Page menu.

Device status

Use the Device Status option to check the current status of a printer or the screen. When you select this option and press **Enter**, the following menu appears:

```
        CON Device
***** DEVICE STATUS *****
```

To check the current status of your monitor, move the cursor block to DEVICE STATUS and press **Enter**. To check the status of another device, move the cursor block to CON Device and press **Enter**. The following menu appears:

```
CON
PRN
LPT1
LPT2
LPT3
```

Move the cursor block to the name of the device you want to check, and then press **Enter**. Then move the cursor block to DEVICE STATUS and press **Enter** again.

Keyboard

Use the Keyboard option to change the keyboard typematic rate settings. When you select this option and press **Enter**, the following menu appears:

```
          RATE=xx
         DELAY=yy
*** SET KEYBOARD MODE ***
```

To change the rate at which a key repeats when you hold it down, move the cursor block to RATE. Then type a number from 1 to 32. To change the length of time you can hold a key down before it begins repeating, move the cursor block to DELAY. Then type a number between 1 and 4. Then move the cursor block to SET KEYBOARD MODE and press **Enter**.

HELP

You can use the HELP option to display onscreen instructions for any MS-DOS command or Epson utility. HELP provides a quick way to learn about a new command or to quickly check something about a familiar command.

The two programs that provide HELP information, HELP.COM and HELP.TXT, must be in the same directory. Also, if you are not logged onto the directory containing these files, you must have previously used the APPEND command to set a path for these files.

When you select the HELP option, a directory listing of all the commands appears on the screen.

You see this prompt at the bottom of the screen:

```
↑↓→←Select command name,
Enter to display command, Esc to Exit
```

Use the cursor keys to select a command name. (You can select only one command name at a time from the menu.) When the correct command is highlighted, press **Enter**. The help screen for the specified command appears.

If there is more than one page of text, you see the prompt PgUp on the top of the screen. Press **Page Up** to display the rest of the text. After reading the help information, press **Esc** to return to the menu.

Enter DOS Command

Use the Enter DOS Command option to run any MS-DOS command from the MENU program. This is useful if you want to run a command that is not included in the list of MENU options, but you don't want to leave the MENU program.

To use an MS-DOS command through MENU, COMMAND.COM must either be in the current directory or must be part of the path specified in the PATH command.

When you select this option, a command line appears for you to type the MS-DOS command. The screen also shows the drive you are currently logged onto, for example:

```
Current Drive: C
```

Enter the command you want to run, including the complete pathname if necessary, and press **Enter** to execute it. For example, to display a directory of the files on drive A, you would type the following and press **Enter**:

```
DIR A:
```

The screen displays a list of all the files on drive A.

After the command has been completed, the screen displays this prompt:

```
Press any key
```

Press a key to return to the command line. When you finish using this option, press ↑ to return to the main menu.

SETPRINT

The SETPRINT program lets you choose a variety of options for your Epson-compatible printer. By selecting from the menu, you can change such items as typeface and skip over perforation.

To run the program, log onto the directory containing the SETPRINT program or enter the correct pathname, and then type SETPRINT and press **Enter**. The screen displays this main menu:

```
      Exit Program
      Initialize Printer
      Send Form Feed
      Send Carriage Return
off   CONDENSED Mode
off   EXPANDED  Mode
off   EMPHASIZEDMode
off   ITALICS   Type Face
off   ELITE     Type Face
off   Skip-Over-Perf
off   Double-Strike
off   Unidirectional Printing
off   Proportional Spacing
      PRINT this message as a test.
```

Each of the menu options acts as a switch for controlling your printer. Use ↓ or ↑ to move the cursor block to an option. Press **Enter** to activate the function or to select the alternate function.

Be sure to turn on and properly connect your printer before beginning.

When you have made your selection, move the cursor block up to Exit Program and press **Enter**. Your printer now prints according to your choices until you change them or turn off or initialize the printer.

Remember two important points about SETPRINT:

- SETPRINT is designed to control the printing of an entire document or file. If you want to use a particular typestyle for only part of a document, use the print control feature of your application program.
- Some application programs, especially word processors, may cancel your SETPRINT settings.

Options

The menu options make it very easy to select the way your printer operates:

Exit Program—This is your cursor's default position when you first run SETPRINT. Select this option to exit the program and return to the MS-DOS command prompt.

Initialize Printer—Select this option to send a hardware reset that clears any commands you may have set and restores all default settings.

Send Form Feed—Select this option to send a form feed code that advances your paper to the next top of form, which is usually the top of the next page. This command is handy for ejecting single sheets when you finish printing.

Send Carriage Return—Select this option to advance the paper in your printer one line. This is convenient any time you want to move the paper up just a few lines. (This option sends a carriage return and a line feed.)

CONDENSED Mode—Select this option to turn on (or turn off) condensed mode. With condensed mode printing, 132 characters appear on each line, instead of the usual 80.

EXPANDED Mode—Select this option to turn on (or turn off) expanded mode. Expanded mode printing creates double-wide characters useful for emphasis and headings.

EMPHASIZED Mode—Select this option to turn on (or turn off) emphasized mode. Emphasized mode printing uses twice as many dots for each character, slightly offsetting them to fill in and make the image darker.

ITALICS Type Face—Select this option to turn on (or turn off) italic character printing.

ELITE Type Face—Select this option to turn on (or turn off) elite character printing. Elite characters print 12 to the inch instead of 10.

Skip-Over-Perf—Select this option to enable or disable automatic skip over paper perforations while printing. When this option is on, your printer leaves margins at the top and bottom of each page of continuous-form paper. Do not use this feature if your application program provides top and bottom margins.

Double-Strike—Select this option to turn on (or turn off) double-strike printing. Double-strike printing makes two passes on each line, giving a darker image.

Unidirectional Printing—Select this option to enable (or disable) unidirectional printing. Ordinarily, your printer prints in both directions (bi-directionally) to save time. When you want a more precise vertical line, such as you might use to rule a table, unidirectional printing provides greater accuracy.

Proportional Spacing—Select this option to enable (or disable) proportional spacing. The width of proportional characters varies from character to character. Therefore, a narrow letter like i receives less space than a wide letter like W.

PRINT this message as a test—When you have made the selections you want from the above list, you can test the appearance of their output by using this option to print a sample line. Press **Enter** to print the sample line using the options you have selected.

Cautions

While you can't damage your printer or stored data through a misuse of SETPRINT, there are things you need to be aware of so the operation goes smoothly.

Always be sure that your printer is on and on-line before using SETPRINT. Your setup codes do not affect the printer unless it is connected and ready.

The SETPRINT codes are mainly for Epson-compatible dot-matrix printers. The typeface and print mode options apply only to dot-matrix printing; naturally, a daisy wheel cannot compress or expand its characters. However, a daisy wheel printer can print unidirectionally, perform double-strike, and skip over perforations, if it uses Epson-compatible codes.

Some application programs automatically send reset codes to the printer, which could clear any settings you make with SETPRINT. Read the documentation for your application program to see if its codes interfere with SETPRINT's printer configurations and if you can disable its reset code. If not, you can probably choose the same options through your application program.

Once you select the configuration for your printer and exit SETPRINT, the new configuration does not appear on the SETPRINT menu. It isn't possible to read the status of the printer and display it. Anytime you are unsure of how your printer is set up, you can print the test message to verify it, or use the Initialize Printer option and reconfigure.

SETPRINT always directs its codes to the parallel port of your computer. If you are using a serial printer connected to the serial port, your printer does not receive SETPRINT's codes unless you have redirected LPT to the COM port. In this case, the Initialize Printer option does not work. (See the description of the Mode Settings option of the MENU command earlier in this chapter.)

Chapter 6

Configuring Your System

Defining the CONFIG.SYS File 6-3
Allocating Memory 6-5
Opening Files 6-6
Identifying Block Devices 6-7
Configuration Constraints 6-7
BREAK 6-8
BUFFERS 6-9
COUNTRY 6-12
DEVICE 6-15
DRIVPARM 6-17
FCBS 6-19
FILES 6-21
INSTALL 6-22
LASTDRIVE 6-23
REM 6-24
SHELL 6-25
STACKS 6-27
SWITCHES 6-29
Device Drivers 6-30
ANSI.SYS 6-31
DISPLAY.SYS 6-32
DRIVER.SYS 6-34
EEMM286.SYS 6-37
EMM386.SYS 6-40
HIMEM.SYS 6-42
PRINTER.SYS 6-44
SMARTDRV.SYS 6-46
VDISK.SYS 6-48
XMAEM.SYS 6-51
XMA2EMS.SYS 6-52

Defining the CONFIG.SYS File

When you configure your system, you instruct MS-DOS to set up the computer according to your specific requirements. You specify your requirements in a configuration file called CONFIG.SYS. The CONFIG.SYS file contains configuration commands that define system information.

The standard system information defined by MS-DOS is adequate for running most applications. You only need to create a CONFIG.SYS file for certain applications and system requirements. If you need to define special system information, your software documentation will specify the necessary information.

System information you can enter in the CONFIG.SYS file includes the following (the name in parentheses is the configuration command):

- Extended checking of **Ctrl Break** or **Ctrl C** (BREAK)
- The number of buffers you wish to allocate in memory (BUFFERS)
- The country whose date and time format you use (COUNTRY)
- Parameters for a block device, overriding the original MS-DOS device driver settings (DRIVPARM)
- Any device drivers you want to install (DEVICE)
- The number of file control blocks (FCBs) that can be open at any one time (FCBS)
- The number of files that can be open at any one time (FILES)
- The number of drives that MS-DOS can access (LASTDRIVE)
- The name of a command processor that MS-DOS loads instead of COMMAND.COM (SHELL)
- Dynamic use of stack frames (STACKS).

Your CONFIG.SYS file may also include a SWITCHES command that instructs the system to use conventional keyboard functions when an enhanced keyboard is installed, and/or an INSTALL command to automatically invoke any of the following MS-DOS commands:

- FASTOPEN
- KEYB
- NLSFUNC
- SHARE

You can embed comments in your CONFIG.SYS file using the REM command.

Each time you start your system, MS-DOS searches the root directory of the current drive for the CONFIG.SYS file. If CONFIG.SYS does not exist, MS-DOS uses default values for your system configuration.

You can create a CONFIG.SYS file with any text processor, or you can use the MS-DOS editor, EDLIN. (EDLIN is described in Chapter 8, Programming Tools.) You can also use the COPY CON command.

Store the CONFIG.SYS file in the root directory of the system disk you use to start MS-DOS. Whenever you add or change commands in the CONFIG.SYS file, you must restart MS-DOS for the new system configuration to take effect.

Note
The SELECT command automatically creates a CONFIG.SYS file on your hard disk or Startup diskette (see the MS-DOS 4.01 Installation Guide). You can use EDLIN or any other text processor to modify the CONFIG.SYS file created by SELECT.

Allocating Memory

Your computer has two types of memory: random access memory (RAM) and read-only memory (ROM).

RAM is read/write memory, which means you can store data in RAM as well as read information from it. RAM is called volatile memory because when you turn the computer off, the information stored in RAM is erased.

ROM is memory that can be read but not written to or altered. ROM provides permanent storage for program instructions, such as the boot program that starts the computer and the basic input/output system (BIOS). ROM is called nonvolatile memory because the information stored in ROM cannot be erased; it is always available to the system.

The first 1MB of RAM in your computer is called standard, or conventional, memory. MS-DOS can use only the first 640KB of standard memory for application programs. If your computer has an 80286 or 80386 microprocessor, you can add memory beyond 1MB. Such additional memory is called extended memory.

MS-DOS cannot use extended memory for application programs. However, you can use extended memory for virtual (RAM) disks, a print spooler, or a disk cache.

A virtual disk is a block of memory that behaves as though it were a disk drive. You can store and retrieve files using a virtual disk much more quickly than you can using a physical disk drive. Because you are working in memory, the contents of a virtual disk are volatile. Be sure to copy any information from a virtual disk to a physical disk or diskette before you turn off or reset your computer.

MS-DOS 4.01 includes VDISK.SYS, a device driver that creates one or more virtual disks. VDISK.SYS is described later in this chapter.

A print spooler is a block of memory that temporarily stores data sent by a program to a printer. A print spooler lets you continue using your computer while your printer is busy.

A disk cache is a block of memory that holds the contents of frequently accessed files. When a program needs to read a file from disk, it checks to see if the information is stored in the cache. If it is, the program can read the file without accessing the physical disk. Because memory access is much faster than disk access, a disk cache can enhance the performance of your application programs.

MS-DOS 4.01 includes SMARTDRV.SYS, a device driver that performs disk caching. SMARTDRV.SYS is described later in this chapter.

You can install virtual disks, print spoolers, and disk caches in standard memory as well as extended memory. However, the use of standard memory for these purposes reduces the amount of memory available for application programs and data.

Extended memory can be used only on computers with 80286 or 80386 microprocessors. Regardless of what kind of microprocessor your computer uses, however, you can install a different kind of memory, called *expanded* memory. Expanded memory can be used by MS-DOS application programs that support the Lotus/Intel/Microsoft Expanded Memory Specification (LIM EMS). Refer to the documentation for your application programs to see whether they support LIM EMS.

Opening Files

You can specify the number of files that can be open at one time in your CONFIG.SYS.

Some application programs open files with file control blocks (FCBs), and others use file handles. MS-DOS uses the information stored in FCBs and file handles to access the files.

You can use the FCBS or the FILES configuration command if your application program opens a large number of files at one time. These commands are described later in this chapter.

Identifying Block Devices

All block devices are identified by a single alphabetic character (A through Z). Therefore, the number of block devices you can install in your computer is limited to 26. Block devices include physical drives, hard disk partitions, and device drivers. (The term block refers to the way the device transfers data; all disk drives transfer data in blocks.)

MS-DOS assigns a letter to each block device, starting with A. The first physical internal diskette drive is drive A; the second internal diskette drive is B. MS-DOS always uses both A and B for diskette drives, even if a computer has only one diskette drive.

MS-DOS assigns the letter C to the primary partition of the first hard disk. If you have a second hard disk, its primary partition is drive D. Extended partitions are named in order following all primary partitions.

MS-DOS names any other block devices following all hard disk partitions. For example, if you have one hard disk with three partitions, and one virtual (RAM) disk, the hard disk partitions are C, D, and E, and the virtual disk is F.

Configuration Constraints

Specifying configuration parameters usually requires additional memory. This decreases the amount of memory available for your application programs.

BREAK

Internal

Purpose

Tells MS-DOS when to check for **Ctrl Break** or **Ctrl C**.

Format

```
BREAK = ON
```

or

```
BREAK = OFF
```

Description

Use this configuration command to turn on or off the MS-DOS cancel current command function.

When BREAK is OFF, MS-DOS checks for **Ctrl Break** or **Ctrl C** only when your program requests standard input, output, print, and auxiliary operations. The default is OFF.

If BREAK is ON, MS-DOS checks for **Ctrl Break** or **Ctrl C** every time your program accesses the system. ON improves the ability to cancel programs but slows the system. Only set BREAK to ON when you need the extra **Ctrl Break** or **Ctrl C** checking.

BUFFERS *Internal*

Purpose

Sets the number of disk buffers allocated in memory.

Format

```
BUFFERS = n[,m][/X]
```

Description

MS-DOS uses disk buffers to store data when reading from and writing to disk. How many buffers you allocate depends on the type of applications you use and the amount of memory in your computer.

For applications that randomly read and write records (such as data base applications), you may be able to speed up access time by increasing the number of buffers. For applications that sequentially read and write records, allocating more buffers does not increase performance.

The BUFFERS command has two parameters, *n* and *m*, and one switch, /X. Their meanings are as follows:

n	Specifies the number of disk buffers to be allocated
m	Specifies the size, in disk sectors, of a lookahead buffer
/X	Allows buffers to be stored in expanded memory.

Note
The /X switch works only with an IBM expanded memory card and an IBM-specific expanded memory manager (XMAEM.SYS or XMA2EMS.SYS).

The value of *n* may be from 1 to 99, unless you use /X. If /X is used, *n* may be from 1 to 10,000.

The default values for *n* are shown in the following table:

RAM size	*Default n*
512KB or more	15
More than 256KB but less than 512KB	10
More than 128KB but less than 256KB	5
Less than 128KB using a diskette larger than 360KB	3
Any other configuration	2

The lookahead buffer, specified by *m*, allows MS-DOS to read one or more following sectors in addition to the sector called for by an application program. This can speed up the performance of applications that read sequential disk records. The value of *m* may be from 1 to 8.

If you omit *m*, MS-DOS does not use a lookahead buffer.

To determine how many buffers you need, run some of your application programs with the BUFFERS command set to different values. For applications such as word processing, set the number of buffers between 10 and 20 for best performance. If you plan to create a lot of subdirectories, set the number to a value between 20 and 30.

Note
Disk caching programs and MS-DOS buffers serve similar purposes. If you use SMARTDRV.SYS, described later in this chapter, you might not improve performance by increasing the BUFFERS value. In fact, you might want to reduce the number of buffers assigned by MS-DOS as a default. The best way to determine the optimum number of buffers for your system is by experimentation.

Examples

To allocate 20 disk buffers in standard memory without a lookahead buffer, enter this command in your CONFIG.SYS file:

```
BUFFERS=20
```

To allocate 99 disk buffers with the maximum lookahead buffer of 8 sectors, enter this command in your CONFIG.SYS file:

```
BUFFERS=99,8
```

To use MS-DOS's default BUFFERS value, do not enter a BUFFERS command in your CONFIG.SYS file.

COUNTRY *Internal*

Purpose

Selects display parameters that are country dependent, such as the date and time format, currency symbols, and decimal separator, and activates a code page.

Format

`COUNTRY = code[,[page][,[d:][path]filename]]`

Description

Use this command to configure your system for a specific country. In the COUNTRY command, *code* specifies the 3-digit country code for a particular country or language. The *page* parameter specifies the code page that you want to use for the selected country (most countries have two code pages). The optional file specification identifies the file that contains country-specific information.

The valid country codes and code pages are:

Country or Language	*Country Code*	*Code Pages*
Arabic countries	785	437, 850
Australia	061	437, 850
Belgium	032	437, 850
Canada (English)	001	437, 850
Canada (French)	002	863, 850
Chinese, simplified	086	437
Chinese, traditional	088	437
Denmark	045	865, 850
Finland	358	437, 850
France	033	437, 850
Germany	049	437, 850
Hebrew	972	437, 850
Italy	039	437, 850
Japan	081	437
Korea	082	437
Latin America	003	437, 850
Netherlands	031	437, 850

Country or Language	*Country Code*	*Code Pages*
Norway	047	865, 850
Portugal	351	860, 850
Spain	034	437, 850
Sweden	046	437, 850
Switzerland	041	437, 850
United Kingdom	044	437, 850
United States	001	437, 850

If you do not include the COUNTRY command in the CONFIG.SYS file, MS-DOS uses these default values (and assumes that the COUNTRY.SYS file is in the root directory of the current drive):

Country code = 001
Code page = 437
Country information file = \COUNTRY.SYS

The format for this COUNTRY command would be:

```
COUNTRY = ØØ1,437,\COUNTRY.SYS
```

Note
When you install MS-DOS 4.01 using the SELECT program (see the MS-DOS 4.01 Installation Guide), MS-DOS automatically enters a COUNTRY command in your CONFIG.SYS or CONFIG.400 file.

Examples

To set up country-dependent information for Latin America, include the following command in your CONFIG.SYS file:

```
COUNTRY = ØØ3,437,C:\MSDOS\COUNTRY.SYS
```

MS-DOS searches for the COUNTRY.SYS file in the \MSDOS directory of drive C.

To set up country-dependent information for the Hebrew language using the default code page and a specific (user-supplied) country information file called HEBREW.SYS, include the following command in your CONFIG.SYS file:

```
COUNTRY = 972,,C:\HEBREW.SYS
```

The default code page is 437. MS-DOS searches for the country information file in the root directory of the system disk you use to boot MS-DOS.

To set up country-dependent information for French-speaking Canada, include the following command in your CONFIG.SYS file:

```
COUNTRY = ØØ2
```

MS-DOS uses the default code page, 863, and searches the root directory of your system disk for the COUNTRY.SYS file.

DEVICE

Internal

Purpose

Installs the specified device driver during system startup.

Format

`DEVICE = [d:][path]filename [parameters]`

Description

A device driver is a set of instructions that controls peripheral devices or other software programs. Device drivers enable MS-DOS to communicate properly with your peripheral devices and software programs.

MS-DOS loads the standard device drivers that support the standard input, output, printer, hard disk drive, and diskette drive(s). You can use the DEVICE command to add other drivers, called installable device drivers.

MS-DOS includes several useful installable device drivers, which are described later in this chapter. The following are included:

ANSI.SYS	Provides screen and keyboard control
DISPLAY.SYS	Supports code page switching for the console device (CON)
DRIVER.SYS	Assigns new logical names to diskette drives
EEMM286.SYS	Converts extended memory on 80286-based computers to LIM 4.0 expanded memory
EMM386.SYS	Converts extended memory on 80386-based computers to LIM 4.0 expanded memory
HIMEM.SYS	Allows computers with extended memory to use an extra 64KB for programs and data
PRINTER.SYS	Supports code page switching for PRN, LPT1, LPT2, and LPT3
SMARTDRV.SYS	Performs disk caching in extended or expanded memory

VDISK.SYS	Creates virtual (RAM) disks
XMAEM.SYS	Converts extended memory on IBM 80386-based computers to LIM 4.0 expanded memory
XMA2EMS.SYS	Manages IBM-supplied LIM 4.0 expanded memory on IBM computers.

When you buy a new device, such as a mouse or a scanner, you may receive device driver software. Install the device driver using the DEVICE command in your CONFIG.SYS file. To install a new driver, make sure the device driver file is in the directory you specify in the DEVICE command.

Note
COUNTRY.SYS and KEYBOARD.SYS are device drivers that MS-DOS loads automatically. Do not attempt to load either of these drivers with the DEVICE command.

Example

To use the ANSI escape sequences described in Appendix D, include the following command in your CONFIG.SYS file:

```
DEVICE = ANSI.SYS
```

MS-DOS searches the root directory of the default drive for the ANSI.SYS file. If it finds the file, MS-DOS replaces all keyboard input and screen output with the ANSI escape sequences.

DRIVPARM

Internal

Purpose

Overrides the original MS-DOS device driver settings for block devices.

Format

```
DRIVPARM = /D:number [/C][/F:factor][/H:heads][/I]
           [/S:sectors][/T:tracks]
```

Description

You can use DRIVPARM to override a previous block device driver definition. You can use the following switches:

Switch	*Function*
/D:*number*	Specifies the physical drive number. Drive A is 0, drive B is 1, and so on. *number* must be between 0 and 255.
/C	Specifies a device with change-line support (the ability to recognize when the drive door has been opened).
/F:*factor*	Specifies the type of device (its form factor). Must be one of the following (the default is 2): 0 160/180KB or 320/360KB 1 1.2MB 2 720KB (3½-inch) 5 Hard disk 6 Tape drive 7 1.44MB (3½-inch)
/H:*heads*	The maximum number of heads, a value from 1 to 99. The default is 2.
/I	Specifies an electrically compatible 3½-inch drive (one that is installed internally and uses the computer's diskette controller).
/N	Specifies a non-removable device.

Switch	*Function*
/S:*sectors*	The number of sectors per track, a value from 1 to 99. The default is 9.
/T:*tracks*	The number of tracks per side, a value from 1 to 999.

FCBS

Internal

Purpose

Specifies the number of file control blocks (FCBs) that can be open at one time.

Format

FCBS = *m*,*n*

Description

Some application programs use FCBs to create, open, read, write, and delete files. One FCB is set up for each file that is opened.

Use the FCBS command only if you are using a network system and the SHARE command. Without file sharing, MS-DOS can set up any number of FCBs.

m is a number from 1 to 255 and specifies the total number of files that can be used concurrently. The default value is 4.

n is a number from 0 to 255 and specifies the number of opened files that cannot be closed by MS-DOS if a particular application attempts to open more than *m* files. Only the first *n* files are protected from being closed. The default value is 0.

Whenever a request is made to open more than *m* files, MS-DOS checks for the oldest FCB. MS-DOS closes this file automatically unless it is protected by the value set for *n*.

If the number of computers in the network system is small, the default values of 4,0 are usually sufficient. The only restriction on the value of *m* and *n* is that *m* must be greater than or equal to *n*. However, if you set *m* = *n*, MS-DOS cannot close a file if a program tries to open more than *m* files.

When you use the FCBS command in your CONFIG.SYS file, the resident size of MS-DOS is increased. This means the amount of space available to your application programs is decreased.

Example

To set the total number of FCB files that can be open at one time to 4 and to protect the first two files from being closed, include this command in your CONFIG.SYS file:

```
FCBS = 4,2
```

In this case, if you have four files open and you open another file, MS-DOS closes the third file that was opened; the first two files that were opened are protected from this type of closure. If you try to access the third file that is now closed, you see this message:

```
FCB unavailable
```

FILES

Internal

Purpose

Sets the maximum number of files that can be open at one time.

Format

FILES = *nnn*

Description

This command affects only files opened and accessed with file handles. It does not affect files opened with FCBs.

The number of files you set depends on your installation. This is the maximum number of open files in the entire system.

The *nnn* setting can be a number from 8 to 255. The default is 8.

MS-DOS automatically opens five standard files: input, output, error, printer, and auxiliary device. These five files are included in the number you set; therefore, if you set FILES = 1Ø, MS-DOS can open the five standard files and five additional files.

If you run an application program that returns an error message indicating an insufficient number of files, include the FILES command in your CONFIG.SYS file to increase the number of files that can be open.

Example

To allow 20 files to be open at one time, include the following command in your CONFIG.SYS file:

```
FILES = 2Ø
```

INSTALL

Internal

Purpose

Loads any of four particular MS-DOS commands from your CONFIG.SYS file.

Format

`INSTALL=[d:][path]filename.ext [parameters]`

Description

By including one or more INSTALL statements in your CONFIG.SYS file, you can instruct MS-DOS to execute any of the following commands each time you start your system:

- FASTOPEN.EXE
- KEYB.COM
- NLSFUNC.EXE
- SHARE.EXE

For information about these commands, see Chapter 4.

You must include the filename's extension in the INSTALL statement. Make sure you use a separate INSTALL statement for each command you want to install.

Example

To start FASTOPEN automatically so that it stores 50 directory entries and 100 FAT (file allocation table) entries on drive C, include this command in your CONFIG.SYS file:

```
INSTALL=FASTOPEN.EXE C:=(50,100)
```

LASTDRIVE *Internal*

Purpose

Sets the last valid drive letter that MS-DOS recognizes.

Format

```
LASTDRIVE = x
```

Description

This command sets the maximum number of drives that MS-DOS can access.

The *x* parameter is any letter of the alphabet. The minimum value for *x* is the letter equivalent of the number of physical drives installed in your computer. The maximum value is Z, which is equal to 26 drives. The default is E.

This command is useful in a network environment and if you routinely use the SUBST command.

Each drive letter you allow using LASTDRIVE reduces the memory available for application programs and data.

Examples

If your system has two diskette drives, one hard disk drive with two MS-DOS partitions, and four virtual disks, enter this command in your CONFIG.SYS file:

```
LASTDRIVE=H
```

If you routinely use the SUBST command and you want to be able to use any drive letter greater than your highest unsubstituted drive, include this command in your CONFIG.SYS file:

```
LASTDRIVE=Z
```

REM

Internal

Purpose

Allows you to enter comments in your CONFIG.SYS file.

Format

REM [*comment*]

Description

Use this command to add explanatory text to your CONFIG.SYS file. MS-DOS ignores any line that begins with REM.

You can also use REM in batch files.

Example

```
REM Set country code to France
COUNTRY=033
```

The line beginning with REM explains the COUNTRY=command in the line that follows. MS-DOS ignores this line when processing the CONFIG.SYS file.

SHELL

Internal

Purpose

Specifies the name and location of a command processor.

Format

SHELL=[*d*:][*path*]*filename* [*/switches*]

Description

Use this command to specify a top-level command processor, called a shell, to load in place of COMMAND.COM. You can also use this command if you store COMMAND.COM in a directory other than the root directory of the system disk.

Whenever you boot the system with a CONFIG.SYS file that includes the SHELL command, MS-DOS starts using the specified command processor. This command processor interprets the commands you enter.

Be sure to specify the complete path and filename of the command processor if it is not in the root directory of the disk from which you are starting the system. If you do not include the SHELL command, the default is COMMAND.COM.

COMSPEC is an environment variable that also specifies the command processor. MS-DOS uses the COMSPEC variable whenever it reloads any transient part of the command processor. Therefore, MS-DOS sets this variable to the file specification you enter in the SHELL command.

If you create a command processor that accepts switches, you can include these switches (such as /C or /P) on the SHELL command line. If you are identifying the location of COMMAND.COM, you can also include the /C, /E, and /P switches.

Remember that the MS-DOS internal commands and batch processor reside in COMMAND.COM. If you create a command processor, these functions are available only if you duplicate them in the new command processor.

Example

If you store COMMAND.COM in a subdirectory called MSDOS on drive C and you want to allocate up to 3000 bytes for the environment table, place this command in your CONFIG.SYS file:

```
SHELL=C:\MSDOS\COMMAND.COM /P /E:3000
```

STACKS

Internal

Purpose

Sets the number of available stack frames and their size.

Format

STACKS=*m*,*n*

Description

MS-DOS uses stack frames to process hardware interrupts. When a hardware interrupt occurs, MS-DOS allocates one free stack frame to process the interrupt. When the interrupt is complete, MS-DOS releases the stack frame.

The number of stack frames, *m*, can be from 0 to 64. The size of the stack frames, *n*, can be from 0 to 512 bytes.

If you set STACKS = Ø, Ø, MS-DOS does not dynamically allocate stack frames.

If you do not include STACKS in your CONFIG.SYS file, the default for 8086- and 8088-based computers is:

```
STACKS=Ø,Ø
```

For all other computers, the default is:

```
STACKS=9,128
```

If you have a large number of devices installed in your computer, you may experience errors due to hardware interrupts. You would see this message:

```
Internal stack overflow
System Halted
```

If you see this message, increase the number of stack frames, which allows the system to process more hardware interrupts. Normally, you'll want to increase the number of stack frames rather than increase the size of the individual frames.

Allocating additional stack space decreases the amount of memory available to your application programs. Therefore, only change stack allocations if necessary.

Example

To allocate eight stacks of 512 bytes each for hardware interrupt handling, include this command in your CONFIG.SYS file:

```
STACKS=8,512
```

SWITCHES

Internal

Purpose

Provides compatibility for applications that don't recognize the enhanced keyboard.

Format

```
SWITCHES = [/K]
```

Description

Some application programs may not recognize the extended keyboard functions offered by the enhanced keyboard. If you experience problems with such a program, enter this command in your CONFIG.SYS file:

```
SWITCHES=/K
```

This instructs the system to use conventional keyboard functions only.

Device Drivers

A device driver is a set of instructions that acts as a liaison between MS-DOS and any other software program or peripheral device you are using. Device drivers enable MS-DOS to communicate properly with your monitor, keyboard, disk drive(s), and other devices.

You install device drivers in your CONFIG.SYS file to customize and optimize your system. The following installable device drivers are included in this version of MS-DOS:

ANSI.SYS
DISPLAY.SYS
DRIVER.SYS
EEMM286.SYS
EMM386.SYS
HIMEM.SYS
PRINTER.SYS
SMARTDRV.SYS
VDISK.SYS
XMAEM.SYS
XMA2EMS.SYS

ANSI.SYS *External*

Purpose

Provides screen and keyboard control.

Format

```
DEVICE = [d:][path]ANSI.SYS [/K][/L][/X]
```

Description

You can use the ANSI.SYS device driver to control cursor movement, erase all or part of the screen, change the screen display, and reassign key functions. These changes are defined by character strings called escape sequences.

The ANSI.SYS device driver contains the escape sequences required by some application programs to control keyboard and screen operation. The ANSI escape sequences are described in Appendix D.

The following switches are available:

Switch	*Function*
/K	Prevents the use of extended keyboard functions on enhanced keyboards. This switch is provided for compatibility with older applications that do not recognize enhanced keyboards.
/L	Retains the number of screen lines set by the MODE command, even after exiting applications that reset the number of screen lines.
/X	Allows you to reassign of the extended key values on enhanced keyboards.

DISPLAY.SYS *External*

Purpose

Supports code page switching for the console device (CON).

Format

```
DEVICE = [d:][path]DISPLAY.SYS
         CON[:]=(type,[,[hwcp][,n]])
```

or

```
DEVICE = [d:][path]DISPLAY.SYS
         CON[:]=(type,[,[hwcp][,(n,m)]])
```

Description

You need to use this device driver only if you want to do code page switching on your console device.

If you have the proper supporting hardware, you can install this device driver to switch between prepared code pages for your console. See Chapter 9, Code Page Switching, for more information about using this device driver to support code page switching.

You specify the following parameters:

Parameter	*Description*
type	Display adapter. Valid values: MONO (monochrome display adapter) CGA (color graphics adapter) EGA (enhanced graphics adapter) LCD (liquid crystal display)
hwcp	Hardware code page built into the display adapter. Valid values: 437 United States 850 Multilingual 860 Portugal 863 French Canada 865 Norway

Parameter	*Description*
n	Number (from 0 to 12) of additional code pages you want to use. MONO and CGA do not support code page switching; therefore, *n* must be zero. For EGA and LCD, *n* defaults to 1.
m	Number (from 0 to 2) of subfonts supported for each code page. MONO and CGA do not support other fonts; therefore *m* must be zero. For EGA, the default is 2. For LCD, the default is 1.

Example

To set up code page switching for an EGA display adapter with a built-in hardware code page of 437 and the ability to add two more code pages, enter this command in your CONFIG.SYS file:

```
DEVICE=C:\DOS\DISPLAY.SYS CON:=(EGA,437,2)
```

DRIVER.SYS *External*

Purpose

Assigns new logical names to diskette drives.

Format

```
DEVICE = [d:][path]DRIVER.SYS /D:ddd [/T:ttt]
         [/S:ss] [/H:hh] [/C] [/F:f]
```

Description

Use this device driver to assign new logical names to your diskette drives or to add external diskette drives to your computer.

DRIVER.SYS lets you access and use a disk device by referencing a logical drive letter.

With DRIVER.SYS in your CONFIG.SYS file, you can copy files from one drive to the *same* drive when the system has more than one diskette drive. Without DRIVER.SYS, MS-DOS automatically copies files to the second drive.

Switch	*Function*
/D:*ddd*	Specifies the physical drive number. A physical drive has a value of 0 through 255. Numbers 0 to 127 identify diskette drives and 128 to 255 identify hard disk drives. The first physical diskette drive is 0; it is referenced as drive A from the command prompt. The second physical diskette drive is 1. To specify a third physical diskette drive, which must be external, use 2. The first physical hard disk is 128, the second hard disk is 129, and so forth.
/T:*ttt*	Specifies the number of tracks per side (1–999). The default is 80 tracks per side.
/S:*ss*	Specifies the number of sectors per track (1–99). The default is 9 sectors per track.
/H:*hh*	Specifies the maximum number of heads (1–99). The default is 2 heads.

Switch	*Function*
/C	Specifies that changeline (door lock) support is required (only for computer models that support diskette changeline).
/F:*f*	Specifies the device type. Valid values are: 0 = 160/180KB or 320/360KB diskette drive 1 = 1.2MB diskette drive 2 = 720KB diskette drive 7 = 1.44MB diskette drive The default is 2.

When you specify additional physical drives with DRIVER.SYS, include LASTDRIVE = *x* in your CONFIG.SYS file (*x* must be a letter from E to Z). See the LASTDRIVE command in this chapter.

Note
DRIVER.SYS is not intended to be used with hard disks. If you need to assign a logical letter to a hard disk, use the SUBST command (see Chapter 4).

Examples

To copy from an internal 1.2MB diskette drive to the same internal drive, use DEVICE = DRIVER.SYS with these switches:

```
DEVICE = DRIVER.SYS /D:Ø /T:8Ø /S:15 /H:2 /C
         /F:1
```

This assigns the logical drive letter D to the first diskette drive. It can now be referenced as drive A *and* drive D. In this example, your computer has two internal diskette drives (the first one is 1.2MB) and one hard disk.

You can now copy a file (called ONEFILE) from a diskette in drive A to another diskette in drive A (referenced as drive D), using the COPY command in the following manner:

```
COPY A:ONEFILE D:
```

To set up a logical drive letter of D for a 720KB external diskette drive on a computer with one internal diskette drive and a hard disk, use DEVICE = DRIVER.SYS with this format:

```
DEVICE = DRIVER.SYS /D:2
```

To be able to copy from a 720KB external diskette to the same drive, use DEVICE = DRIVER.SYS with the above format, but include it twice in the CONFIG.SYS file (the diskette drive can now be referenced as drive D and drive E):

```
DEVICE = DRIVER.SYS /D:2
DEVICE = DRIVER.SYS /D:2
```

EEMM286.SYS

External

Purpose

Converts extended memory on 80826-based or 80386-based computers to LIM 4.0 expanded memory, or uses the Epson Equity IIe's expanded memory.

Format

```
DEVICE = [d:][path]EEMM286.SYS [size][/K][/Pn][/F]
         [/T][/W]
```

Description

EEMM286.SYS is an Epson-supplied device driver that uses extended memory on Epson 80286- or 80386-based computers to emulate LIM 4.0 expanded memory, or uses expanded memory on the Epson Equity IIe. To use EEMM286.SYS, you must have one of the following:

- An Epson 80286- or 80386-based computer with 640KB of standard memory and at least 128KB of extended memory.
- An Epson Equity IIe with SIMM (single inline memory module) expanded memory.

Note
If you have an 80286-based computer with an expanded memory option board, use the device driver that came with the memory board. Do not use EEMM286.SYS.

The *size* parameter specifies the amount of extended memory to be used as expanded memory. You must include this parameter if you are using an Epson computer that is PC AT compatible. Do not include *size* if you are using an Equity IIe.

Note
If you have an Equity IIe, you use the SETUP program to specify the amount of extended memory allocated to EEMM286.SYS. See your User's Guide for details.

You may use the following switches with EEMM286.SYS:

Switch	*Function*
/K	Specifies that the *size* parameter is expressed in kilobytes. If you do not include /K, *size* is expressed in pages (a page = 16KB).
/P*n*	Specifies the I/O ports used by the Equity IIe. (EEMM286.SYS ignores this switch if the computer is not an Equity IIe or a *size* parameter is included.) Use this switch only if you experience a conflict with an option card installed in your Equity IIe. The meaning of *n* is explained below.
/F	Provides compatibility with LIM Expanded Memory Specification 3.2. By including /F, you instruct EEMM286.SYS to regard zero page allocation as an error. Use this switch only with application programs that support LIM EMS 3.2 but not LIM EMS 4.0.
/T	Instructs EEMM286.SYS to test all allocated memory each time you start or reset your computer. On the Equity IIe, this test occurs automatically; you do not need to use /T.
/W	Provides compatibility with Microsoft Windows and Microsoft Excel. To use expanded memory with Windows or Excel, you must include this switch. You must also use the STARTEMM command before running Windows or Excel (see below).

Note
If you install devices that use expanded memory, be sure the DEVICE = EEMM286.SYS command appears in your CONFIG.SYS file before the commands to install those devices.

On the Equity IIe, EEMM286.SYS uses the following I/O ports (in hexadecimal notation): 2*n*8, 12*n*8, 42*n*8, 52*n*8, 82*n*8, 92*n*8, C2*n*8, and D2*n*8. The default value of *n* is 0. By including the /P*n* switch, you can change *n* to any other value from 1 to F.

If you want to use expanded memory with Microsoft Windows or Microsoft Excel, you must include the /W switch and you must use the STARTEMM command (on your Equity IIe Reference diskette) before running Windows or Excel. One way to do this is to use a batch file to run Windows or Excel and include the STARTEMM command at the beginning of the batch file. Be sure that the STARTEMM.EXE file is in the current directory or a directory specified in a PATH or APPEND command.

Note
Using EEMM286.SYS without /W may lock up your system when you run Microsoft Windows or Microsoft Excel.

If you use EEMM286.SYS, do not use IBM's VDISK.SYS driver with the /E option.

Examples

To convert 1024KB of extended memory on an Epson 80286-based computer to expanded memory, and to test that memory at startup, include this command in your CONFIG.SYS file:

```
DEVICE=EEMM286.SYS 1Ø24 /K /T
```

To specify the I/O ports 2A8, 12A8, 42A8, 52A8, 82A8, 92A8, C2A8, and D2A8 on an Equity IIe, include this command in your CONFIG.SYS file:

```
DEVICE=EEMM286.SYS /PA
```

EMM386.SYS *External*

Purpose

Converts extended memory on 80386-based computers to LIM 4.0 expanded memory.

Format

```
DEVICE = [d:][path]EMM386.SYS [size] [X:mmmm-nnnn]
         [Mx]
```

Description

EMM386.SYS uses extended memory on an 80386-based computer to emulate Lotus/Intel/Microsoft (LIM) 4.0 expanded memory. To use EMM386.SYS, you must have an 80386-based computer that uses the IBM PC AT standard hardware/software interface, and you must have extended memory.

You can use these parameters with EMM386.SYS:

Parameter	*Function*
size	Specifies the amount of extended memory to use, in kilobytes. The default is 256.
X:*mmmm-nnnn*	Specifies a memory exclusion range, in hexadecimal notation. EMM386.SYS will not locate its page frame or other mappable pages in this memory range. Include this parameter only if you experience conflict with an expansion option board. You may include more than one X: parameter in your DEVICE=EMM386.SYS command.

Parameter	*Function*
M*x*	Specifies an address for the EMM386.SYS page frame. Do not use this parameter unless you want to force EMM386.SYS to use a particular address. The address specified by *x* is as follows:

x	Page frame begins at segment
0	C000
1	C400
2	C800
3	CC00
4	D000
5	D400
6	D800
7	DC00
8	E000

Note
If you install devices that use expanded memory, be sure the DEVICE=EMM386.SYS command appears in your CONFIG.SYS file before the commands to install those devices.

Examples

To convert 1MB (1024KB) of extended memory to LIM 4.0 expanded memory, include this command in your CONFIG.SYS file:

```
DEVICE=EMM386.SYS 1Ø24
```

To convert 512KB of extended memory to expanded memory and ensure that EMM386.SYS does not locate its page frame or other mappable pages in the ranges C400 to C7FF and E000 to E3FF, include this command in your CONFIG.SYS file:

```
DEVICE=EMM386.SYS 512 X:C4ØØ-C7FF X:EØØØ-E3FF
```

HIMEM.SYS *External*

Purpose

Supports the Extended Memory Specification (XMS) version 2.00, allowing 80286- or 80386-based computers with extended memory to use an extra 64KB for programs and data and providing MS-DOS programs with a standard method of storing data in extended memory.

Format

```
DEVICE = [d:][path]HIMEM.SYS [/HMAMIN=h]
         [/NUMHANDLES=n]
```

Description

HIMEM.SYS implements the Extended Memory Specification (XMS) 2.0, which serves two purposes: It adds almost 64KB of memory to the 640KB that MS-DOS programs can use, and it provides a machine-independent standard for programs to follow in storing data in extended memory. The extra block of memory, called the *high memory area* (HMA), extends from address FFFF:10 to FFFF:FFFF and is 64KB minus 16 bytes in length.

To use HIMEM.SYS, you must have a computer with an 80286 or 80386 microprocessor, and you must have at least 1MB of memory.

You can use these switches with HIMEM.SYS:

Switch	*Function*
/HMAMIN=*h*	Specifies the minimum amount number of kilobytes to be allocated to a program from the high memory area. The minimum is 0, the maximum is 63, and the default is 0.
/NUMHANDLES=*n*	Specifies the maximum number of extended memory block (EMB) handles that may be open at any time. The minimum is 1, the maximum is 128, and the default is 0. Each handle uses 6 bytes of resident memory.

The /HMAMIN switch lets you specify the smallest program that can be stored in the high memory area. This is important if you use several TSR (terminate and stay resident) programs. By specifying a large enough value for *h*, you can avoid fragmentation of the high memory area by small TSRs. If you do not include the /HMAMIN switch, the high memory area is allocated on a first-come, first-served basis.

Examples

To use HIMEM.SYS with its default settings, include this command in your CONFIG.SYS file:

```
DEVICE=HIMEM.SYS
```

To ensure that no program smaller than 32KB will use the high memory area, include this command in your CONFIG.SYS file:

```
DEVICE=HIMEM.SYS /HMAMIN=32
```

PRINTER.SYS *External*

Purpose

Supports code page switching for parallel printers.

Format

```
DEVICE = [d:][path]PRINTER.SYS LPTx(:)=
         (type[,[hwcp][,n]])
```

Description

Use this device driver only if you want to do code page switching on your parallel printer (LPT1, LPT2, or LPT3).

If you have the proper supporting hardware, you can install this device driver to switch between prepared code pages for your printer. See Chapter 9, Code Page Switching, for more information about using this device driver to support code page switching.

You specify the following parameters:

Parameter	*Description*
LPT*x*	Printer device. Valid values: LPT1 LPT2 LPT3 (You can substitute PRN for LPT1)
type	Printer. Valid values: 4201 (IBM 4201 Proprinter family, IBM 4202 Proprinter XL) 4208 (IBM 4207 Proprinter X24, IBM 4208 Proprinter XL24) 5202 (IBM 5202 Quietwriter III)

Parameter	*Description*
hwcp	Hardware code page built into the printer. Valid values: 437 United States 850 Multilingual 860 Portugal 863 French Canada 865 Norway
n	Number (from 0 to 12) of additional code pages you want to use. This number is hardware dependent.

Example

To set up code page switching for a printer that emulates the IBM Proprinter II Model 4201, such as the Epson FX-286™, and that has a hardware code page of 437 built-in and the capability to access one more code page, enter this command in your CONFIG.SYS file:

```
DEVICE=C:\MSDOS\PRINTER.SYS LPT1:=
       (4201,437,1)
```

SMARTDRV.SYS *External*

Purpose

Provides disk caching for computers with hard disks and extended or expanded memory.

Format

```
DEVICE = [d:][path]SMARTDRV.SYS [size][/A]
```

Description

SMARTDRV.SYS allocates a disk cache—a block of memory to hold the contents of frequently accessed files. When a program needs to read a file from disk, it checks to see if the information is stored in the cache. If it is, the program can read the file without accessing the physical disk. Because memory access is much faster than disk access, SMARTDRV.SYS can enhance the performance of your application programs.

To use SMARTDRV.SYS you must have a hard disk and either extended or expanded memory.

Note
If you use another disk caching program, do not use SMARTDRV.SYS.

You can use these parameters with SMARTDRV.SYS:

Parameter	*Function*
size	Specifies the number of kilobytes to use for the cache. The default is 256, unless you use the /A switch. If you use /A and omit *size*, SMARTDRV.SYS uses all of your expanded memory.
/A	Uses expanded memory for the cache. If you do not include /A, SMARTDRV.SYS uses extended memory.

Follow these rules in deciding whether to use extended or expanded memory:

- If you have only extended memory, use extended.
- If you have only expanded memory, use expanded.
- If you have memory that can be configured as either extended or expanded, use extended memory for SMARTDRV.SYS and reserve enough expanded memory for your application programs.

Examples

To create a 256KB disk cache in extended memory, include this command in your CONFIG.SYS file:

```
DEVICE=SMARTDRV.SYS
```

To create a 512KB disk cache in expanded memory, include this command in your CONFIG.SYS file:

```
DEVICE=SMARTDRV.SYS 512 /A
```

To use all of your expanded memory for disk caching, include this command in your CONFIG.SYS file:

```
DEVICE=SMARTDRV.SYS /A
```

VDISK.SYS *External*

Purpose

Supports virtual (RAM) disks.

Format

```
DEVICE = [d:][path]VDISK.SYS [cap] [sec] [dir] [/A]
         [/E]
```

Description

VDISK.SYS sets up an area in RAM memory that simulates a real disk drive. This area, once established, is commonly referred to as a virtual disk or a RAM disk.

The advantage of using a virtual disk is speed. Because RAM can be accessed quickly by the computer, all disk read/write operations are performed many times faster than using an actual disk. However, the contents of RAM are lost each time you reset or turn off the computer.

You can customize your virtual disk by including parameters on the command line for virtual disk size, sector size, and number of directory entries. If you have extended memory installed in your computer, you can use the /E switch to create a virtual disk in memory at or above 1MB. If you have expanded memory, you can use the /A switch to create the virtual disk in expanded memory.

The parameter *cap* represents the size of the virtual disk in kilobytes. The range is from 1 to the amount of memory available on your computer. If you don't specify a *cap* value, the default of 64KB is assigned, subject to the following conditions:

- If you specify an unacceptable value for *cap* (less than 1 or greater than available memory), the default value of 64KB is assigned.
- VDISK always leaves at least 64KB of RAM memory for use by the system. Therefore, if you specify a value that would cause less than this amount to be set aside, VDISK automatically decreases its own size accordingly.

- If you have installed more than one virtual disk in the CONFIG.SYS file and a subsequent DEVICE = VDISK.SYS command finds less than 64KB of memory left, VDISK does not install the extra virtual disk. You see an error message.

The parameter *sec* selects the sector size in bytes. Valid sizes are 128, 256, or 512. If you omit a size or enter an incorrect size, the default value of 128 is assigned.

When MS-DOS reads a disk, it always reads a minimum of one sector. If you save small files (less than 128 bytes) to the virtual disk, use a value of 128 for *sec*. If you save large files (more than 512 bytes) to the virtual disk, use a value of 512 to speed up disk access.

The parameter *dir* specifies the number of directory entries recognized by the virtual disk. Acceptable values are from 2 to 1024. If you do not specify *dir*, a default value of 64 is assigned.

VDISK modifies the value for *dir* as it is loaded. VDISK automatically selects the correct number of directories that fill a sector. Each directory entry uses 32 bytes. Therefore, if you choose a sector size of 128 bytes, the number of directories is always a multiple of 4, regardless of the value used for *dir*. For 256 sectors, the multiple is 8, and for 512 sectors the multiple is 16.

/A allows you to use VDISK with expanded memory (LIM-EMS). Expanded memory is an option for your computer. Once installed, it allows your computer to access more than 640KB of memory. You cannot use /A with /E.

/E allows you to use VDISK with extended memory. Extended memory is an option for your computer. Once installed, it allows your computer to access memory at and beyond 1MB. You cannot use /E with /A.

To configure your system with more than one virtual disk, simply add the required number of DEVICE = VDISK.SYS commands to your CONFIG.SYS file. The only limit to the number of drives you can install is the amount of memory available.

Example

To install a virtual disk of 160KB capacity with 512 sectors and 64 directory entries in standard memory, place the following command in CONFIG.SYS:

```
DEVICE=VDISK.SYS 160 512 64
```

```
Microsoft VDISK version x.xx, VDISK Virtual Disk x:
Buffer size:                xxx KB
Sector size:                xxx
Directory entries:          xx
```

XMAEM.SYS

External

Purpose

Converts extended memory on 80386-based IBM computers to LIM 4.0 expanded memory.

Format

`DEVICE = [d:][path]XMAEM.SYS [size]`

Description

XMAEM.SYS uses extended memory on an 80386-based IBM computer to emulate Lotus/Intel/Microsoft (LIM) 4.0 expanded memory. To use XMAEM.SYS, you must have an 80386-based IBM computer, such as the PS/2 Model 70 or Model 80, and you must have extended memory.

Note
To convert extended memory to expanded memory on non-IBM 80386-based computers, use EMM386.SYS. Do not use XMAEM.SYS.

The *size* parameter specifies the number of 16KB memory pages to use. The minimum is 64 (1MB). The default and maximum are the total amount of available extended memory.

XMA2EMS.SYS *External*

Purpose

Manages IBM-supplied LIM 4.0 expanded memory on IBM computers.

Format

```
DEVICE = [d:][path]XMA2EMS.SYS
         [FRAME=address][P254=address][P255=address]
         [/X:size]
```

or

```
DEVICE = [d:][path]XMA2EMS.SYS
         [P0=address][P1=address][P2=address]
         [P3=address][P254=address][P255=address]
         [/X:size]
```

Description

XMA2EMS supports the Lotus/Intel/Microsoft (LIM) 4.0 expanded memory specification on IBM computers using IBM-supplied expanded memory hardware. To use XMA2EMS, you must have an IBM computer and one of the following:

- IBM 2MB Expanded Memory Adapter
- IBM Personal System/2 80286 Expanded Memory Adapter /A
- IBM Personal System/2 80286 Memory Expansion Option
- 80386-based extended memory converted to expanded memory using the XMAEM.SYS device driver.

Note

If you have a non-IBM computer with an expanded memory option board, use the device driver that came with the option board. Do not use XMA2EMS.SYS.

If you have a non-IBM computer that is compatible with the IBM PC AT and you are not using an expanded memory option board, use the EEMM286.SYS device driver to convert extended memory to expanded memory. Do not use XMA2EMS.SYS.

If you have a non-IBM 80386-based computer, use the EMM386.SYS device driver to convert extended memory to expanded memory. Do not use XMA2EMS.SYS.

You may use the following parameters with XMA2EMS.SYS:

Parameter	*Function*
FRAME=*address*	Specifies a starting page address for a contiguous 64KB memory block comprising pages P0, P1, P2, and P3. *address* must be a hexadecimal value between A000 and E000.
P*n*=*address*	Specifies a starting page address for a 16KB memory block. *n* must be 0, 1, 2, 3, 254, or 255. If you include FRAME=*address*, *n* must be 254 or 255.
/X:*size*	Specifies the number of 16KB expanded memory pages to allocate. The minimum is 4 (64KB). The maximum and default are the total amount of available expanded memory.

If you are using XMA2EMS.SYS on an 80386-based computer, your CONFIG.SYS file must include a DEVICE=XMAEM.SYS command before the DEVICE=XMA2EMS.SYS command.

Chapter 7

Batch Processing Commands

Creating Batch Files 7-3
Using an AUTOEXEC.BAT File 7-5
Using Replaceable Parameters 7-6
Using Environment Variables 7-7
Using the Batch Processing Commands 7-9
CALL 7-10
ECHO 7-11
FOR 7-13
GOTO 7-15
IF 7-16
PAUSE 7-18
REM 7-19
SHIFT 7-20

Creating Batch Files

You may find yourself typing the same sequence of commands over and over to perform a task. With MS-DOS you can put this command sequence into a special file called a batch file. You can then run the sequence of commands by simply typing the name of the batch file. MS-DOS executes the commands in a batch file just as if you had typed each command from the keyboard.

You can create a batch file using a text editor, such as EDLIN, or using the COPY command, as shown here. When you create a batch file, you must include a .BAT extension. When you run the batch file, you type the filename only; you don't type the file extension.

To create a batch file to format and check a new disk, follow these steps:

1. Type the following command and press **Enter**:

   ```
   COPY CON CHECKNEW.BAT
   ```

   This command tells MS-DOS to copy the information from the keyboard to the file CHECKNEW.BAT.

2. Type the following lines, pressing **Enter** after each line:

   ```
   REM    This is a file to format and
   REM    check new disks.
   REM    It is named CHECKNEW.BAT.
   PAUSE  Insert new disk in drive B:.
   FORMAT B:
   CHKDSK B:
   ```

   REM and PAUSE are batch file commands described later in this chapter.

3. After typing the last line, press **Ctrl Z** followed by **Enter** to save the batch file. MS-DOS displays the following message:

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

4. To execute the file, type the following command and press **Enter**:

```
CHECKNEW
```

MS-DOS executes the commands in sequence just as if you typed each line as an individual command.

Remember the following things about batch files:

- You must name each batch file with a .BAT extension.
- To execute a batch file, you type only the filename and not the extension.
- You can terminate a batch file while it is running by pressing **Ctrl C** or **Ctrl Break**.
- If you are running a batch file from a diskette and, during the processing, you remove the diskette, MS-DOS prompts you to reinsert it so it can complete processing the batch file.
- You can specify the name of another batch file as the last command in a batch file. MS-DOS will then execute the next batch file after it executes all the commands in the first batch.
- You can use the CALL command to run one batch file from within another. When MS-DOS finishes executing the second batch file, it resumes execution of the first batch file, at the first following the CALL command.
- You can use any of the redirection symbols (| < >) in a batch file. For more information on using these symbols, see Chapter 3, Using MS-DOS Commands.
- MS-DOS ignores any line in a batch file that starts with a colon (:). Therefore, you can use the colon to add comment lines to your batch file.
- You may use an @ character preceding a command in a batch file to prevent the command line from displaying on the screen.

- Changing the directory or drive affects subsequent commands in the batch file.
- Setting environment strings affects subsequent commands in the batch file.
- Do not use MS-DOS command names for batch files.

Using an AUTOEXEC.BAT File

An AUTOEXEC.BAT file lets you execute specific MS-DOS commands automatically when you start MS-DOS. This is useful when you want to run a specific application or set certain system parameters each time you start your computer.

Each time you start your computer, MS-DOS searches the root directory of the default disk drive for a file named AUTOEXEC.BAT. If it finds an AUTOEXEC.BAT file, MS-DOS immediately processes it, bypassing the date and time prompts. (If MS-DOS does not find an AUTOEXEC.BAT file, it displays the date and time prompts.)

If your computer does not contain a real-time clock, include the DATE and TIME commands in your AUTOEXEC.BAT file. MS-DOS needs the correct date and time information to keep your directories current.

An AUTOEXEC.BAT file can help you use MS-DOS more efficiently. You can set the date and time, set a path and any other system options you use, change the system prompt, and start executing an application program.

For example, to automatically run the DATE and TIME commands, set the MS-DOS command prompt so that it displays the current drive and directory, then clears the screen, follow these steps:

1. Make sure the current drive and directory are the ones you use to boot MS-DOS. That is, if you boot from a hard disk, log onto C:\. If you boot from a diskette, insert that diskette in drive A. Then type `A:` and press **Enter**.

2. Type the following and press **Enter**:

```
COPY CON AUTOEXEC.BAT
DATE
TIME
PROMPT $P$G
```

3. After the last line, press **Ctrl Z** followed by **Enter** to copy these lines to the AUTOEXEC.BAT file.

Using Replaceable Parameters

Replaceable parameters let you run batch files with different sets of data. You specify the replaceable parameters in your batch file and then enter the actual data on the command line when you execute the batch file. The replaceable parameters are %0 through %9.

For example, you can create a batch file that sorts a file based on a specific string of characters. Then each time you run the batch file, you can specify a different file and/or a different string of characters.

Follow these steps:

1. Type the following command and press **Enter**:

```
COPY CON SORTER.BAT
```

2. Type the following command lines, pressing **Enter** after each line:

```
TYPE %2 ¦ FIND "%1" > %3
TYPE %3 ¦ SORT > PRN
DEL %3
```

3. Assume the following file, named CUSTOMER, exists on drive A:

```
SHORES, BETTY   NORTH
MOYNIHAN, ANN   SOUTH
KRAIG, HEIDY    NORTH
MARTIN, PETE    EAST
LENNON, PAT     SOUTH
EVANS, RICK     NORTH
```

4. To print an alphabetical list of the customers in the north, type the following command line followed by **Enter**:

```
SORTER NORTH A:CUSTOMER TEMP.FIL
```

The output on the printer looks like this:

```
EVANS, RICK     NORTH
KRAIG, HEIDY    NORTH
SHORES, BETTY   NORTH
```

The result is the same as if you had typed each of the following commands:

```
TYPE A:CUSTOMER ¦ FIND "NORTH" > TEMP.FIL
TYPE TEMP.FIL ¦ SORT > PRN
DEL TEMP.FIL
```

Remember these things when using replaceable parameters:

- You can specify up to 10 replaceable parameters (%0–%9). To use more than 10, use the SHIFT command described later in this chapter.
- MS-DOS always replaces the %0 parameter with the drive (if specified) and the filename of the batch file. (In the above example, %0 is SORTER.)
- If a filename or parameter in the batch file has a percent sign in it, you must type the percent sign twice. For example, to specify the file ABC%.DAT, you must type ABC%%.DAT.

Using Environment Variables

You can also access environment variables with replaceable parameters in a batch file. Enter the parameter with a percent sign before and after the name of the environment variable, such as %FILE%.

For example, to create a batch file that moves a file you want to delete into a separate directory, follow these steps:

1. Type the following command and then press **Enter**:

```
COPY CON MYDEL.BAT
```

2. Type the following commands, pressing **Enter** after each line:

```
ECHO OFF
ECHO Before using this batch file,
ECHO you must specify the directory.
ECHO At the MS-DOS prompt, type:
ECHO SET DELDIR = directory
ECHO Press Ctrl C to exit if you
ECHO need to set DELDIR.
PAUSE
COPY %1 %DELDIR%
DEL %1
DIR /W %DELDIR%
ECHO Complete
```

3. After the last line, press **Ctrl Z** and then press **Enter**.

4. For MYDEL.BAT to work, you must create your delete directory. To create the DELETED directory, type the following command and press **Enter**:

```
MD \DELETED
```

5. Use the SET command to specify a directory named DELETED:

```
SET DELDIR = \DELETED
```

6. To move the file REPORT10.SLS to the delete directory, type the following command and press **Enter**:

```
MYDEL REPORT10.SLS
```

   MS-DOS automatically replaces the %DELDIR% parameter with the directory named DELETED.

Using the Batch Processing Commands

The batch processing commands described in the remainder of this chapter add power and flexibility to your batch files. You can execute any of these batch processing commands from the MS-DOS command prompt. These commands are called batch processing commands because they are most commonly used in batch files, and because they affect batch file operations.

CALL

Internal

Purpose

Allows one batch file to call another batch file without ending the first batch file.

Format

`CALL [d:][path]filename [arguments]`

Description

Use this command to execute one batch file from another batch file. MS-DOS executes the specified batch file and then returns to the calling batch file.

Use the *arguments* parameter to specify any arguments, such as replaceable parameters, required by the batch file being called.

You can specify the CALL command on any line of a batch file.

The file that you call must be a batch file (that is, it must have a .BAT extension).

A batch file can call itself. Make sure the batch file eventually ends.

Note
Do not use piping or redirection with the CALL command.

Example

The following sequence of commands sends control to the ERRFORM batch file if an error occurs when you format the diskette in drive A:

```
ECHO OFF
FORMAT A: /S
IF ERRORLEVEL 1 CALL ERRFORM
ECHO End of batch file
```

After the ERRFORM batch file is executed, control returns to the ECHO command that follows the CALL statement.

ECHO

Internal

Purpose

Turns on and off command echoes on your screen.

Format

```
ECHO [ON]
```

or

```
ECHO [OFF]
```

or

```
ECHO [message]
```

Description

Use ECHO to control the display of batch file commands on the screen. Normally, MS-DOS displays, or echoes, commands in a batch file while they are executing. When you turn on or reset the system, ECHO is on.

Type `ECHO OFF` to turn this feature off. Type `ECHO ON` to turn it back on.

You can display the current setting of the command by typing `ECHO` without the ON or OFF parameter.

If you are using a batch file, you can type `ECHO` *`message`* to display messages during the execution. These messages are displayed on the standard output device regardless of the ECHO setting.

This command does not affect the messages that are produced during command execution.

Examples

Include ECHO in a batch file such as this one:

```
ECHO OFF
REM Command display off
TREE B:
ECHO ON
TREE B:
```

When this batch file executes, the screen displays ECHO OFF but does not display the REM or TREE command. As TREE executes, the output of the TREE command displays. When ECHO is back on, the second TREE command displays. Then, as before, the output of the TREE command displays.

To turn off the display of normal output from a command (COPY, for example), redirect that output to the null device, like this:

```
COPY *.BAK BAKDIR >NUL
```

You will gain execution speed by suppressing screen display in this manner. Error messages still appear on screen, however, when you redirect output to NUL.

To turn ECHO off without displaying the command ECHO OFF, use @, like this:

```
@ECHO OFF
```

FOR

Internal

Purpose

Performs a single repetitive operation on a set of files, devices, or system parameters.

Format

FOR *%%c* IN *set* DO *command*

Description

Use the FOR command to repeat certain batch operations for each item in a list. The format of this command means FOR each item IN a set of items DO the specified command.

The parameters are:

- *c* can be any character (except the numbers 0 through 9 to avoid confusion with the %0—%9 batch replaceable parameters)
- *set* is either a filename or a list of possible values
- *command* is any MS-DOS command.

The %%c variable applies sequentially to each item in the parameter, and the command then evaluates each item. If a member in the set of items is an expression with an asterisk (*) or a question mark (?), then the variable is set to each matching pattern from disk. You can only use * or ? in one item in the set; FOR ignores any other item in the set.

You must include two percent signs (%%) so that after batch parameter (%0–%9) processing is done, one percent sign (%) remains. If you enter only one percent sign, such as %f, it appears as an error (bad parameter reference). MS-DOS discards the %f, which makes it unavailable when you execute FOR. However, if you use FOR outside a batch file, you only need to type one percent sign.

Examples

To assemble all files with the extension .ASM, type:

```
FOR %%f IN (*.ASM) DO MASM %%f
```

To list all files in the SALES directory, type:

```
FOR %%f IN (LEDGER\SALES\*.*) DO REM %%f
```

GOTO

Internal

Purpose

Jumps to a labelled line within the same batch file.

Format

GOTO *label*

Description

Use this command to skip to another location in your batch file. MS-DOS starts executing commands from that location. MS-DOS executes the command on the line following the label.

Typically, you use this command to skip over an unwanted portion of commands, or to return to a line in order to repeat commands.

If you do not define the specified *label* in the batch file, MS-DOS displays the error message `Label not found` and terminates the batch file.

You define a label using the colon as the first character on a line followed by the label name (*:label*). MS-DOS ignores the remainder of the line following the colon except when searching for a label. In this way, you can also use the colon to enter comments in your batch file.

Example

The following sequence of commands sends control to the END statement only if no errors occur when you format the diskette in drive A:

```
:BEGIN
ECHO OFF
FORMAT A: /S
IF NOT ERRORLEVEL 1 GOTO END
ECHO AN ERROR OCCURRED DURING FORMATTING.
:END
ECHO END OF BATCH FILE.
```

IF

Internal

Purpose

Executes a second command depending on whether a specified condition is true or false.

Format

`IF [NOT] condition command`

Description

Use this command to test for some logical condition (such as an error in a previous program or the existence of a file), and then to execute another command depending on the results of the test. You can enter any valid MS-DOS command. The *condition* parameter is one of the following:

Condition Parameter	*Result*
ERRORLEVEL *number*	This condition is true only if the previously executed program had an exit code number that is the same or higher than this *number*.
string1 == *string2*	This condition is true only if *string1* and *string2* are identical after parameter substitution (do not enter embedded delimiters in strings).
EXIST *filename*	This condition is true only if *filename* exists.

If you enter the NOT parameter, the condition is true only if *condition* is false.

The command descriptions in Chapter 4 tell whether a command sets an ERRORLEVEL parameter.

Examples

To display a message based on whether the file called 620.TXT exists, type the following and press **Enter**:

```
IF NOT EXIST 62Ø.TXT ECHO Can't find file
```

If the file does not exist, MS-DOS displays the message:

```
ECHO Can't find file
```

To link to another file based on the exit code of a previous program, type the following and press **Enter**:

```
IF NOT ERRORLEVEL 3 LINK %1,,;
```

If the previously executed program did not have an exit code of three or higher, then link the file identified by the first replaceable parameter.

To transfer control based on a replaceable parameter, type:

```
IF %1 = = QUIT GOTO END
```

If the first replaceable parameter is QUIT, then transfer control to END.

PAUSE *Internal*

Purpose

Suspends execution of the batch file.

Format

PAUSE [*comment*]

Description

Use PAUSE to suspend the execution of your batch file until you press any key. Use this command in your batch file when you need to change diskettes or perform some other action.

When MS-DOS finds a PAUSE command, you see the prompt:

```
Press any key to continue . . .
```

After you perform the desired function, press any key to continue.

Use the *comment* parameter in your PAUSE command to display a message on the screen during program suspension. If ECHO is on, your comment displays before the MS-DOS message instructing you to strike a key when ready. A comment may be up to 121 characters in length.

You can stop a batch job at any time. Just press **Ctrl C**.

Example

Type the following sequence of commands in your batch file. A pause occurs before MS-DOS displays the directory on drive B. This allows you to insert a new diskette in drive B.

```
DIR A:
ECHO Display the directory of drive B next
PAUSE
DIR B:
```

REM

Internal

Purpose

Adds remarks to batch files.

Format

`REM [comment]`

Description

Use this command to add comment lines in a batch file.

The comment can be as many as 123 characters in length, and may include spaces, tabs, commas, and quote marks. If you need to use other punctuation symbols (¦, for example), surround those symbols in double quotes.

If ECHO is off, MS-DOS does not display the word REM or the comment.

You can also use REM to add comments to the CONFIG.SYS file (see Chapter 6).

Another way to add a comment to a batch file is to precede the commentary text with a colon. The colon signals the start of a label, and if a label is not used as the target of a GOTO command, MS-DOS ignores it.

Example

You can enter the following sequence of commands in your batch file:

```
REM This file checks new disks
REM It is named NEWDISK.BAT
PAUSE Insert new disk in drive B
FORMAT B:/S
DIR B:
CHKDSK B:
```

The remark lines tell you what the file does and gives the name of the file.

SHIFT *Internal*

Purpose

Enables more than 10 replaceable parameters within a batch file.

Format

SHIFT

Description

Use this command to shift batch file parameters up one place. SHIFT makes it easier for a batch file to process a list of parameters that can vary in number. SHIFT changes the replaceable parameters by shifting up each value for %0 through %9.

Example

If your data file parameters are:

```
%Ø = x
%1 = y
%2 = z
%3-%9 are empty
```

Then a SHIFT command produces the following:

```
%Ø = y
%1 = z
%2-%9 are empty
```

If you give more than 10 parameters on a command line, those that appear after %9 move one at a time into %9 when you enter successive SHIFT commands.

Chapter 8

Programming Tools

Introduction 8-3
DEBUG 8-4
Starting DEBUG 8-4
DEBUG Command Format 8-6
DEBUG Commands 8-9
A (Assemble) 8-10
C (Compare) 8-10
D (Dump) 8-10
E (Enter) 8-11
F (Fill) 8-12
G (Go) 8-12
H (Hex) 8-12
I (Input) 8-13
L (Load) 8-13
M (Move) 8-14
N (Name) 8-14
O (Output) 8-15
P (Proceed) 8-16
Q (Quit) 8-17
R (Register) 8-17
S (Search) 8-19
T (Trace) 8-19
U (Unassemble) 8-20
W (Write) 8-21
XA (Allocate Expanded Memory) 8-22
XD (Deallocate Expanded Memory) 8-22
XM (Map Expanded Memory Pages) 8-23
XS (Get Expanded Memory Status) 8-23
EDLIN 8-29
Starting EDLIN 8-30
Command Information 8-31
Special Editing Features 8-33
EDLIN Commands 8-34
A (Append) 8-34

C (Copy) 8-35
D (Delete) 8-36
E (End) 8-37
I (Insert) 8-38
L (List) 8-39
M (Move) 8-41
P (Page) 8-41
Q (Quit) 8-42
R (Replace) 8-42
S (Search) 8-45
T (Transfer) 8-46
W (Write) 8-47
EXE2BIN 8-48
LINK 8-50
Files Used By LINK 8-53
Starting LINK 8-54
LINK Parameters 8-56
LINK Switches 8-57
LINK Processing 8-62

Introduction

This chapter describes the MS-DOS commands you use when you write your own programs. The following commands are programming tools:

Command	*Description*
DEBUG	Monitors and modifies the actual content of computer memory
EDLIN	Creates and edits text files
EXE2BIN	Converts executable (.EXE) files to binary format
LINK	Combines object code program modules into executable programs.

You can use the EDLIN editor to create any text files, including your system files such as AUTOEXEC.BAT and CONFIG.SYS.

Normally, the DEBUG, EXE2BIN, and LINK commands are used only by experienced programmers.

DEBUG *External*

Purpose

Monitors and modifies the contents of computer memory.

Format

`[d:][path]DEBUG [[d:][path]filename][parameters]`

Description

Use DEBUG to debug programs. This utility lets you test binary and executable object files. You can also alter the contents of files and disks (this is called patching) as well as change CPU registers. To use DEBUG, you need to be familiar with programming and the use of debugging tools.

You can use these two control codes with the DEBUG commands:

- **Ctrl C** stops operation of a DEBUG command.
- **Ctrl S** halts the scrolling of the display. You can restart the display by pressing any key other than **Ctrl C** or **Ctrl S**.

Starting DEBUG

You can start DEBUG in one of two ways.

The first way to start DEBUG is to load DEBUG and then enter a specific filename.

1. At the command prompt, type the following and press **Enter**:

   ```
   DEBUG
   ```

   The DEBUG program loads and presents you with a hyphen (–) prompt.

2. Use the Name (N) command along with the filename, and include the appropriate parameters. At the hyphen prompt, type the following and press **Enter**:

   ```
   N filename [parameters]
   ```

For example:

```
N TESTFILE.EXE/D
```

3. Now you can execute commands to act on memory, files, registers, or disk sectors, and to load and save files.

The second way to start DEBUG is to load DEBUG and the specified file in one step.

1. At the command prompt, type the following and press **Enter** (you do not use the N command when you start DEBUG in this manner):

   ```
   DEBUG filename [parameters]
   ```

   For example:

   ```
   DEBUG TESTFILE.EXE/D
   ```

   This loads the file TESTFILE. The /D option invokes the double-precision routines.

2. Now you can execute commands to act on memory, files, registers, or disk sectors, and to load and save files.

When DEBUG loads a file into memory, it starts at address 0100 hexadecimal in the lowest available segment. It also loads the number of bytes placed in memory into the BX:CX registers.

Remember these two points when using DEBUG with files:

- When working with a .COM or .EXE file, if you try to modify the program header below address 5C, DEBUG stops.
- Do not attempt to restart a program after you see the message `Program terminated normally`. Use the Name and Load commands to reload the program, and then rerun it.

DEBUG Command Format

Each DEBUG command you use is one or two letters which you follow with one or more parameters. The DEBUG commands and their parameters are shown in the following table. The standard MS-DOS control and editing function keys apply while using DEBUG, and you can use either upper- or lowercase commands.

Commands and Parameters	*Function*
A [*address*]	Assemble
C *range address*	Compare
D [*range*]	Dump
E *address* [*list*]	Enter
F *range list*	Fill
G [=*address* [*address...*]]	Go
H *value value*	Hex
I *value*	Input
L [*address* [*drive: record record*]]	Load
M *range address*	Move
N *filename* [*filename*]	Name
O *value byte*	Output
P [=*address*][*number*]	Proceed through loop or subroutine
Q	Quit
R [*register-name*]	Register
S *range list*	Search
T [=*address*] [*value*]	Trace
U [*range*]	Unassemble
W [*address* [*drive:record record*]]	Write
XA *count*	Expanded memory allocate
XD *handle*	Expanded memory deallocate
XM *lpage ppage handle*	Expanded memory map pages
XS	Expanded memory status

All DEBUG commands accept parameters, with the exception of the Quit command. You need not separate parameters with delimiters (spaces or commas), but you must enter a delimiter between two consecutive hex values. The following examples are equivalent:

```
dcs:1ØØ 11Ø
d cs:1ØØ 11Ø
d,cs:1ØØ,11Ø
```

When a format error occurs in a DEBUG command, DEBUG reprints the command line and displays a second line containing an arrow, which pinpoints the error. For example:

```
dcs:1ØØ cs:11Ø
        ^ Error
```

You can use the following command parameters with DEBUG:

address

This is a two-part designation consisting of either an alphabetic segment register designation or a 4-digit segment address, plus an offset value. If you omit the segment designation or segment address, DEBUG uses the default. DS is the default for all commands except G, L, T, U, and W, for which the default is CS. All numeric values are hexadecimal. For example:

```
1ØØ
CS:Ø1ØØ
Ø4BA:Ø1ØØ
```

You must enter a colon between a segment designation (whether numeric or alphabetic) and an offset value.

byte

This is a 2-digit hexadecimal value that DEBUG places in or reads from an address or register.

drive

This is a 1-digit hexadecimal value identifying the drive a file loads from or writes to. The valid values are: 0 = A:, 1 = B:, 2 = C:, 3 = D:, and so forth.

list

This is a series of strings or byte values, which must be the last parameter on the command line. For example:

```
ecs:1ØØ 42 45 52 54 41
fcs:1ØØ 1FF 42 45 52 54 41
```

range

This parameter has two forms:

- *Address address* refers to two addresses.
- *Address* L *value* where *value* is the number of bytes the command is to operate on, and L80 is the default. Do not use this form if another hex value follows the *range*, because DEBUG interprets the hex value as the second address of the range.

  These examples are legal:

  ```
  CS:1ØØ 11Ø
  CS:1ØØ L 1Ø
  CS:1ØØ
  ```

  This example is illegal and produces an error:

  ```
  CS:1ØØ CS:1ØØ
         ^ Error
  ```

  The limit for range is 10000 hex. To specify a value of 10000 hex, type ØØØØ (or only Ø).

record

This is a 1- to 3-digit hexadecimal value, which you use to indicate the logical record number on the disk and the number of disk sectors you want to write or load. Logical records correspond to sectors. However, their numbering differs because they represent the entire disk space.

string

This is any number of characters you enclose in quotation marks. Quotation marks can be either single (') or double ("). If you require quotation marks to delimit within a string, you can use two sets of quotation marks or a combination. For example, the following strings are legal:

```
'This "string" is okay.'
'This ''string'' is okay.'
"This 'string' is okay."
"This ""string"" is okay."
```

These strings are illegal:

```
'This 'string' is not okay.'
"This "string" is not okay."
```

DEBUG uses the ASCII values of the characters in the string as a list of byte values, which means abc is the same as 61 62 63.

value

This is a hexadecimal value of up to 4 digits. Use this parameter to specify a port number or the number of times a command repeats its function.

count

The number of 16K pages to allocate

handle

An EMS handle number

lpage

The logical page of the handle to map

ppage

The physical page that the logical page will be mapped into

DEBUG Commands

The remainder of this section summarizes the DEBUG commands and their uses.

A (Assemble)

Assembles 8086/8087/8088 mnemonics into memory.

Format: A[*address*]

Use this command to assemble 8086/8087/8088 mnemonics directly into memory at *address*. Enter data as hexadecimal values after mnemonics on separate lines. You must be familiar with assembly language programming to use this command.

If DEBUG detects a format error, it points out the error and redisplays the current assembly address.

C (Compare)

Compares portions of memory.

Format: C *range address*

Use this command to compare the portion of memory you specify in *range* to another portion of the same size beginning at *address*. Nothing displays if the two areas of memory are identical; otherwise, the difference displays with the addresses bracketing the different bytes.

For example, both of these commands compare the block of memory from $100 to $1FF with the block of memory from $300 to $3FF:

```
C100, 1FF 300
C100L100 300
```

D (Dump)

Displays a portion of memory.

Format: D[*range*]

Use this command to display the contents of the region of memory you specify in *range*. If you omit *range*, DEBUG displays 128 bytes starting at the first address (DS:100) after the address displayed by the previous Dump command.

The Dump display contains two portions. Each byte is shown in hexadecimal on the left. The ASCII representation of the byte appears on the right and identifies nonprinting characters with a period. Each display line shows 16 bytes with a hyphen between the eighth and ninth bytes and begins on a 16-byte boundary.

An example of this display is shown below:

```
-d
0A63:0100   B9 2C 00 BE 20 00 8A 14-B4 02 CD 21 46 E2 F7 CD 9,.>...4.M!FbwM
0A63:0110   20 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ................
0A63:0120   54 68 69 73 20 69 73 20-61 20 74 65 73 74 20 66 This is a test f
0A63:0130   69 6C 65 20 66 6F 72 20-75 73 65 20 77 69 74 68 ile for use with
0A63:0140   20 44 45 42 55 47 2E 0D-0A 0A 07 00 00 00 00 00 DEBUG..........
0A63:0150   20 20 74 68 65 20 20 66-69 6C 65 00 08 73 70 65   the file..spe
0A63:0160   63 69 66 69 63 61 74 69-6F 6E 20 20 77 69 6C 6C cification will
0A63:0170   20 6E 65 65 64 20 74 6F-20 62 65 20 63 68 61 6E need to be chan
```

E (Enter)

Enters a value into memory.

Format: E *address* [*list*]

Use this command to enter byte values into memory at the address you specify. The values in *list* replace the contents of one or more bytes beginning at *address*; in the event of an error, no byte values change.

If you omit *list*, DEBUG displays the specified address and its contents. It then repeats the address on the next line and waits for your input. You can respond in one of the following ways:

- Replace the value of this address by typing a new value.
- Advance to the next byte by pressing the space bar.
- Return to the preceding byte by typing a hyphen (−).
- Terminate the command by pressing **Enter**.

For example, the command E 100 01 02 changes the byte at location 100 to 01 and the byte at location 101 to 02.

F (Fill)

Enters values into memory.

Format: F *range list*

Use this command to fill the addresses in the *range* with the values in the *list*. For example, if you type the following command:

```
FØ4BA:1ØØ L 1ØØ 42 45 52 54 41
```

DEBUG fills memory locations from 04BA:100 to 04BA:1FF with the values in the *list* until all bytes are filled.

G (Go)

Allows you to execute a program at an address you specify.

Format: G[=*address1* [*address...*]]

Use this command to execute the program currently in memory at *address1*. If you don't specify the optional address parameters, the program executes normally.

Use the =*address1* parameter to run the program from a particular address. Use the other optional addresses to set breakpoints. You may set up to ten breakpoints.

H (Hex)

Displays the sum and difference of two hexadecimal values.

Format: H *value value*

Use this command to display the sum and difference of the two values. First, DEBUG adds the two parameters; then it subtracts the second parameter from the first. The results display on one line: first the sum, then the difference. For example, type H19F 1ØA. DEBUG performs the calculations and displays the results:

```
Ø2A9 ØØ95
```

I (Input)

Displays a value.

Format: `I value`

Use this command to input and display one byte from the port you specify in *value*. You can specify a 16-bit port address.

L (Load)

Loads a file into memory.

Format: `L[address [drive record number]]`

Use this command to load a file into memory. After the file loads, the BX:CX registers contain the number of bytes read. You must name the file with the N command. DEBUG formats a filename in a file control block at CS:5C.

If you type the L command without any parameters, DEBUG loads the file specified by the N command into memory beginning at address CS:100 and sets the BX:CX registers to the number of bytes read. If you type the L command and include an *address* parameter, loading begins at the memory address you specify.

If you type all the parameters, absolute logical disk sectors load, not a file. The records are taken from the *drive* you specify. DEBUG begins loading with the first *record* and continues until the number of sectors you specify in the *number* parameter load.

For example, to load the file TEST1.COM after DEBUG loads, type the following commands:

```
-NTEST1.COM
-L
```

The N command names the file to load, and the L command carries out the loading process.

To load the boot sector of a diskette in drive A at address CS:100, type the following command:

```
-L,100,0,0,1
```

M (Move)

Moves a block of memory to a new location.

Format: `M` *`range address`*

Use this command to move a block of memory from *range* to the location beginning at the *address*. This command copies data from one area into another and writes over the new addresses.

However, this command first moves addresses that could be overwritten. Overlapping moves (that is, where part of the block overlaps some of the current addresses) always occur without loss of data.

Moves from higher addresses to lower addresses begin at the block's lowest address and work toward the highest. Moves from lower addresses to higher addresses begin at the block's highest address and work toward the lowest. If the addresses in the block being moved will not have new data written to them, the original data remains.

For example, type `MCS:100 110 CS:500.` DEBUG first moves address CS:110 to address CS:510, then CS:10F to CS:50F, and so on until CS:100 moves to CS:500. Use the D command to see the results of the move:

```
D CS:500.
```

N (Name)

Assigns new filenames.

Format: `N` *`filename`* `[`*`filename...`*`]`

Use this command to set filenames to perform two functions:

- You can use the Name command to assign a *filename* for a later Load or Write command. Thus, if you start DEBUG without naming a file, you must use the N command before you can load a file.
- You can use the Name command to assign *filename* parameters to the file that DEBUG is going to affect. For example, you can specify a set of data files that you want a program file to use.

These two functions may overlap. For example, the following set of DEBUG commands name, load, and then execute the file FILE1.EXE:

```
-NFILE1.EXE
-L
-G
```

If the file requires that you add some data files as parameters, you can use the Name command in the form:

```
-NFILE1.EXE ONE.DAT TWO.DAT
-L
-G
```

You can also specify these data files using the N command after loading FILE1.EXE:

```
-NFILE1.EXE
-L
-NONE.DAT TWO.DAT
-G
```

Although the Go command causes FILE1.EXE to execute as if you typed `ONE.DAT` and `TWO.DAT` at the MS-DOS command level, this can cause problems if you enter a Write command. If you execute a Write command at this point, it saves FILE1.EXE—the file you intend to debug—with the name ONE.DAT. To avoid this result, always execute a Name command before either a Load or a Write command.

O (Output)

Outputs a value.

Format: `O` *`value byte`*

Use this command to send the *byte* you specify to the output port you identify in *value*. A 16-bit port address is allowed.

For example, to send a value of 50H to port 3ACE, use this command:

```
-O 3ACE 5Ø
```

P (Proceed)

Traces a program.

Format: P

Use this command to trace a program while it executes so you can see the contents of all registers and flags next to the decoded instruction. Interrupts, subroutine calls, repeat-string instructions, and loop instructions are treated as a single operation. (To trace individual steps, use the Trace command.) Control is returned to the instruction immediately after the called routine. This is extremely useful in tracing across system calls and procedures.

WARNING
The P command connot be used to trace through ROM.

For example, the Trace command traces through each step of interrupt service routines:

```
AX=3000 BX=0000 CX=62BC DX=0000 SP=083D BP=0000 SI=0000 DI=0000
DS=1787 ES=1787 SS=1787 CS=1787 IP=0E35 NV UP EI PL NZ NA PO NC
1787:0E35 CD21        INT21
-T

AX=3000 BX=0000 CX=62BC DX=0000 SP=0837 BP=0000 SI=0000 DI=0000
DS=1787 ES=1787 SS=1787 CS=0273 IP=1460 NV UP DI PL NZ NA PO NC
0273:1460 2E          CS:
0273:1461 3A26FF0D    CMP   AH,[0DFF]
```

The Proceed command, however, traces over interrupts, subroutines and loop instructions without displaying every step inside them:

```
AX=3000 BX=0000 CX=62BC DX=0000 SP=083D BP=0000 SI=0000 DI=0000
DS=1787 ES=1787 SS=1787 CS=1787 IP=0E35 NV UP EI PL NZ NA PO NC
1787:0E35 CD21        INT   21
-P

AX=1E03 BX=0000 CX=0000 DX=0000 SP=083D BP=0000 SI=0000 DI=0000
DS=1787 ES=1787 SS=1787 CS=1787 IP=0E37 NV UP EI PL NZ NA PO NC
1787:0E37 3D031E      CMP   AX,1E03
-
```

Q (Quit)

Exits DEBUG.

Format: Q

Use this command, which has no parameters, to exit DEBUG and return to the command level without saving the file currently in operation.

R (Register)

Displays or alters the contents of CPU registers.

Format: R [*register-name*]

Use this command to display the contents of one or more CPU registers. If you do not type a *register-name*, the R command dumps the register save area and displays the contents of all registers and flags.

If you type a *register-name*, the 16-bit value of that register appears in hex and a colon appears as a prompt. You can type a value to change the register, or press **Enter** to leave the register unchanged.

The only valid register names are the following:

AX	DX	SI	ES	IP
BX	SP	DI	SS	PC
CX	BP	DS	CS	F

IP and PC both refer to the Instruction Pointer.

Any other entry for *register-name* results in a BR Error (bad register).

If you enter F as the *register-name*, DEBUG displays each flag with a two-character alphabetic code. To alter any flag, type the opposite two-letter code. You can either set or clear the flags.

The flags and their codes for SET and CLEAR are:

Flag Name	*Set*	*Clear*
Overflow	OV	NV
Direction	DN Decrement	UP Increment
Interrupt	EI Enabled	DI Disabled
Sign	NG Negative	PL Plus
Zero	ZR	NZ
Auxiliary Carry	AC	NA
Parity	PE Even	PO Odd
Carry	CY	NC

For example, to load 5AH into the BX register, you can use the R BX command:

```
-R BX
BX ØØØØ
:5A
```

DEBUG indicates that BX currently holds 0000. Type the value you want to place in that register after the colon prompt and press **Enter**. To check the new value of the BX register, you can use the Register command again:

```
-R
AX=ØØØØ BX=ØØ5A CX=ØØØØ DX=ØØØØ SP=FFEE BP=ØØØØ SI=ØØØØ DI=ØØØØ
DS=176F ES=176F SS=176F CS=176F IP=Ø1ØØ  NV UP EI PL NZ NA PO NC
176F:Ø1ØØ Ø43C          ADD   AL,3C
```

S (Search)

Searches for a value.

Format: S *range list*

Use this command to search the *range* for the *list* of bytes. The *list* may contain one or more bytes; separate each one with a space or comma. If the *list* contains more than one byte, only the address of the first byte is returned.

For example, if you type SCS:1ØØ 11Ø 41, DEBUG displays a response similar to this:

```
Ø4BA:Ø1Ø4
Ø4BA:Ø1ØD
```

T (Trace)

Traces a program.

Format: T [=*address*] [*value*]

Use this command to trace a program while it executes so you can see the contents of all registers and flags next to the decoded instruction. The optional =*address* parameter starts tracing at *address*. The optional *value* parameter sets the number of steps you want to trace. If you don't specify a *value*, one step is traced.

The T command uses the trap flag of the 8086/8088 microprocessor, which means you may also trace instructions stored in ROM.

For example, tracing through the following instruction leaves Ø83DH in the SP register:

```
AX=ØØØØ BX=ØØØØ CX=62BC DX=ØØØØ SP=FFFE BP=ØØØØ SI=ØØØØ DI=ØØØØ
DS=1787 ES=1787 SS=1787 CS=1787 IP=ØE3Ø  NV UP EI PL NZ NA PO NC
1787:ØE3Ø BC3DØ8        MOV   SP,Ø83D
-T

AX=ØØØØ BX=ØØØØ CX=62BC DX=ØØØØ SP=Ø83D BP=ØØØØ SI=ØØØØ DI=ØØØØ
DS=1787 ES=1787 SS=1787 CS=1787 IP=ØE33  NV UP EI PL NZ NA PO NC
1787:ØE33 B430        MOV   AH,3Ø
```

Notice that SP now holds Ø83DH. The Trace command traces through interrupt routines, subroutines, and loops. To trace over them, you use the Proceed command instead of Trace.

U (Unassemble)

Converts memory into assembly language mnemonics.

Format: U [*range*]

Use this command to convert the bytes in memory to the corresponding source statements, showing addresses, byte values, and mnemonics. You must be familiar with assembly language mnemonics to understand the output from this command.

For example, to unassemble a program from CS:Ø1ØØ to CS:Ø116, you can use the unassemble command this way:

```
-u1ØØ 116
1787:Ø1ØØ E92DOD          JMP    ØE3Ø
1787:Ø1Ø3 BADAØA          MOV    DX,ØADA
1787:Ø1Ø6 3DØ5ØØ          CMP    AX,ØØØ5
1787:Ø1Ø9 741B            JZ     Ø126
1787:Ø1ØB BABFØA          MOV    DX,ØABF
1787:Ø1ØE 3DØ2ØØ          CMP    AX,ØØØ2
1787:Ø111 7413            JZ     Ø126
1787:Ø113 BA85ØA          MOV    DX,ØA85
1787:Ø116 3DØ8ØØ          CMP    AX,ØØØ8
```

W (Write)

Writes a file.

Format: `W [address [drive record number]]`

Use this command to write the debugged file to a disk file. If you use the W command without parameters, BX:CX must already be set to the number of bytes to be written; the file begins writing from CS:100. If you use the W command with just an *address*, then the file begins writing at that address. If you use a G or T command, BX:CX must be reset before using the Write command without parameters. Note that if a file is loaded and modified, the name, length, and starting address are all set correctly to save the modified file (as long as the length has not changed).

You must name the file to be written either when loading DEBUG or with the N command. In both cases, DEBUG lists the filename in a file control block at CS:5C.

If you type the W command with parameters, the write begins from the memory *address* and to the *drive* you specify. DEBUG writes the file beginning at the logical record number specified by *record*, and continues to write the file until the number of sectors specified in the *number* parameter have been written.

WARNING
Writing to absolute sectors is extremely dangerous. This process bypasses the file management portion of MS-DOS.

XA (Allocate Expanded Memory)

Allocates a specified number of EMS (expanded memory) pages.

Format: XA *count*

The *count* parameter specifies the number of 16K pages you want to allocate. If the allocation is successful, DEBUG displays the number of the handle created.

To allocate three EMS pages, type the following and press **Enter**:

```
XA 3
```

If three pages of EMS memory are available, DEBUG displays a message similar to the following:

```
Handle created=0001
```

XD (Deallocate Expanded Memory)

Deallocates a specified expanded memory handle.

Format: XD *handle*

The parameter *handle* specifies the handle to be deallocated.

If the handle is valid, DEBUG displays a message confirming the deallocation.

To deallocate handle 0004, type the following and press **Enter**:

```
XD 0004
```

If the handle is deallocated successfully, DEBUG displays this message:

```
Handle 0004 deallocated
```

If the handle does not exist, DEBUG displays this message:

```
Handle not found
```

XM (Map Expanded Memory Pages)

Maps a specified logical page within a specified expanded memory handle into a specified physical page.

Format: XM *lpage ppage handle*

The *lpage* parameter specifies the page number within the pages allocated to *handle*.

The *ppage* parameter specifies the physical page number into which *lpage* is to be mapped.

The *handle* parameter specifies the EMS handle number.

If the operation is successful, DEBUG displays a confirming message.

To map logical page 3 of EMS handle 0005 to physical page 2, type the following and press **Enter**:

```
XM 3 2 5
```

If the operation is successful, DEBUG displays this message:

```
Logical page Ø3 mapped to physical page Ø2
```

XS (Get Expanded Memory Status)

Displays expanded memory status information.

Format: XS

DEBUG presents a display similar to the following:

```
Handle ØØØØ has ØØ24 pages allocated
Handle ØØØ1 has ØØ9Ø pages allocated
Physical page Ø9=Frame segment 1ØØØ
Physical page ØA=Frame segment 14ØØ
.
.
.
B6 of a total F6 EMS pages have been allocated
 2 of a total FF EMS handles have been allocated
```

Example

The following example shows how you can use DEBUG to load and alter a file and then re-save it on disk. This procedure is commonly called patching a file. This example makes use of a sample file, TESTBUG.COM, included on one of your MS-DOS disks. To try the examples in this section, copy DEBUG.COM and TESTBUG.COM onto a working disk. Do not write-protect this disk because the examples include the command to write the altered file to the disk.

Follow these steps to alter a file using DEBUG:

1. Start DEBUG by typing `DEBUG` and pressing **Enter**.
2. Name the file to be examined or altered with the Name command.
3. Load the file with the Load command.
4. Examine the file in memory with the Dump and Unassemble commands.
5. Alter the file with the Edit, Fill, and Assemble commands.
6. Test the program by running parts of it under the control of the Go command.
7. If necessary, rename the file with the Name command.
8. Put the file back on to the disk with the Write command.

Preparing to alter the file

Before altering the sample file, run it to see what it does. At the command prompt, type `TESTBUG` and press **Enter**. A message prints, the cursor moves down three lines, and the computer beeps.

Now start DEBUG. To load TESTBUG.COM at the same time as DEBUG, type `DEBUG TESTBUG.COM` and press **Enter**.

Examining the file

TESTBUG.COM is a short file, 50H bytes long. DEBUG always loads the file you specify starting at 100H in the current segment. To display the area of memory containing the file, type D 1ØØ L5Ø and press **Enter**. You see the following:

```
ØA63:Ø1ØØ  B9 2C ØØ BE 2Ø Ø1 8A 14-B4 Ø2 CD 21 46 E2 F7 CD 9,.>...4.M!FbwM
ØA63:Ø11Ø  2Ø ØØ ØØ ØØ ØØ ØØ ØØ ØØ-ØØ ØØ ØØ ØØ ØØ ØØ ØØ ØØ ................
ØA63:Ø12Ø  54 68 69 73 2Ø 69 73 2Ø-61 2Ø 74 65 73 74 2Ø 66 This is a test f
ØA63:Ø13Ø  69 6C 65 2Ø 66 6F 72 2Ø-75 73 65 2Ø 77 69 74 68 ile for use with
ØA63:Ø14Ø  2Ø 44 45 42 55 47 2E ØD-ØA ØA ØA Ø7 ØØ ØØ ØØ ØØ DEBUG.........
```

The first parameter tells DEBUG where to start the dump. The parameter following the L is the number of bytes to display. The first two lines of the display consist of the program to print the message and do not mean anything in this form. The message itself starts on the third line; you can read it in the block at the right of the screen.

The memory address (in these examples, 0A63) varies depending on where MS-DOS loads DEBUG into memory.

To display the program in a more readable form, use the Unassemble command. Type U 1ØØ 14F and press **Enter**. You see the program scroll by on your screen in the following format:

```
ØA63:Ø146 2E          CS:
ØA63:Ø147 ØDØAØA      OR      AX,ØAØA
ØA63:Ø14A ØAØ7        OR      AL,[BX]
ØA63:Ø14C ØØØØ        ADD     [BX+SI],AL
ØA63:Ø14E ØØØØ        ADD     [BX+SI],AL
```

This time the second parameter is the address where DEBUG stops unassembling. With most commands like this, you can use either method to tell DEBUG how much of the file to process. The program displays, using assembly language mnemonics. Where the program ends (after the eighth line), the unassembled codes stop making sense. To see just the working part of the program, type U 1ØØ L1Ø and press **Enter**.

The other thing you often need to display when you are using DEBUG is the condition of the registers and flags of the CPU. The Register command lets you display and alter the register contents. To see the contents of all the registers and flags, type R and press **Enter**. You see:

```
AX=ØØØØ   BX=ØØØØ CX=Ø256 DX=ØØØØ SP=FFFE BP=ØØØØ SI=ØØØØ DI=ØØØØ
DS=0A63   ES=ØA63 SS=ØA63 CS=ØA63 IP=Ø1ØØ  NV UP EI PL NZ NA PO NC
ØA63:Ø1ØØ B92CØØ          MOV           C,XØØ2C
```

The most important registers are CX and BX in the top row, and IP in the second row. When you save a file, CX and BX tell DEBUG how much to save. The IP register indicates where DEBUG looks for the next instruction to execute. The DS, ES, SS, and CS registers vary depending on where in memory DEBUG loads. The assembly language mnemonics on the last line are the instructions to which the IP register is currently pointing.

Running the program

The last instruction in the display produced by the U command starts at 10FH. Because the IP register currently points to the start of the program at 100H, you can run it using the Go command by typing G 1ØF and pressing **Enter**. After the message is printed, you see the registers again:

```
AX=ØØØØ   BX=ØØØØ CX=ØØØØ DX=ØØØ7 SP=FFFE BP=ØØØØ SI=Ø14C DI=ØØØØ
DS=ØA63   ES=ØA63 SS=ØA63 CS=ØA63 IP=Ø1ØØ  NV UP EI PL NZ NA PO NC
ØA63:Ø1ØF CD2Ø       INT         2Ø
```

Now you can see that DX contains the value 7, the character that made the beep, and SI contains 14C, the address of the character after the beep. Similarly, IP is now 10F, the address of the next instruction to be performed. To reset the IP register ready to run it again, type RIP and press **Enter**; then type 1ØØ and press **Enter**.

Altering the program

To alter the program to display a shorter message, followed by just a carriage return and a line feed, you need to change the program by putting in the new message, editing in the carriage return and line-feed codes, and changing the byte containing the length of the message. (If you look at the dump from the Dump command, this is located at address 101H.) To change the message to *This is a test message*, you must alter bytes starting at 12FH. To edit text in this way, put it in single quotes. Type E 12F 'message' and press **Enter**.

If you dump the program again with the D command, you see the altered message. The byte following the end of the new message is at location 137H. To put in the carriage return and line feed codes, type E 136 and press **Enter**. DEBUG responds with:

```
ØA63:Ø136 72.
```

The byte displayed is the current value stored; 72H is the code for an r. Type the new value, ØD (carriage return), and press the spacebar to display the next byte. Type ØA (line feed), then press **Enter**. The screen shows the following:

```
ØA63:Ø136 72.ØD 20.ØA
```

The total length of the message is now 18H, so you can delete the rest of the message by typing E 1Ø1 and pressing **Enter**. Type 18 and press **Enter**.

You can now try the program using the Go command. (Reset the IP register afterwards as described above.)

Another useful command for changing the program is the Fill command. Use this command to fill an area of memory with a particular byte or pattern of characters. The following two commands overwrite the new message with the letter A (character code 41H) and the pattern az. They also show the two ways of giving the range to fill. Press **Enter** after each command. Use the D command after each fill to examine the results.

```
-F 12Ø 135 41
-F 12Ø L16 'az'
```

Writing the file to disk

When you finish correcting the program, reset the IP register to the beginning of the program using the R command. Then set the CX register to tell DEBUG how many bytes to save. The program is still 50H bytes long, so type R CX and press **Enter**. You see CX ØØØØ. Type 5Ø and press **Enter**.

Use the R command alone to check that the registers are set correctly, including the BX register, which must contain 0.

If you want to give the altered program a new name, use the Name command. You can save the file by typing W and pressing **Enter**. You see the following message:

```
Writing ØØ5Ø bytes
```

To leave DEBUG, type Q and press **Enter**.

EDLIN

External

Purpose

Creates and edits text files for your system.

Format

`[d:][path]EDLIN [d:][path]filename [/B]`

Description

The EDLIN utility is a line editing program that comes complete with its own subcommands. You can use EDLIN to create, change, and display files, whether they are source programs or text files you create using a word processor.

EDLIN is useful only when you want to edit files that are in ASCII format. This is true of some word-processing and GW-BASIC files. EDLIN is of limited use with other types of files.

Use the /B switch when editing files that may have end-of-file characters in the text. The /B switch tells EDLIN to ignore Ctrl Z (1AH).

You can use EDLIN to perform the following functions:

- Create new source files and save them (you can even create a BASIC program with EDLIN)
- Update existing files and save both the updated and original files
- Delete, edit, insert, and display lines or characters
- Search for, delete, or replace text within one or more lines.

When you create or edit text in files, EDLIN divides the text into lines. Each line can contain as many as 253 characters. EDLIN displays line numbers during the editing process but does not actually record them in the file. EDLIN automatically renumbers lines in the file when you insert or delete text.

Starting EDLIN

To create a new file, type EDLIN followed by the name of the file you wish to create. The following message and prompt appear:

```
New file
*
```

Notice that the EDLIN prompt is an asterisk (*).

You can now type lines of text into the file. To begin entering text, use the I command to insert lines. (The I command is explained later in this chapter.)

To edit an existing file, type EDLIN and the name of the file you want to edit. EDLIN loads the file into memory and displays this message on your screen:

```
End of input file
*
```

You can then edit the file using the EDLIN commands.

If the file is too large to be loaded into memory, EDLIN loads lines of the file until memory is three-quarters full; then it displays the * prompt. You can then edit the portion of the file in memory.

To edit the remainder of the file, you must first save the edited lines. EDLIN then loads the unedited lines from disk into memory. Use the Write and Append commands for this procedure.

When you complete the editing session, you can save the original and the updated (new) file to disk using the End command. The original filename now includes a .BAK extension, and the new file has the filename and extension you originally specified with the EDLIN command. Do not erase any previous .BAK files until the end of the editing session or until you require extra disk space.

WARNING
Do not try to edit a file with an extension of .BAK because EDLIN assumes that any .BAK file is a backup file. If you find it necessary to edit such a file, rename the file with another extension (using the MS-DOS RENAME command), then start EDLIN and specify the new filename.

When editing a new file or an existing one with EDLIN, remember that you can use path names along with the filename. For example, typing `EDLIN \WORDPROC\PERSONAL\RESUME.TXT` lets you edit RESUME.TXT in the subdirectory PERSONAL.

Command Information

The following list provides a summary of the EDLIN commands.

Command	*Function*
line	Edits line number
A	Appends lines
C	Copies lines
D	Deletes lines
E	Ends editing; saves all changes
I	Inserts lines
L	Lists lines
M	Moves lines
P	Inserts page breaks
Q	Quits editing; does not save changes
R	Replaces lines
S	Searches text
T	Transfers text
W	Writes lines

Several EDLIN commands accept one or more options. The effect of an option varies, depending on the command you are using. The command options are shown on the following page.

line

This option indicates a line number that you type. You can enter a comma or a space to separate line numbers from other line numbers, other options, and from the command.

You can specify *line* in one of four ways:

- Type any whole number less than 65534. If you specify a number greater than the largest existing line number, then EDLIN assumes you mean the line after the last existing line number.
- Type a period (.) to indicate the current line number. The current line is the last line edited. An asterisk (*) appears on your screen between the line number and the first character to mark the current line.
- Type a pound sign (#) to indicate the line after the last line number in the file. This has the same effect as specifying a number larger than the last line number in the file.
- Press **Enter** without any line specifiers to direct EDLIN to use the default value for the command.

?

The question mark option directs EDLIN to ask if a correct string has been found when using the Search and Replace commands. Before continuing, EDLIN waits for you to press either **Y** or **Enter** for a Yes response, or any other key for a No response.

string

The string option represents text you want to find or replace, or the replacement text. Use the string option only with the Search and Replace commands. You must press **Ctrl Z** or **Enter** to end each string. Do not leave spaces between strings or between a string and its command letter, unless you want to include those spaces as part of the string.

Special Editing Features

There are several special features that enhance the EDLIN editing commands:

- You can reference line numbers relative to the current line. Use a minus sign (−) with a number to indicate lines before the current line. Use a plus sign (+) with a number to indicate lines after the current line. For example, the command `-1Ø,+1ØL` lists 10 lines before the current line, the current line, and 10 lines after the current line.

- You can issue multiple commands on one command line. When issuing a command to edit a single line using a line number, you must use a semicolon to separate commands on the line. Otherwise, one command may follow another without any special separators. In the case of a Search or Replace command, you can end the string by a **Ctrl Z** instead of **Enter**. For example, the command line `15;-5,+5L` edits line 15 and then displays lines 10 through 20 on the screen.

  To search for *This string* and then display 5 lines before and 5 lines after the line containing the matched string, type `STh is string` and press **Ctrl Z**; then type `-5,+5L`.

- You can type EDLIN commands with or without a space between the line number and command. For example, to delete line 6, the command `6D` is the same as `6 D`.

- Press **Ctrl V** while in the insert mode to insert control characters into text. **Ctrl V** indicates that the next uppercase letter you type is a control character. You can also use a control character in any of the Search and Replace string arguments. For example, to find the first occurrence of Ctrl Z, type `S`, press **Ctrl V**, and type `Z`.

  To replace all occurrences of Ctrl Z in a file with the End command, type `R`, press **Ctrl V**, type `Z`, press **Ctrl Z**, and type `End`.

  To insert a **Ctrl V** into the text, press **Ctrl V** and type `V`.

- Ctrl Z normally indicates end-of-file. If you use Ctrl Z characters elsewhere in a file, you must tell EDLIN that these do not mean end-of-file. When you start EDLIN using the optional /B switch, EDLIN ignores all Ctrl Z characters and shows the entire file.

EDLIN also makes use of the special editing keys. See Chapter 1 for a complete description of how to use these keys for editing.

EDLIN Commands

This section provides an alphabetical list of all EDLIN commands and examples of how to use them.

A (Append)

Adds data to a file.

Format: `[n]A`

Use this command to append or add a specific number of lines (*n*) from disk to the file you are editing in memory. EDLIN adds these lines at the end of the lines currently in memory.

This command is meaningful only if the file you are editing is too large to fit into memory.

To edit the remainder of the file that does not fit into memory, you must write the lines that you have already edited to disk. After this is done, you can load unedited lines from disk into memory with the Append command. Refer to the Write command in this section for information on how to write edited lines to disk.

If you do not specify the number of lines to append, EDLIN appends lines until available memory is three-quarters full. No action is taken if available memory is already three-quarters full. Once the last line of the file is read, you see the message `End of input file`.

Example: After you save edited lines to disk with the Write command, type `2ØA` at the * prompt to append the next 20 lines to memory.

C (Copy)

Copies lines of text.

Format: [*begline*][,*endline*,]*line*[,*count*]C

Use this command to copy a range of lines (*begline* to *endline*) to a specific *line* number. You can copy the lines as many times as you like by using the *count* option.

If you do not specify *count*, EDLIN copies the lines once. If you omit *begline* or *endline*, the current line is the default. After the copy, the file is renumbered automatically.

If the line numbers overlap, the message `Entry error` displays. For example, typing `3,2Ø,15C` produces this error message.

If you want to place text within other text, specify the line before the line where you want the copied text to appear. The copied text is inserted after the line you specify.

Examples: Assume the following file exists and is ready to edit:

```
1: This is a sample file
2: used to show copying lines.
3: See what happens when you use
4: the (C)opy command
5: to copy text in the file
```

Type `1,5,6C` to copy this entire block of text. The result is:

```
 1: This is a sample file
 2: used to show copying lines.
 3: See what happens when you use
 4: the (C)opy command
 5: to copy text in the file
 6: This is a sample file
 7: used to show copying lines.
 8: See what happens when you use
 9: the (C)opy command
10: to copy text in the file
```

Type `1,5,6,2C` to make two copies of the lines.

D (Delete)

Deletes lines of text.

Format: [*begline*][,*endline*]D

Use this command to delete a range of lines from a file.

If you omit *begline*, it defaults to the current line (the line with the asterisk). To delete *begline* only, omit *endline*. After the deletion, the line immediately following the deleted section becomes the current line with the same line number as the original *begline*.

Examples: Assume that the following file exists and is ready for you to edit:

```
 1: This is a sample file
 2: used to show dynamic line numbers
 3: See what happens when you use
 4: Delete and Insert
  .
  .
  .
25: (the D and I commands)
26: to edit the text
27:*in your file
```

Type 5,24D to delete lines 5 through 24. The result is the following:

```
1: This is a sample file
2: used to show dynamic line numbers
3: See what happens when you use
4: Delete and Insert
5: (the D and I commands)
6: to edit the text
7:*in your file
```

Type 6D to delete this single line. The result is the following:

```
1: This is a sample file
2: used to show dynamic line numbers
3: See what happens when you use
4: Delete and Insert
5: (the D and I commands)
6:*in your file
```

Next, to change the current line to 3, type 3 at the asterisk and then press **Enter**. Press **Enter** again at the EDLIN prompt.

Type ,6D to delete the lines from the current line, which is now 3, to line 6. The result is the following:

```
1: This is a sample file
2: used to show dynamic line numbers
3:*in your file
```

Notice that the lines are automatically renumbered.

E (End)

Exits EDLIN.

Format: E

Use this command to save the edited file on disk, rename the original input file to filename .BAK, and then exit EDLIN. If you originated this file during the editing session, EDLIN does not create a .BAK file.

The E command has no options. Therefore, you cannot indicate the drive where you want to save the file using the E command. You must select the drive where you want to save the file when you start the editing session. If you did not specify a drive, then EDLIN saves the file on the default drive. You can copy the file to a different drive using the MS-DOS COPY command.

You must be sure that the disk contains enough free space for the entire file. If the disk does not contain enough free space, the disk writing process aborts and parts of the edited file are lost.

After you execute the E command, the MS-DOS prompt is displayed. You can use the MS-DOS DIR command to check the status of the saved file.

I (Insert)

Inserts text.

Format: [*line*]I

Use this command to insert text immediately before the specified *line*. If you do not specify *line*, new lines are inserted before the current line. If *line* is any number larger than the last line number, or if you use a pound sign (#) to specify line, the inserted lines append to the end of the file. In this case, the last inserted line becomes the current line.

If you are creating a new file, you must enter the I command before you can type (insert) text. Text begins with line number 1 and successive line numbers appear automatically each time you press **Enter**.

EDLIN remains in insert mode until you press **Ctrl C**. After the insert is complete and you exit insert mode, the line immediately following the inserted lines becomes the current line. All line numbers following the inserted section are incremented by the number of inserted lines.

Examples: Assume that the following file exists and is ready for you to edit:

```
1: This is a sample file
2: used to show dynamic line numbers.
3: See what happens when you use
4: Delete and Insert
5: to exit text
6: and remember lines
7:*in your file
```

Type 5I to insert text before a specific line (in this case, line 5) that is not the current line.

EDLIN displays a blank line with the number 5 and an asterisk, and waits for you to insert text.

Now, type (the D and I commands) as the text for line 5. When line number 6 appears, press **Ctrl C** to end the insertion.

Type L to list the entire file. You see the following:

```
1: This is a sample file
2: used to show dynamic line numbers.
3: See what happens when you use
4: Delete and Insert
5: (the D and I commands)
6:*to edit text
7: and renumber lines
8: in your file
```

Type I only to insert text immediately before the current line.

To exit insert mode, press **Ctrl C** once again.

You can type #I or 2Ø I to append new lines to the end of the sample file.

Both formats tell EDLIN to start appending after the last line. In the second case, 20 is larger than the last line number that exists in the file, so EDLIN starts appending after the last line number.

L (List)

Displays text.

Format: [*begline*][,*endline*]L

Use this command to list, or display, a range of lines including the two lines you specify. If you omit either *begline* or *endline*, EDLIN uses default values.

If you omit *begline*, the display starts 11 lines before the current line and ends with *endline*. You must enter the beginning comma to indicate the default option:

,*endline*L

If you omit *endline* the screen displays 23 lines starting with *begline*:

*begline*L

If you omit both parameters, 23 lines are displayed. These include the 11 lines before the current line, the current line, and the 11 lines after. If there are fewer than 11 lines before the current line, more than 11 lines after the current line are displayed for a total of 23.

Examples: Assume that the following file exists and is ready for you to edit:

```
 1: This is a sample file
 2: used to show dynamic line numbers
 3: See what happens when you use
 4: Delete and Insert
 5: (the D and I commands)
  .
  .
  .
15:*The current line contains an asterisk.
  .
  .
  .
26: to edit text
27: in your file.
```

Type 2,5L to list a range of lines without reference to the current line. The result is the following:

```
 2: used to show dynamic line numbers
 3: See what happens when you use
 4: Delete and Insert
 5: (the D and I commands)
```

Type ,26L to list a range of lines beginning with the current line. The result is the following:

```
15:*The current line contains an asterisk.
  .
  .
  .
26: to edit text
```

Type L to list a range of text centered around the current line.

M (Move)

Moves text to a new location.

Format: [*begline*][,*endline*,]*line*M

Use this command to move a block of text to another location in the file. The lines renumber according to the direction of the move. If the line numbers overlap, EDLIN displays the message `Entry error`.

Examples: Type ,+25,1ØØM to move text from the current line plus 25 lines to line 100. Type 2Ø,3Ø,1ØØM to move lines 20 through 30 to line 100.

P (Page)

Pages through text.

Format: [*begline*][,*endline*]P

Use this command to page through a file 23 lines at a time. If you omit *begline*, that number defaults to the current line plus 1. If you omit *endline*, the 23 lines following *begline* display. The new current line becomes the last line displayed.

Examples: Type P to display the 23 lines following the current line.

Type 5P to display the 23 lines following line 5.

Q (Quit)

Ends EDLIN.

Format: `Q`

Use this command to quit the editing session without saving any editing changes you have made. You exit EDLIN, and the default drive prompt appears.

Because using this command has drastic effects, EDLIN prompts you to make sure that you don't want to save the changes. Press **Y** if you want to quit the editing session. Press **N** or any other character (except Y) if you want to continue the editing session.

WARNING
When you start this command, EDLIN erases the .BAK file associated with the specified filename. If you reply **Y** to the abort message, the previous .BAK file no longer exists.

Remember to use the E command to make permanent changes to the file.

Examples: To quit the editing session, type `Q` and press **Enter**.

You see the message:

```
Abort edit (Y/N)?
```

Press **Y** and then press **Enter** to abort the file.

R (Replace)

Replaces text.

Format: `[`*`begline`*`][,`*`endline`*`][?]R`*`string1`*`Ctrl Z`*`string2`*

Use this command to replace all occurrences of a string of text in the specified range with a different string of text or blanks. After you type `R`*`string1`*, press **Ctrl Z** and then type *`string2`*.

EDLIN finds each occurrence of *string1* and replaces it with *string2*. EDLIN displays each line in which a replacement occurs. If a line

contains two or more replacements of *string1* with *string2*, the line is displayed once for each occurrence. When EDLIN replaces all occurrences of *string1* with *string2*, the R command terminates and the asterisk prompt is displayed.

When you enter a second string as a replacement, you must separate *string1* from *string2* with a **Ctrl Z**. You must also end *string2* with a **Ctrl Z** and **Enter** combination or with a single **Enter**.

If you omit *string1*, Replace takes the old *string1* as its value. If there isn't a prior *string1*, the replacement process stops immediately. If you omit *string2*, you can press **Enter** to end *string1*.

If you omit *begline* in the range argument, then *begline* defaults to the line after the current line. If you omit *endline*, # is the default. This is the same as *begline*,#. Remember, # indicates the line after the last line of the file.

If you end *string1* with **Ctrl Z** and there is no *string2*, an empty string becomes the new replace string. For example, type `Rstring2` and press **Ctrl Z** and **Enter** to delete occurrences of *string1*. But if you type `Rstring1` and press **Enter** and then type R and press **Enter**, EDLIN replaces *string1* with the old *string2*, and the old *string1* with the old *string2*, respectively. Note that old refers to a string previously specified in a Search and Replace command.

If you use the question mark (?) option, the Replace command stops at each line with a string that matches *string1*, displays the line with *string2* in place, and then displays `O.K.?` as the prompt.

If you press **Y** or **Enter**, *string2* replaces *string1*, and EDLIN finds the next occurrence of *string1*. The `O.K.?` prompt displays again. This process continues until the end of the range or the end of the file. After finding the last occurrence of *string1*, EDLIN displays the asterisk prompt.

If you press any key other than **Y** or **Enter** after the `O.K.?` prompt, *string1* does not change and Replace goes to the next occurrence of *string1* (even if *string1* occurs more than once in a line). EDLIN replaces each occurrence of *string1* individually and the `O.K.?` prompt displays before each replacement. In this way, you can select each *string1* replacement, and prevent unwanted substitutions.

Examples: Assume that the following file exists and is ready for you to edit:

```
 1:*This is a sample file
 2: used to show dynamic line numbers
 3: See what happens when you use
 4: Delete and Insert
 5: (the D and I commands)
 6: to edit text
 7: in your file.
 8: The insert command can place new lines
 9: in the file; there's no problem
1Ø: because the line numbers are dynamic;
11: they'll go all the way to 65533.
```

To replace all occurrences of *and* with *or* in a specified range, type `2,12Rand` and press **Ctrl Z**. Then type `or` and press **Enter**. The result is the following:

```
 4: Delete or Insert
 5: (the D or I commands)
 5:*(the D or I commors)
 8: the insert commor can place new lines
```

To avoid unwanted substitutions and to confirm each replacement, you can use the same original file with a slightly different command.

In the next example, only certain occurrences of *and* with *or* can be made. To force the O.K.? prompt, type `2,12?Rand` and press **Ctrl Z**. Then type `or` and press **Enter**. The result is:

```
 4: Delete or Insert
O.K.? Y
 5: (The D or I commands)
O.K.? Y
 5: (The D or I commors)
O.K.? N
 8: The insert commor can place new lines
O.K.? N
*
```

Now, type the List command (L) to see the result of all these changes. You should see the unchanged lines and the new ones.

S (Search)

Searches for a string of text.

Format: [*begline*][,*endline*][?]S*string*

Use this command to search the specified range of lines for the specified string of text. You must press **Enter** to end the string.

The first line that matches *string* displays and becomes the current line. If you do not specify the question mark option, the Search command stops when it finds a match. If it cannot find a match, you see the message `Not found`.

If you include the question mark option (?) in the command, EDLIN displays the first line with a matching string and the `O.K.?` prompt.

If you press either **Y** or **Enter**, the line becomes the current line and the search terminates. If you press any other key, the search continues.

If you omit *begline*, it defaults to the line after the current line. If you omit *endline*, it defaults to # (the line after last line of the file), which is the same as *begline*,#S*string*. If you omit *string*, Search takes the old string if there is one. (Note that old refers to a string specified in a previous Search or Replace command.) If an old string does not exist, the command ends.

Examples: Assume the following file exists and is ready for you to edit:

```
 1: This is a sample file
 2: used to show dynamic line numbers
 3: See what happens when you use
 4: Delete and Insert
 5: (the D and I commands)
 6: to edit text
 7: in your file.
 8: The insert command can place new lines
 9: in the file; there's no problem
10: because the line numbers are dynamic;
11: they'll go all the way to 65533.
```

To search for the first occurrence of the string *and* between lines 2 and 12 inclusive, type `2,12Sand` and press **Enter**. The following line appears:

```
4: Delete and Insert
```

To find the next *and* type `,12Sand` and press **Enter**. The search then continues from the line after the current line (line 4), because you did not specify the first line. The result is the following:

```
5: (the D and I commands)
```

To search for several occurrences of the string *and*, type `1?Sand` and press **Enter**. The result is:

```
4: Delete and Insert

O.K.?
```

If you press any key (except **Y** or **Enter**), the search continues, so press **N** here. You see the following:

```
5: (the D and I commands)
O.K.?
```

Now press **Y** to terminate the search. You see the EDLIN * prompt.

To search for string XYZ without verification (O.K.?), type SXYZ with no options. EDLIN reports a match and continues to search for the same string. *String* defaults to any string previously specified in a Replace or Search command.

T (Transfer)

Moves text into the current file.

Format: `[`*`line`*`]T`*`filename`*

Use this command to insert, or merge, the contents of *filename* into the file you are currently editing at the *line* you specify. If you omit *line*, EDLIN uses the current line.

This command is useful if you want to place the contents of a file into another file or into the text you are typing. The text inserts at the line number you specify in *line* and the lines automatically renumber.

Examples: Type `TNEW.TXT` to insert text from a file called NEW.TXT at the current line.

Type `1ØØTINS.DOC` to insert the text from INS.DOC starting from line 100.

W (Write)

Writes text to disk.

Format: `[n]W`

Use this command to write a specific number of lines (*n*) to disk from the lines that you are editing in memory. EDLIN writes the lines to the disk beginning with line number 1.

This command is meaningful only if the file you are editing is too large to fit into memory. When you start EDLIN to edit a file, EDLIN reads lines from the file until memory is three-quarters full. To edit the remainder of your file, you must write edited lines in memory to disk. Then, use the Append command to load additional unedited lines from disk into memory.

If you do not specify *n*, EDLIN writes lines until one-quarter of memory is available. If available memory is already greater than one-quarter, no action is taken. All lines renumber, so that the first remaining line becomes line number 1.

Example: Type `1ØØW` to save the first 100 lines of text to disk.

EXE2BIN *External*

Purpose

Converts .EXE files to .BIN files.

Format

```
[d:][path] EXE2BIN [d:][path]filename
                   [d:][path][filename]
```

Description

Use this command only if you want to convert your executable (.EXE) files to binary format. This conversion saves disk space and provides faster program loading.

The first file you name is the input file. If you do not specify an extension, it defaults to .EXE.

If you do not specify a drive, this command uses the drive of the input file. If you do not specify an output filename, the command uses the input filename. If you do not specify a filename extension in the output filename, the new file takes the extension .BIN.

To execute the new .BIN file, you need to rename it with an extension of .COM.

The input file must be in a valid .EXE format as produced by the LINK command. The resident, or actual code, and data part of the file must be less than 64K. There must be no STACK segment.

Two kinds of conversions are possible, depending on whether or not the .EXE file contains a specified initial Code Segment and Instruction Pointer (CS:IP).

- If the .EXE file does not contain a specified CS:IP, a pure binary conversion takes place. If segment fixes are necessary (that is, the program contains instructions requiring segment relocation), prompts appear asking for the fixup value. This value is the absolute segment where the program is to load. The resulting program is usable only when loaded at the absolute memory address specified by a user application. Otherwise, the command processor is not capable of properly loading the program.
- If you specify CS:IP as 0000:100H, this command runs the file as a .COM file with the location pointer set at 100H by the assembler statement ORG; the first 100H bytes of the file are deleted. You cannot provide segment fixups because .COM files must be segment relocatable. Once the conversion is complete, you may rename the resulting file with a .COM extension. Then, the command processor can load and execute the program in the same way as the .COM programs supplied on your MS-DOS system diskettes.

LINK

External

Purpose

Combines object code program modules.

Format

```
[d:][path]LINK [filename[,filename...]
               [/switch...]]
```

or

```
[d:][path]LINK @[d:][path]filename
```

Description

The MS-DOS LINK utility is a program that does the following:

- Combines separately produced object modules into one relocatable load module, which is a program you can run
- Searches library files for definitions of unresolved external references
- Resolves external cross-references
- Produces a listing that shows both the resolution of external references and error messages.

When you write a program, you write it in source code, either in a high-level programming language such as Pascal or in assembly language. If you use a high-level language, your source code passes through a compiler. If you use assembly language, the source code passes through an assembler. Both the compiler and the assembler produce object modules.

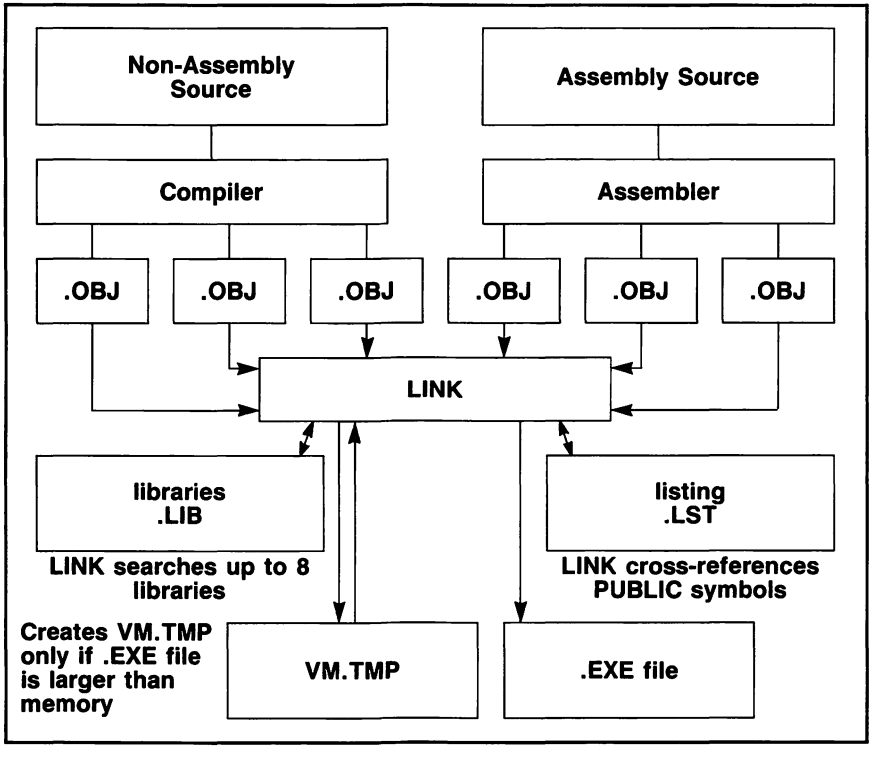
The object modules must then pass through a linking process to produce a language that the computer can understand directly. Machine language is the form your computer requires in order to run programs.

You may wish to link, or combine, several programs and run them together. LINK enables you to do this. LINK combines several object modules into one relocatable load module, which is then given the .EXE extension.

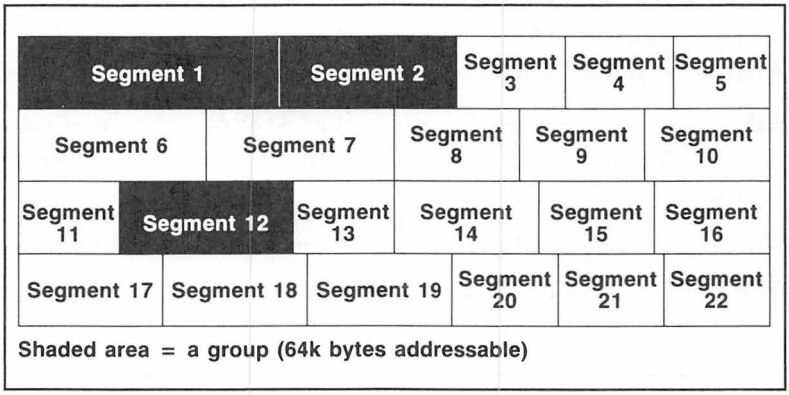
Your programs may contain symbols (called external references), which are defined in other object modules. As it combines modules, LINK makes sure that all external references between object modules are defined. LINK can search several library files for definitions of any external references that are not defined in the object modules. LINK produces a List file that shows resolved external references and also displays any error messages.

LINK uses available memory as much as possible. When no memory is available, LINK creates VM.TMP, which is a temporary disk file on the default drive.

The following figure illustrates the various parts of the LINK operation:



If you are writing and compiling programs in assembly language, you need to understand how this language divides memory into segments, classes, and groups. The following figure illustrates this division of memory.



In this illustration, the divisions may be:

	Segment Name	*Segment Class Name*
Segment 1	PROG.1	CODE
Segment 2	PROG.2	CODE
Segment 12	PROG.3	DATA

Segments 1, 2, and 12 have different segment names but may or may not have the same segment class name. Segments 1, 2, and 12 form a group.

Each segment has a name and a class name. LINK loads all segments into memory by class name from the first segment it encounters to the last. All segments with the same class load into memory contiguously.

A group is a collection of segments that fit within a 64KB area of memory. The segments do not need to be contiguous to form a group. The address of any group is the lowest address of the first segment in the group. LINK analyzes the groups and references the segments by their addresses in memory. A program may consist of one or more groups.

When you write assembly language programs, you can assign the group and class names in your program. In high-level languages, such as BASIC, COBOL, FORTRAN, and Pascal, the compiler names groups and classes automatically.

Files Used By LINK

You can direct LINK to use general files. You can do the following:

- Work with one or more input file
- Produce one or two output files
- Create a temporary disk file
- Search up to eight library files.

You use the standard specification when specifying files in LINK. Be sure to include the drive and path if necessary.

If you don't specify a file extension for the input, or object file, LINK recognizes the following default extensions:

.OBJ	Object
.LIB	Library

LINK appends the following default extensions to the Run and List output files:

.EXE	Run
.MAP	List

You cannot change these extensions.

LINK uses available memory first. If your files create an output file that exceeds available memory, LINK creates a temporary file. It names this file VM.TMP and puts it on the disk in the default drive. If LINK creates VM.TMP, this message appears:

```
VM.TMP has been created.
Do not change disk in drive, d:
```

Once this message appears, you must not remove the disk from the default drive until the link session ends. If you do, the operation of LINK is unpredictable and LINK may display the following error message:

```
Unexpected end of file on VM.TMP
```

During the link process, LINK prompts you for the name of a Run file:

```
Run File:
```

LINK writes the contents of VM.TMP to this file. VM.TMP is a working file and LINK deletes it at the end of the linking session. If you do not care to save the contents of VM.TMP, do not enter a filename for the `Run File` prompt.

WARNING
Do not use VM.TMP as a filename for any file. If you do and LINK requires a temporary file, LINK deletes any existing VM.TMP file to create a new file.

Starting LINK

To start LINK, you must enter the LINK command and its parameters. You do this in one of three ways:

- You can type LINK only and then enter the parameters in response to individual prompts. Type `LINK` and press **Enter**.
- You can type the LINK command and all its parameters on the command line. Type `LINK` *`filenames`* `[/switches]` and press **Enter**.

  You can also further define this format as:

  ```
  LINK object-list,runfile,listfile,
       lib-list[/switch...]
  ```

  In this format, *object-list* is a list of object modules, which you separate with plus signs. *runfile* is the name of the file to receive the executable output. *listfile* is the name of the file to receive

the listing. *lib-list* is a list of library modules you want to search. */switch* refers to optional switches, which you can place following any of the response entries (just before any of the commas or after the lib-list, as shown).

To select the default for a parameter, simply type a second comma with no spaces between the two commas.

This format of the LINK command can easily fill up the command line. You can press **Ctrl J** to move the cursor to the beginning of a new line without terminating the line. After you type all parameters, press **Enter**.

- You can create a response file that contains all the necessary parameters. Then type LINK and the name of the response file on the command line. Type `LINK @filename` and press **Enter**.

When specifying LINK commands, you can use the plus sign (+), the semicolon (;), and **Ctrl C** to control the link process.

- You can use the plus sign (+) or blanks to separate filenames. You can also extend the response to the Object Modules and Libraries prompts using the plus sign. With a large number of responses, type a plus sign and press **Enter** at the end of the line. LINK repeats either the Object Modules or Libraries prompt and you can continue to type responses. Once you enter all the modules to link and libraries to search, be sure to end the response line with a module name (not with a plus sign) followed by **Enter**.
- Use a semicolon (;) to select default responses.
- Use **Ctrl C** at any time to abort the link session. If you type an erroneous response, such as an incorrect filename, you must press **Ctrl C** to exit and then restart LINK. If you make an error before you press **Enter**, you can backspace to delete the erroneous characters on that line only.

After you enter the last parameter and press **Enter**, LINK begins the linking process. When complete, LINK exits to the operating system. When you see the command prompt, you know that the LINK process was successful. If the link session is unsuccessful, an appropriate error message appears.

LINK Parameters

This section describes the parameters LINK requires for operation.

Object module

The first parameter must be the name of the object module. The prompt you see is the following:

```
Object Modules [.OBJ]:
```

Type a list of the object modules you want to link. LINK assumes that the filename extension is .OBJ. If one of your object modules has any other filename extension, you must enter the extension.

Remember that LINK loads segments into classes as it encounters them. Therefore, enter the object modules in the order in which you want LINK to read them.

Separate the object modules by blank spaces or plus signs (+). If a plus sign is the last character you type, the prompt reappears. There is no default; you must enter a response.

Run file

The next parameter is the name of the Run File. The prompt you see is:

```
Run File [Object-filename.EXE]:
```

Type a filename if you want to create a file for storing the executable object code that results from the link session. All Run files receive the filename extension .EXE, even if you specify an extension other than .EXE.

If you don't type a response to the Run File prompt, LINK uses the first filename typed in response to the Object Modules prompt as the Run filename.

List file

The third parameter is the name of the List file. The prompt you see is:

```
List File [NUL.MAP]:
```

This file contains an entry for each segment in the input (object) modules. Each entry shows the addressing in the Run file.

If you don't enter a List filename, LINK does not create a map file.

Libraries

The last parameter is the name of the library files you want LINK to search for external references. The prompt you see is:

```
Libraries [.LIB]:
```

You can enter up to eight library filenames. You must create the library files with a library utility. LINK assumes that the filename extension is .LIB for library files. If you press **Enter** only, LINK searches all files with the .LIB extension.

Separate the library filenames with a blank space or a plus sign (+). If a plus sign is the last character you type, the prompt reappears.

LINK searches your library files in the order you list them to resolve external references. When it finds the module that defines the external symbol, LINK processes that module as another object module.

If LINK cannot find a library file on the disk in the specified disk drive, you see a message instructing you to type a new drive letter:

```
Cannot find library <library-name>
Enter new file spec:
```

LINK Switches

You can specify switches to control various LINK functions. You type the switches after each parameter, regardless of how you enter the parameters. If you type more than one switch at the end of a parameter, you must precede each switch with a forward slash (/).

You can abbreviate all the switches. The only restriction is that an abbreviation must be sequential from the first letter through the last letter you type; you cannot have gaps or transpositions. For example, legal and illegal abbreviations for the DSALLOCATE switch include:

Legal	*Illegal*
/D	/DSL
/DS	/DAL
/DSA	/DLC
/DSALLOCA	/DSALLOCT

The remainder of this section describes the switches you can use with LINK.

/CPARMAXALLOC:*n*

This switch sets the maximum number of 16-byte paragraphs that will be allocated for this program when it is loaded into memory by MS-DOS. (*n* can be any number from 1 to 65535.)

LINK normally allocates 65535 paragraphs, which represents all addressable memory. Use this switch to free additional space in memory for other programs.

/DOSSEG

Use this switch to tell LINK to arrange all segments in the executable file according to these rules:

- All segments with the class name CODE are placed at the beginning of the executable file.
- All other segments that do not belong to the group DGROUP are placed immediately after the CODE segments.
- All segments that belong to the group DGROUP are placed at the end of the file.

/DSALLOCATE

Use this switch to tell LINK to load all data at the high end of the data segment. Otherwise, LINK loads all data at the low end of the data segment. At runtime, the DS pointer is set to the lowest possible address in order to use the entire DS segment. If you use the /DSALLOCATE switch in combination with a low default load (that is, the /HIGH switch is not used) permits the user application to dynamically allocate any available memory below the area specifically allocated with DGroup, yet remain addressable by the same DS pointer. Pascal and FORTRAN programs need this dynamic allocation.

Note
Your application program may dynamically allocate up to 64KB (or the actual amount of memory available), less the amount allocated with DGroup.

/EXEPACK

Use this switch to pack executable files. LINK removes sequences of repeated bytes (typically NULs) and optimizes the load-time relocation table before creating the executable file.

Packed executable files may be smaller and load faster than other files. Programs that have a large number of load-time relocations (about 500 or more) and long streams of repeated characters are usually shorter if packed. Other programs may not be shorter. You cannot use the Microsoft Symbolic Debug Utility (SYMDEB) with packed files.

/HELP

Use this switch to view a list of the available options on your screen. Do not include a filename when using this switch.

/HIGH

This switch places the Run file as high as possible in memory. Otherwise, LINK places the Run file as low as possible.

> **Note**
> Do not use the /HIGH switch with Pascal or FORTRAN programs.

/LINENUMBERS

This switch includes line numbers and addresses to identify the source statements from the input modules in the List file. Otherwise, the List file does not include line numbers.

> **Note**
> Not all compilers produce object modules that contain line number information. In these cases, LINK cannot include line numbers.

/MAP

This switch lists all global symbols defined in the input modules. If /MAP is not given, LINK only lists errors (including undefined globals).

LINK lists the symbols alphabetically at the end of the List file. For each symbol, LINK lists its value and its segment:offset location in the Run file.

/NODEFAULT

This switch (which is short for NODEFAULT LIBRARY SEARCH) prohibits a search of the default libraries in the object modules. For example, if you are linking object modules in Pascal, and specify the /NODEFAULT switch, LINK does not automatically search the library named PASCAL.LIB to resolve external references.

/NOGROUP

Use this switch (which is short for NO GROUP ASSOCIATION) to ignore group associations when assigning addresses to data and code items.

> **Note**
> This switch exists strictly for compatibility with older versions of FORTRAN and Pascal (Microsoft versions 3.13 or earlier or any IBM version prior to 2.0). Only use this option to link with object files or libraries produced by these compilers.

/NOIGNORE

This switch (which is short for NO IGNORE CASE) tells LINK to recognize the difference between upper- and lowercase characters. With this switch, LINK treats TWO, two, and Two as different symbols.

Typically, you use this switch with object files created by high-level language compilers.

/OVERLAYINTERRUPT:*n*

Use this switch to set the interrupt number of the overlay loading routine to *n* (*n* can be a value from 0 to 255). This overrides the normal overlay interrupt number of 03FH.

Use this switch only if you are linking with a run-time module from a language compiler that supports overlays. Do not use interrupt numbers that conflict with the standard MS-DOS interrupts.

/PAUSE

LINK pauses during the session when it encounters this switch. Normally, LINK performs the linking session from beginning to end without stopping. This switch allows you to change disks before LINK outputs the Run (.EXE) file.

When LINK encounters the /PAUSE switch, you see the message:

```
About to generate .EXE file
Change disks <hit any key>
```

LINK resumes processing when you press any key.

Note
Do not remove the disk to receive the List file, or the disk containing the VM.TMP file, if one was created.

/SEGMENTS:*n*

Use this switch to process a maximum of *n* segments per program (*n* can be any number from 1 to 1024). If LINK encounters more than *n* segments, you see an error message and LINK stops. This switch overrides the default limit of 128 segments.

/STACK

This switch specifies the size of stack to be created. The format is:

```
/STACK:<number>
```

where *number* represents any positive numeric value (in hexadecimal) up to 65536 bytes. If you type a value from 1 to 511, LINK uses 512.

If you do not specify the /STACK switch, LINK calculates the necessary stack size automatically. All compilers and assemblers should provide information in the object modules to allow the LINK program to compute the required stack size.

At least one object module must contain a stack allocation statement. Otherwise, LINK displays the message:

```
WARNING: NO STACK SEGMENT
```

LINK Processing

LINK creates an executable file by concatenating a program's code and data segments according to the instructions in the original source files. These concatenated segments form an executable image that is copied directly into memory when you run the program. The order in which LINK copies segments to the executable file defines the order in which the executable file is loaded into memory.

You can tell LINK how to link a program's segments using directives that define group associations, classes, and align and combine types. These directives define the order and relative starting addresses of all segments in a program. These directives, which are defined in your object modules, work in addition to any LINK switches you provide on the LINK command line.

Aligning segments

LINK uses a segment's align type to set the starting address for the segment. The align types are byte, word, paragraph, and page, which represent address boundaries that are multiples of 1, 2, 16, and 256 respectively. The default align type is paragraph.

When LINK encounters a segment, it checks the align type before copying the segment to the executable file. If the align type is word, paragraph, or page, LINK checks the executable image to see if the last byte copied ends at an appropriate boundary. If it does not, LINK pads the image with extra NUL bytes.

LINK computes a starting address for each segment in a program. The starting address is based on a segment's align type and on the size of the segments already copied to the executable file. The address consists of an offset and a frame number, which specifies the address of the first paragraph in memory that contains one or more bytes of the segment.

A frame number is always a multiple of 16 (a paragraph address), and the offset is the number of bytes from the start of the paragraph to the first byte in the segment. For byte and word align types, the offset may be nonzero; for paragraph and page align types, the offset is always zero. The frame number of a segment is the first five hexadecimal digits of the start address specified for the segments.

LINK copies segments to the executable file in the same order that it encounters them in the object files. LINK maintains this order throughout the program unless it encounters two or more segments with the same class name. Segments with identical class names belong to the same class type and are ordered contiguously.

Combining segments

LINK uses combine types to determine whether or not to combine two or more segments with the same name. Combine types are public, stack, common, memory, at, and private.

LINK automatically combines public combine types with any other segments that have the same name and same class. When combining segments, LINK makes sure that the segments are contiguous and that all addresses in the segments can be accessed using an offset from the same frame address. The result is the same as if the segments were defined as a whole in the source file.

LINK preserves each segment's align type. Even though the segments are combined into one large segment, the code and data in the segments retain their original align type. If the combined segments exceed 64KB, LINK displays an error message.

LINK carries out the same combine operations for stack combine types as for public segments. The only difference for stack combine types is that LINK copies an initial stack-pointer value to the executable file. This stack-pointer value is the offset to the end of the first stack segment (or combined stack segment) that LINK encounters.

If you use the stack type for stack segments, you do not need to give instructions to load the segment into the SS register.

LINK combines common combine types automatically with any other segments with the same name and same class. When combining common segments, LINK places the start of each segment at the same address, creating a series of overlapping segments. The result is a single segment no larger than the largest of the combined segments.

LINK treats segments with memory combine types exactly like segments with public combine types. This combine type is available for compatibility with linkers that support a separate combine type for memory segments.

A segment has a private combine type only if no explicit combine type is defined for it in the source file. LINK does not combine private segments.

Processing groups

Groups permit noncontiguous segments that do not belong to the same class to be addressable relative to the same frame address. When LINK encounters a group, it adjusts all memory references to items in the group so that they are relative to the same frame address.

Segments in a group need not be contiguous, belong to the same class, or have the same combine type. The only requirement is that all segments in a group fit within 64KB of memory. LINK may encounter a fixup-overflow error if all segments in a group do not fit within 64KB of memory.

Groups do not affect the order in which segments are loaded. For segments to be contiguous, you must use class names and enter object files in the correct order. LINK may even place segments that do not belong to a group in the same 64KB of memory.

LINK fixups

Once LINK knows the starting address of each segment in a program and establishes all segment combinations and groups, it can fix up any unresolved references to labels and variables. To fix up unresolved references, LINK computes an appropriate offset and segment address and replaces the temporary values (generated by the assembler) with the new values.

LINK processes fixups for four different references: short, near self-relative, near segment-relative, and long.

The size of the value to be computed for the fixup depends on the type of reference. If LINK discovers an error in the anticipated size of a reference, it displays a fixup-overflow message. This type of error can occur if a program attempts to reach an instruction in a segment that has a different frame address or if the segments in a group do not fit within a single 64KB block of memory.

A short reference occurs in JMP instructions that attempt to pass control to labeled instructions in the same segment or group. The target instruction must be no more than 128 bytes from the point of reference. LINK computes a signed, 8-bit number for the JMP instruction and displays an error message if the target instruction belongs to a different segment or group (has a different frame address), or if the target is more than 128 bytes from the point of reference.

A near self-relative reference occurs in instructions that access data relative to the same segment or group. LINK computes a 16-bit offset for this reference and displays an error message if the data are not in the same segment or group.

A near segment-relative reference occurs in instructions that attempt to access data in a specified segment or group or data that is relative to a specified segment register. LINK computes a 16-bit offset for this reference and displays an error message if either of the following conditions exists:

- The offset of the target within the specified frame is greater than 64KB or less than 0.
- The beginning of the frame of the target is not addressable.

A long reference occurs in CALL instructions that attempt to access an instruction in another segment or group. LINK computes a 16-bit frame address and a 16-bit offset for this reference and displays an error message if either of the following conditions exists:

- The computed offset is greater than 64KB or less than 0.
- The beginning of the frame of the target is not addressable.

For a complete description of the LINK process, see the Microsoft Macro Assembler Reference Manual.

Examples

When you type LINK only on the command line, you can respond to the prompts as follows:

Line Prompt	*Your Response*
`Object Modules [.OBJ]:`	`PGM1 TEXT TABLE +` and press **Enter**
`Object Modules [.OBJ]:`	`PGM2 TABLE2` and press **Enter**
`Run File [PGM1.EXE]:`	`PGM3` and press **Enter**
`List File [NUL.MAP]:`	`PGM3 /MAP` and press **Enter**
`Libraries [.LIB]:`	`LIB2` and press **Enter**

When you enter LINK and all its parameters on one line, you might type:

```
LINK TEXT+TABLE/P/M,,PGMLIST,COBLIB.LIB
```

This command line causes LINK to load the object modules, TEXT.OBJ and TABLE.OBJ. LINK then pauses as a result of the /P switch. When you press any key, the object modules link and the /MAP switch produces a global symbol map. The Run file defaults to TEXT.EXE, PGMLIST.MAP becomes the name of the List file, and LINK searches the library file COBLIB.LIB.

To use a response file with the LINK command, you must first create and save the file to disk. Your response file may look like this:

```
TEXT TABLE PARAM
/PAUSE/MAP
PGMLIST
COBLIB.LIB
```

This response file tells LINK to load the three object modules named TEXT, TABLE, and PARAM. LINK pauses before producing a global symbol map to permit you to swap disks. When you press any key, the output filenames become TEXT.EXE and PMGLIST.MAP. Then LINK searches the library file COBLIB.LIB and uses the default settings for the switches.

This example shows you the type of information that appears on your screen during a LINK session. In response to the MS-DOS prompt, type `LINK`.

Enter your responses when you see these line prompts:

Line Prompt	*Your Response*
`Object Modules [.OBJ]:`	`IØ SYSINIT`
`Run File [IO.EXE]:`	**Enter**
`List File [NUL.MAP]:`	`IØ /MAP`
`Libraries [.LIB]:`	**Enter**

By specifying /MAP, you get both an alphabetical listing and a chronological listing of global symbols.

Once LINK locates all libraries, it displays a map of segments in order of appearance in the load module. The list might look like this:

```
Start     Stop        Length    Name
ØØØØØH    ØØ9ECH      Ø9EDH     CODE
ØØ9FØH    Ø1166H      Ø777H     SYSINITSEG
```

The information in the Start and Stop columns shows the 20-bit hex address of each segment relative to location zero. Location zero is the beginning of the load module.

Because you used the /MAP switch, LINK displays the global symbols by name and value. For example:

```
ADDRESS                 PUBLICS BY NAME

ØØ9F:ØØ12               BUFFERS
ØØ9F:ØØØ5               CURRENT DOS LOCATION
ØØ9F:ØØ11               DEFAULT DRIVE
ØØ9F:ØØØB               DEVICE LIST
ØØ9F:ØØ13               FILES
ØØ9F:ØØØ9               FINAL DOS LOCATION
ØØ9F:ØØØF               MEMORY SIZE
ØØ9F:ØØØØ               SYSINIT
ØØ9F:ØØØØ               SYSINIT
```

```
ADDRESS          PUBLICS BY NAME

009F:0005        CURRENT DOS LOCATION
009F:0009        FINAL DOS LOCATION
009F:000B        DEVICE LIST
009F:000F        MEMORY SIZE
009F:0011        DEFAULT DRIVE
009F:0012        BUFFERS
009F:0013        FILES
```

You could modify this example in the following ways:

- You can redirect output to the printer by responding PRN to the `List File` prompt.
- You can use the /LINE switch to produce a listing of all line numbers for all modules. (Note that the /LINE switch may generate a large volume of output.)
- You can press **Enter** in response to the Libraries prompt to perform an automatic library search.

Chapter 9

Code Page Switching

Introduction 9-3
Using Code Page Switching 9-4
Setting Up Your System for Code Page Switching 9-7
Switching Between Code Pages 9-9

Introduction

Code page switching lets your computer use various character sets. If you live in the United States and use only the English character set, you do not need to use this feature. If you live outside the United States or work with other languages, you may want to use this feature to select different character sets.

A character set consists of certain letters, numbers, and symbols that are used for typing, displaying, and printing. Your computer comes with a built-in character set called a code page. Code page switching provides additional character sets that your computer can use.

Each code page, which is identified by a number, contains one character set of 256 characters. The following code pages are available with MS-DOS 4.01:

Code Page Number	*Country or Language*
437	United States (English)
850	Multilingual
860	Portuguese
863	Canada (French)
865	Nordic (Norwegian and Danish)

See Appendix C, Code Page Tables, for illustrations of the exact letters, numbers, and symbols in each of these code pages.

Code page 850 contains all the characters for most European and North and South American countries. This code page provides additional characters that may be useful. The countries with languages that are supported by code page 850 include Australia, Belgium, Canada (English), Canada (French), Denmark, Finland, France, Germany, Italy, Latin America, Netherlands, Norway, Portugal, Spain, Sweden, Switzerland (French), Switzerland (German), the United Kingdom, and the United States (English).

The following table shows the code pages available for each country or language:

Country or Language	*Code Page(s)*
Arabic countries	437, 850
Australia	437, 850
Belgium	437, 850
Canada (English)	437, 850
Canada (French)	863, 850
Denmark	865, 850
Finland	437, 850
France	437, 850
Germany	437, 850
Hebrew	437
Italy	437, 850
Latin America	437, 850
Netherlands	437, 850
Norway	865, 850
Portugal	860, 850
Spain	437, 850
Sweden	437, 850
Switzerland (French)	437, 850
Switzerland (German)	437, 850
United Kingdom	437, 850
United States	437, 850

If a file is created using a particular code page, you must view it in that code page. Because most countries have multiple code pages, you may need to use code page switching.

Using Code Page Switching

To use code page switching, you must have an EGA monitor and a printer that comes with the required code pages built-in or that allows downloading of character sets from your computer.

Code page switching requires several MS-DOS commands and files and is used by several commands. Your MS-DOS startup diskette includes two code page information (CPI) files that contain data required to prepare and select code pages:

- EGA.CPI contains code page information related to your EGA monitor
- LCD.CPI contains code page information related to LCD monitors
- 4201.CPI contains code page information related to the IBM Proprinter II Model 4201 printer and any printers that emulate the 4201 printer, such as the Epson FX-286 printer.
- 4208.CPI contains code page information related to the IBM Proprinter X24 and XL24 printers and any printers that emulate them.
- 5202.CPI contains code page information related to the IBM Quietwriter III Printer and any printer that emulates the IBM Quietwriter III.

You use the following MS-DOS commands to prepare and change the code pages:

CHCP	Changes or displays the current code page. You must install the NLSFUNC command before you use this command.
NLSFUNC	Installs support for code page switching. This command uses the country information provided by the specified file or by the COUNTRY.SYS file (or any other country file you specify with the COUNTRY configuration command).
KEYB	Specifies a keyboard layout and a code page for your keyboard. See Appendix B, International Keyboard Layouts, for the possible keyboard layouts.

COUNTRY Specifies country-dependent display parameters (such as date and time formats) and activates the default or new code page for the specified country. You enter this command in your CONFIG.SYS file.

DEVICE=DISPLAY.SYS

Installs the DISPLAY.SYS device driver, which sets the hardware code page for your monitor. You enter this command in your CONFIG.SYS file.

DEVICE=PRINTER.SYS

Installs the PRINTER.SYS device driver, which sets the hardware code page for the specified printer. You can enter one or more of these device drivers in your CONFIG.SYS file.

MODE CODEPAGE PREPARE

Prepares one or more code pages for use with a specific device. The device must be previously defined with a DEVICE command in your CONFIG.SYS file.

MODE CODEPAGE SELECT

Selects a prepared code page for a specific device. The device and code page must be previously prepared with the MODE CODEPAGE PREPARE command.

MODE CODEPAGE/STATUS

Displays the code pages currently prepared and selected for a device. The device and code page must be previously prepared with the MODE CODEPAGE PREPARE command.

MODE CODEPAGE REFRESH

Reselects a code page that was deactivated because of a hardware error. You may need to use this command to restore a code page to a printer that was turned off by mistake.

The following commands are affected by the setting of the code page:

GRAFTABL — Provides for displaying graphics characters with decimal values above 127 for a specific code page. These characters can be displayed when your computer is in graphics mode.

SORT — Sorts data based on the active code page.

Setting Up Your System for Code Page Switching

To set up your computer to use a character set other than the United States character set, you need to create a CONFIG.SYS file, and optionally, an AUTOEXEC.BAT file. After you create these files, you must restart MS-DOS to enable the new information.

In your CONFIG.SYS file, include the following commands:

COUNTRY — Use this command to define your country code, code page, and country file with country-specific information.

The code page you specify with this command is the default code page whenever you start the system with this CONFIG.SYS file.

This command sets up country and code page information for your system only. You must also install this type of information for each device attached to your computer.

To use the German language with code page 850, enter the following command in your CONFIG.SYS file:

```
COUNTRY = Ø49,85Ø
```

DEVICE — Use this command to set up code page information for your console and printers. The device drivers specify the device type, the hardware code page, and the maximum number of code pages.

If you have an EGA monitor, an Epson FX-286 printer, and an IBM Proprinter II Model 4201, your DEVICE commands might look like this:

```
DEVICE=DISPLAY.SYS CON=(EGA,437,2)
DEVICE=PRINTER.SYS PRN=(4201,437,1)
DEVICE=PRINTER.SYS LPT2=(4201,437,2)
```

The commands you enter in an AUTOEXEC.BAT file are executed automatically when you turn on or reset your computer. In your AUTOEXEC.BAT file, you can include the following commands:

NLSFUNC

Installs support for code page switching. Include this command in your AUTOEXEC.BAT file to ensure that code page switching is installed each time you turn on or reset your computer:

```
NLSFUNC COUNTRY.SYS
```

KEYB

Loads keyboard support and a code page for a specific country or language.

To set up your keyboard for the German language with code page 850, enter this command in your AUTOEXEC.BAT file:

```
KEYB GR, 850
```

MODE CODEPAGE PREPARE

Prepares one or more code pages for use with the device(s) you specified with a DEVICE command in your CONFIG.SYS file.

The number of code pages you prepare for a device depends on the number of code pages you specified in the DEVICE command.

If you installed an EGA monitor, an Epson FX-286 printer, and an IBM Proprinter II Model 4201 in your CONFIG.SYS file, you might prepare them for code page switching with these commands:

```
MODE CON CP PREP=((850)EGA.CPI)
MODE PRN CP PREP=((850)4201.CPI)
MODE LPT2 CP PREP=((850)4201.CPI)
```

If you prefer, you can use the INSTALL command to load KEYB and NLSFUNC from your CONFIG.SYS file. See Chapter 6 for information about using INSTALL.

Switching Between Code Pages

Once you have installed code page switching for your system and devices, you use two commands to switch between code pages: CHCP and MODE.

CHCP

Selects a code page at the system level. When you use the COUNTRY command, MS-DOS automatically assigns two code pages to your country and selects one of them. To switch to the other code page assigned to your country, use the CHCP command.

To activate code page 850 at the system level for all installed devices, enter the following command:

```
CHCP 850
```

MODE CODEPAGE SELECT

Selects a code page for a specific device. You must have prepared the device with the MODE CODEPAGE PREPARE command.

To activate code page 850 for the CON device, enter the following command:

```
MODE CON CP SEL = 850
```

MODE CODEPAGE REFRESH

Reselects a code page that was deactivated because of a hardware error. You may need to use this command to restore a code page to a printer that was turned off by mistake. You must prepare the code page again and then use this command.

To reselect your printer, enter the following commands:

```
MODE PRN CP PREP=((437)4201.CPI)
MODE PRN CP REF
```

Appendix A

Error Messages

Introduction A-3
Error Message Listing A-4

Introduction

You may see three types of error messages on your screen:

- MS-DOS utility messages
- MS-DOS device error messages
- Application program messages.

MS-DOS utility and device error messages are listed in this appendix. For information about error messages generated by an application program, see the documentation for that application program.

If a disk or device error occurs at any time during a command or program, MS-DOS displays an error message, and includes this prompt:

```
Abort, Ignore, Retry, Fail?_
```

MS-DOS waits for you to enter one of the following responses:

A Abort. End the program requesting the disk read or write.

I Ignore. Ignore tells the computer to disregard the bad sector and pretend the error did not occur. This may result in lost data.

R Retry. Repeat the operation. You should use this response when you have corrected the error (for example, with Not ready or Write protect errors).

F Fail. This causes the current MS-DOS system operation to end (fail) and the application to continue. The command line may redisplay so you can retype it.

Error Message Listing

This section describes MS-DOS messages, their causes, and how to correct them.

`10 Mismatches - ending compare`
The COMP command stopped because the files you are comparing differ in at least ten places.

`80386 XMA Emulator not installed`
`Insufficient memory for requested parameter value`
XMAEM.SYS cannot be installed because you do not have enough extended memory for the *size* parameter you specified. Reduce the value for *size* or omit this parameter and accept the default value.

`80386 XMA Emulator not installed`
`Protected mode already in use`
XMAEM.SYS cannot be installed because your 80386 processor is already running in protected mode.

`80386 XMA Emulator not installed`
`Parameter value less than 64 page minimum required`
XMAEM.SYS cannot be installed because you specified a *size* parameter less than 64. Increase the value of *size* to at least 64 or omit this parameter and accept the default value.

`80386 XMA Emulator not installed`
`This system unit is not supported`
You must have an IBM 80386-based computer to use XMAEM.SYS.

`Access denied`
You tried to write to or delete a read-only or locked file.

`Active Code Page not available from con device`
The code page that the system is currently using is not supported on the console (screen) you are using.

`Adapter error`
XMA2EMS.SYS was not installed because of hardware problems it encountered during startup.

`All available space in the Extended DOS Partition is assigned to logical drives`

FDISK displays this message if you try to add more logical drives but no space is available in the extended partitions.

`Allocation error, size adjusted`

CHKDSK displays this message if the size of a file indicated by its directory entry does not match the amount of data actually allocated to the file. CHKDSK truncates the file to match the directory entry.

`An error occurred while installing DOS`
`Press Enter to continue, or Esc to exit SELECT`

The MS-DOS installation program encountered an unexpected error.

`ANSI.SYS must be installed to perform requested function.`

You asked MODE to perform a screen function that requires ANSI.SYS. Add a DEVICE=ANSI.SYS command to your CONFIG.SYS file, then restart your computer.

`APPEND already installed`

You issued an APPEND /E or APPEND /X command after APPEND was already installed. You can include these switches only the first time you use APPEND.

`APPEND/ASSIGN Conflict`

You cannot use APPEND with an ASSIGNed drive.

`Attempting to recover allocation unit x`

FORMAT found a bad cluster while formatting a disk or diskette.

`/B invalid with a black and white printer`

GRAPHICS /B can be used only with color printers.

`Bad command or file name`

You tried to run a program or use a command that is not stored in the current directory or a directory specified by PATH or APPEND.

`Bad or missing d:path SMARTDRV.SYS`
The SMARTDRV.SYS file is not located in the directory specified in your CONFIG.SYS file. Check the location of SMARTDRV.SYS, then edit your CONFIG.SYS file and restart your computer.

`Bad or missing Command interpreter`
MS-DOS cannot find the file COMMAND.COM, or the COMMAND.COM file is invalid. You may also receive this message if COMMAND.COM has been moved from the directory it was in when you started MS-DOS.

`Bad or missing filename`
CONFIG.SYS is unable to find a device driver file that you specified in a DEVICE= command. Check the location of the file, edit your CONFIG.SYS file, and restart your computer.

`Bad or Missing Keyboard definition file`
KEYB is unable to find the KEYBOARD.SYS file or the keyboard definition file you specified.

`Bad Partition Table`
There is no MS-DOS partition on the hard disk. Use FDISK to create at least one MS-DOS partition.

`Bad unit error reading drive x:`
Invalid subunit numbers were passed to a device driver.

`Batch file missing`
A batch file name was referenced that could not be found.

`Baud rate required`
You must specify a baud rate when using MODE to set serial port parameters.

`Cannot CHDIR to path -`
`tree past this point not processed`
CHKDSK is unable to go to the specified directory while checking your disk. All subdirectories of the specified directory are not verified. To correct the problem, use CHKDSK again with the /F switch.

`Cannot CHDIR to root`

CHKDSK is unable to return to the root directory while checking your disk. All remaining subdirectories are not verified. Try to restart your computer. If this error persists, your disk is unusable.

`Cannot Chkdsk a Network drive`

You cannot use CHKDSK on a drive that is redirected over the network.

`Cannot Chkdsk a SUBSTed or ASSIGNed drive`

You cannot use CHKDSK on a drive that has been created with the SUBST or ASSIGN command.

`Cannot create a zero size partition`

FDISK requires that you allocate at least 1 cylinder to each partition.

`Cannot create extended DOS partition without primary DOS partition on disk 1`

FDISK requires that a primary MS-DOS partition be created on your first hard disk before you can create an extended partition.

`Cannot create Logical DOS drive without an Extended DOS Partition on the current drive`

FDISK requires that you create an extended MS-DOS partition before you create any logical drives.

`Cannot delete Extended DOS Partition while logical drives exist`

You must delete all logical drives in an extended partition before using FDISK to delete the extended partition.

`Cannot DISKCOMP to or from an ASSIGNed or SUBSTed drive`

One of the drives specified in a DISKCOMP command was created using the ASSIGN or SUBST command.

`Cannot DISKCOMP to or from a network drive`

You cannot compare disks on drives that have been redirected over the network.

`Cannot DISKCOPY to or from a network drive`
You cannot copy disks to or from drives that have been redirected over the network.

`Cannot do binary reads from a device`
You cannot use COPY with the /B switch when copying from a device. Remove the /B switch or use the /A switch to specify an ASCII copy.

`Cannot edit .BAK file-rename file`
EDLIN displays this message if you attempt to edit a file with the extension .BAK. You must rename the file before editing it.

`Cannot find adapter`
XMA2EMS.SYS was unable to find an expanded memory board in your computer.

`Cannot find FORMAT.COM`
You tried to back up files to an unformatted diskette, and BACKUP was unable to find the FORMAT.COM file.

`Cannot find the GRAPHICS profile`
GRAPHICS was unable to find GRAPHICS.PRO or the graphics profile file you specified.

`Cannot find System Files`
FORMAT was unable to create a system diskette because it couldn't find the MS-DOS system files.

`Cannot find XMA Emulator device driver`
You tried to install XMA2EMS.SYS without first installing XMAEM.SYS. Edit your CONFIG.SYS file, then restart your computer.

`Cannot format an ASSIGNed or SUBSTed drive`
You cannot use FORMAT on a drive that has been created with the SUBST or ASSIGN command.

`Cannot FORMAT a Network drive`
You cannot format drives that are redirected over the network.

`Cannot FORMAT nonremovable drive x:`
You tried to back up files to an unformatted hard disk. You must format the disk first.

`Cannot JOIN a Network drive`
You cannot use JOIN with drives that are redirected over the network.

`Cannot LABEL a Network drive`
You cannot use LABEL with a drive that is redirected over the network.

`Cannot LABEL a SUBSTed or ASSIGNed drive`
You cannot use LABEL with a drive that is redirected over the network.

`Cannot load COMMAND, system halted`
MS-DOS cannot reload the command processor. Restart your computer.

`Cannot perform a cyclic copy`
When using XCOPY with the /S switch, you cannot specify a target that is a subdirectory of the source.

`Cannot read file allocation table`
RECOVER displays this message if the FAT (file allocation table) on your disk has been damaged.

`Cannot recover . entry, processing continued`
CHKDSK displays this message when the "." entry (the working directory) is defective and cannot be recovered.

`Cannot recover .. entry,`
`Entry has a bad attribute (or link or size)`
CHKDSK displays this message when the ".." entry (the parent directory) is defective. If you specified the /F switch, CHKDSK tries to correct the error.

`Cannot RECOVER a Network drive`
You cannot recover files on a drive that is redirected over the network.

`Cannot setup expanded memory`
FASTOPEN displays this message when an expanded memory card in your system is not working properly.

`Cannot specify default drive`
SYS displays this message when the target diskette is in the current default drive. Log onto a different drive, then reuse the SYS command.

`Cannot start COMMAND, exiting`
The FILES value in CONFIG.SYS needs to be increased, or the path to the command processor was set incorrectly, or there was not enough memory available to load the command processor. Restart your computer after editing your CONFIG.SYS file.

`Cannot SUBST a Network drive`
You cannot use the SUBST command with a drive that is redirected over the network.

`Cannot SYS to a Network drive`
You cannot use the SYS command with a drive that is redirected over the network.

`Cannot use FASTOPEN for drive x:`
You cannot use FASTOPEN with diskettes, with drives that are redirected over a network, or with more than four hard disks at a time.

`Cannot use PRINT - Use NET PRINT`
You cannot use PRINT on a network server system. Use Net Print instead.

`Cannot XCOPY to a reserved device`
The target for an XCOPY command cannot be a printer or other character device.

`CHDIR .. failed, trying alternate method`
When checking the tree structure, CHKDSK was unable to return to a parent directory. It tries to return to that directory by starting over at the root.

`CHKDSK not available on drive x:`
You are trying to use CHKDSK with an alternate file system that cannot be found.

`xxxxxxxx code page drive cannot be initialized`
You specified an invalid code page in your CONFIG.SYS file. Restart your computer after editing your CONFIG.SYS file.

`Code page not prepared`
MODE displays this message if you select a code page that has not yet been prepared for the system or one that does not have the correct font to support the current video mode.

`Code page xxx not prepared for all devices`
CHCP displays this message if you select a code page that is not currently supported by a device.

`Code page xxx not prepared for system`
CHCP displays this message if NLSFUNC is not installed.

`Code page operation not supported on this device`
MODE displays this message if you specify an invalid device and code page combination.

`Code page requested xxx is not valid for given keyboard code`
You used the KEYB command with an invalid keyboard code and code page.

`Code page specified has not been designated`
You used the KEYB command with an option the system does not recognize. First prepare the associated code page for your display screen.

`Code page specified has not been prepared`
You used the KEYB command with an option the system does not recognize. First prepare the associated code page for your display screen.

`Code page specified is inconsistent with invoked code page`

This message warns that the KEYB option you selected does not coincide with the code page for your console screen device (CON).

`Code page specified is inconsistent with selected code page`

This message warns that the KEYB option you selected does not coincide with the code page for your console screen device (CON).

`Code pages cannot be prepared`

MODE displays this message if you specify a duplicate code page for a device or try to prepare more than the total number of code pages supported for a device.

`Compare error at OFFSET` *x*

COMP displays this message to indicate the location of a difference between specified files.

`Compare error on disk side` *s*`, track` *t*

DISKCOMP displays this message to indicate the location of a difference between specified diskettes.

`Compare process ended`

DISKCOMP displays this message if a fatal error occurs during the comparison.

`COM port does not exist`

MODE displays this message if you specify an invalid COM port.

`Content of destination lost before copy`

The source file you specified in a COPY command was overwritten before the copy could be completed.

`Convert directory to file (Y/N)?`
CHKDSK displays this prompt if a directory is no longer usable because it contains too much invalid information. If you press **Y**, CHKDSK converts the directory to a file, allowing you to fix or delete it.

`Convert lost chains to files (Y/N)?`
CHKDSK displays this prompt if you use the /F switch and CHKDSK finds lost blocks (data on the disk that isn't allocated correctly in the file allocation table). If you press **Y**, CHKDSK recovers the lost blocks and creates a directory entry and file for each recovered chain. If you press **N**, CHKDSK frees the lost blocks so they can be used by other files; the data in the lost blocks is not recovered.

`Copy process ended`
DISKCOPY displays this message if a fatal error occurs during the copying.

`Corrections will not be written to disk`
CHKDSK displays this message if it finds errors but you do not include the /F switch.

`Current drive is no longer valid`
Your current drive is either a network drive or a disk drive with no disk in it.

`Current keyboard does not support this code page`
KEYB displays this message if the code page you selected is not compatible with the current keyboard code.

`Data error reading drive x:`
MS-DOS could not read the diskette or disk properly.

`Device Error during Status`
MODE displays this message if it finds an error with the specified device when checking the status of that device. The device may not support code pages, the device may not be prepared for code page switching, the device may not support more code pages than the number already prepared, or the device may have a bad or irregular font file.

`Device Error during Prepare`

MODE displays this message if it finds an error with the specified device while preparing that device for code page switching. The device may not support code pages, the device may not be prepared for code page switching, the device may not support more code pages than the number already prepared, or the device may have a bad or irregular font file.

`Device Error during Select.`

MODE displays this message if it finds an error with the specified device. The device may not support code pages, the device may not be prepared for code page switching, the device may not support more code pages than the number already prepared, or the device may have a bad or irregular font file.

`Device Error during write of font file to device`

MODE displays this message if it finds an error while trying to write a font file to the specified device. The device may not support code pages, the device may not be prepared for code page switching, the device may not support more code pages than the number already prepared, or the device may have a bad or irregular font file.

`Device or code page missing from font file`

MODE displays this message if it does not find a definition of the indicated code page for the specified device in the font file. Check to see that the font file supports the code page you want to use. Use MODE to prepare and refresh lost code pages.

`Device` *`ddd`* `not prepared`

MODE displays this message if no code page has been prepared for the specified device.

`Directory is joined`

CHKDSK displays this message if you try to check a drive with JOINed directories. Use JOIN /D to separate the directories, then use CHKDSK again.

`Directory is totally empty, no . or ..`

CHKDSK displays this message if a directory does not contain references to the working and parent directories. Delete the specified directory and recreate it.

`Directory not empty`

JOIN displays this message if you try to join a drive to a directory that is not empty.

`Disk error reading (or writing) drive x:`

MS-DOS cannot read data from the specified disk. The disk may be defective.

`Disk error reading (or writing) FAT x`

CHKDSK displays this message if one of the FATs (file allocation tables) for the specified drive is defective. CHKDSK automatically uses the duplicate FAT.

`Disk full, Edits lost`

EDLIN could not save your file because there was not enough disk space.

`Disk full error writing to BACKUP Log File`

BACKUP could not create the log file because there was not enough disk space.

`Disk unsuitable for system disk`

FORMAT could not transfer the MS-DOS system files because it found a bad track where the system files should reside.

`Do not specify filename(s)`
`Command format: DISKCOMP d: d:[/1][/B]`

You specified an invalid switch or tried to include a filename in a DISKCOMP command.

`Do not specify filename(s)`
`Command format: DISKCOPY d: d:[/1]`

You specified an invalid switch or tried to include a filename in a DISKCOPY command.

`(.)(..) Does not exist`
CHKDSK displays this message if either the "." or the ".." directory entry is invalid.

`Drive x already deleted`
FDISK displays this message if you try to delete a logical drive that has already been deleted.

`Drive letter must be specified`
FORMAT displays this message if you use the command without specifying a target drive.

`Drive types or diskette types not compatible`
This message appears if you try to use DISKCOMP or DISKCOPY with two drives that are not of the same type.

`Duplicate file name or file not found`
RENAME displays this message if the file you want to rename does not exist or if the new name is already in use by another file in the same directory.

`Duplicate parameters not allowed`
GRAPHICS displays this message if you try to specify a parameter twice.

`Entry error`
EDLIN displays this message when the last command you typed contains a syntax error.

`EOF mark not found`
COMP displays this message when a file does not end in Ctrl Z.

`Error during read of Font file`
MODE found an error when it tried to read the font file for the specified code page.

`Error in country command`
The syntax for the COUNTRY command in your CONFIG.SYS file is incorrect. Restart your computer after editing your CONFIG.SYS file.

`Error in .EXE file`
You tried to run a program with an invalid .EXE file.

`Error opening log file`
The BACKUP command was unable to open the BACKUP.LOG file. Be sure to specify a valid filename or accept the default (BACKUP.LOG). Do not try to store the log file on the target drive.

`Error reading directory`
After formatting, the FORMAT command was unable to read the root directory of the target disk or diskette.

`Error reading fixed disk`
After five tries, FDISK was unable to read the boot record of the target disk.

`Error reading GRAPHICS profile`
The GRAPHICS command was unable to read GRAPHICS.PRO or the specified graphics profile file. If the file is stored on a diskette, be sure the drive door is closed.

`Error reading/writing partition table`
FORMAT displays this message if it cannot read the partition table of the target hard disk. Use FDISK again to partition the disk, then use FORMAT again.

`Errors found, F parameter not specified`
`Corrections will not be written to disk`
CHKDSK displays this message if it finds errors but you do not include the /F switch.

`Errors on list device indicate that it`
`may be off-line. Please check it.`
PRINT displays this message if your printer is turned off, off line, or disconnected from your computer.

`Error writing directory (FAT or fixed disk)`
FORMAT and FDISK display this message if the target disk's root directory or file allocation table has bad sectors.

`File allocation table bad drive x:`
CHKDSK displays this message if the target disk is not formatted, was formatted incorrectly, or was formatted by an operating system other than MS-DOS.

`File filename canceled by operator`
PRINT displays this message when you use the /T switch to remove a file from the print queue.

`File cannot be converted`
EXE2BIN displays this message if the input file is in the wrong format.

`File cannot be copied onto itself`
The source file specification (drive, pathname, and filename) for a COPY, REPLACE, or XCOPY command is the same as the target file specification.

`File creation error`
MS-DOS was unable to create a specified file. Possible reasons include the following: the root directory already has 512 entries; the filename is the same as a volume label, directory, or hidden or system file; there is not enough disk space for the file; you tried to copy over a read-only file.

`File is READ-ONLY`
EDLIN displays this message if you try to write to a file with the read-only attribute. Use ATTRIB to remove the read-only attribute, or use a different filename.

`File name must be specified`
EDLIN displays this message if you try to start without naming a file. Type EDLIN again followed by a filename.

`File not found`
You tried to access a file that does not exist.

`File not in PRINT queue`
You tried to use PRINT /T to remove a file that is not in the print queue.

`Files are different sizes`
COMP does not compare the specified files because they are not the same size.

`FIND: Access denied`
FIND displays this message if you try to access a read-only or locked file or a file on a write-protected diskette.

`FIND: File not found`
The file you specified in a FIND command does not exist.

`FIND: Invalid number of parameters`
You specified too many or too few parameters in a FIND command.

`FIND: Invalid Parameter`
One or more of the switches you specified in a FIND command is invalid.

`FIND: Read error in` *`filename`*
The FIND command could not read the specified file.

`FIND: Syntax error`
You entered incorrect syntax in a FIND command.

`First cluster number is invalid, entry truncated`
CHKDSK displays this message if the directory entry for a specified file points to an invalid cluster. If you include the /F switch, CHKDSK truncates the file to a length of 0 bytes.

`FIRST diskette bad or incompatible`
DISKCOMP does not recognize the format on the source disk. Use CHKDSK to help identify the problem.

`Fixed backup device` *`x`* `is full`
BACKUP displays this message if your target disk is a hard disk with insufficient free space.

`Fixups needed – base segment hex:`
EXE2BIN displays this message when the source file contains information indicating that a load segment is required.

`Font File contents invalid`
MODE displays this message if a specified font file is invalid or corrupted. If you typed the name of the file correctly, recopy the file from your MS-DOS distribution diskette to your working diskette or hard disk and try again.

`For cannot be nested`
You cannot nest FOR statements in a batch file.

`Format failure`
FORMAT was unable to format the disk in the specified drive. This message is usually accompanied by another explaining the reason for the failure.

`Format not supported on drive x:`
You cannot use FORMAT to format this drive. You may have specified device parameters that your computer cannot support.

`Function not supported on this computer`
MODE displays this message if your computer does not have the device or adapter required to perform the function you specified.

`General failure reading (or writing) drive x:`
An unusual error has occurred. In most cases, this error must be fixed by an experienced programmer.

`Graphics characters already loaded`
The GRAFTABL command displays this message if you have already loaded the table of graphics characters into memory.

`Has invalid cluster, file truncated`
CHKDSK displays this message if the directory entry for a specified file points to an invalid cluster. If you include the /F switch, CHKDSK truncates the file to a length of 0 bytes.

`Illegal device name`
MODE displays this message if your computer does not recognize a specified device name.

`Incompatible system size`
SYS displays this message if the system files occupy more space on the source disk than is available on the target disk.

`Incorrect APPEND Version`
You tried to use the APPEND command from an earlier version of MS-DOS.

`Incorrect DOS Version`
You tried to use a command file from an earlier version of MS-DOS.

`Incorrect DOS Version, use DOS 2.00 or later`
LINK displays this message if you try to use certain MS-DOS utilities from a version earlier than 2.00.

`Incorrect number of parameters`
JOIN and SUBST display this message if you specify too many or too few parameters.

`Incorrect parameter`
ASSIGN and SHARE display this message if one of the parameters you specify is wrong.

`Insufficient disk space`
You tried to store a file that is larger than the available disk space.

`Insufficient memory`
BACKUP, CHKDSK, DISKCOMP, DISKCOPY, EDLIN, REPLACE, RESTORE, SORT, and XCOPY display this message if your computer does not have enough memory available to perform the specified operation.

`Insufficient room in root directory`
`Erase files in root and repeat CHKDSK`
If there are too many entries in the root directory, CHKDSK displays this message when attempting to recover lost clusters. Erase some files from the root directory and try again.

`Intermediate file error during pipe`
Pipe operations create intermediate files that are deleted at the end of the operation. An error has occurred that involves one of these files. Be sure there is enough room on the current disk and that the current disk is not write-protected.

`Internal error`
This message indicates an error in the FC, MODE, or SHARE command.

`Internal stack overflow`
`System Halted`
The system tried to use more stacks than were available. Restart MS-DOS. Then edit your CONFIG.SYS file to allocate more stack resources (see Chapter 6). Restart your computer after editing your CONFIG.SYS file.

`Invalid argument`
You used incorrect syntax in the BACKUP, FC, or RESTORE command.

`Invalid baud rate specified`
The baud rate you specified in a MODE command is not supported. Valid choices are 110, 150, 300, 600, 1200, 2400, 4800, and 9600. You must specify at least two digits of the baud rate.

`Invalid characters in volume label`
FORMAT and LABEL display this message if you specify a volume label that includes any of the following characters:
. " / \ [ ] : * < > ¦ + = ; , ?

`Invalid code page specified`
CHCP displays this message if you select an invalid code page number.

`Invalid combination of parameters`
You specified a combination of parameters that cannot be used.

`Invalid COMMAND.COM`
`Insert COMMAND.COM disk in default drive`
`and strike any key when ready`

MS-DOS needs to reload the transient portion of COMMAND.COM but cannot find it, or the version it finds is incorrect.

`Invalid country code or code page`

The COUNTRY command in your CONFIG.SYS file specifies an invalid country code or code page. Edit the CONFIG.SYS file and restart your computer.

`Invalid current directory`

Your disk has an invalid directory. Try to recover as many files as possible by copying them with the COPY command, then replace the disk.

`Invalid date`

You specified an invalid date in a DATE or XCOPY command.

`Invalid Date/Time`

You specified an invalid date or time in a BACKUP command.

`Invalid device`

You tried to specify a device other than AUX, CON, NUL, or PRN.

`Invalid device parameters from device driver`

FORMAT displays this message if a partition of the hard disk you are formatting does not start on a track boundary. This might happen if you previously formatted the disk under MS-DOS 2.x without first running FDISK or if you set the device driver parameters incorrectly. Check your CONFIG.SYS file for incorrect DEVICE or DRIVPARM commands. Restart your computer after editing your CONFIG.SYS file.

`Invalid directory`

You specified a directory that is invalid or does not exist.

`Invalid disk change reading (or writing) drive x:`
You changed the diskette in a drive when you were not supposed to. Replace the diskette and press **R** (Retry).

`Invalid drive or filename`
EDLIN and RECOVER display this message if you do not type a valid drive or filename.

`Invalid drive specification`
You tried to run a program on or log onto a drive that does not exist, or you specified an invalid drive as a parameter for BACKUP, CHKDSK, DISKCOMP, DISKCOPY, FORMAT, LABEL, PRINT, REPLACE, RESTORE, SYS, TREE, or XCOPY.

`Invalid entry, please enter x`
FDISK displays this message if you enter a value outside the acceptable range.

`Invalid entry, please press Enter`
FDISK displays this message if you try to type a character after the percent sign when specifying a partition or logical drive size. If you want to enter a different character, first press the backspace key to erase the percent sign.

`Invalid environment size`
The COMMAND command displays this message if you specify an invalid number of bytes with the /E switch.

`Invalid extent entry`
FASTOPEN displays this message if you enter a value for the *m* parameter that is not between 1 and 999.

`Invalid file/directory entry`
FASTOPEN displays this message if you enter a value for the *n* parameter that is not between 10 and 999.

`Invalid keyboard code specified`
KEYB displays this message if you specify an invalid keyboard code.

`Invalid number of parameters`
You specified the wrong number of parameters in an ATTRIB, BACKUP, FC, FIND, RECOVER, RESTORE, or XCOPY command.

`Invalid parameter (or parameter combination)`
You specified an incorrect switch or parameter in an APPEND, CHKDSK, FASTOPEN, or REPLACE command.

`Invalid parameter(s)`
You specified an incorrect switch or parameter in a BACKUP, CHKDSK, DISKCOMP, DISKCOPY, EDLIN, FIND, FORMAT, JOIN, MODE, PRINT, REPLACE, RESTORE, SORT, SUBST, SYS, TREE, or XCOPY command.

`Invalid partition table`
FDISK found invalid information in a hard disk's partition table. Use FDISK to display the partition information and correct the problem.

`Invalid path`
You specified a path that is longer than 63 characters or that includes invalid characters.

`Invalid path, not directory, or directory not empty`
RMDIR is unable to remove the directory you specified.

`Invalid path (or file not found)`
ATTRIB, BACKUP, COPY, RESTORE, TREE, and XCOPY display this message if you enter a pathname or filename that does not exist.

`Invalid path or parameter`
APPEND displays this message if you specify a file or directory that does not exist.

`Invalid profile statement on line` *x*
GRAPHICS displays this message if the GRAPHICS profile file contains an invalid command.

`Invalid STACK parameter`
The STACKS command in your CONFIG.SYS file contains an invalid parameter. Edit your CONFIG.SYS file and restart your computer.

`Invalid sub-directory entry`
CHKDSK displays this message if you specify a subdirectory that does not exist or is invalid.

`Invalid switch type`
FASTOPEN displays this message if you specify an invalid switch, a duplicate switch, or switches in the wrong order.

`Invalid syntax`
You used incorrect syntax in entering an MS-DOS command.

`Invalid syntax on DISPLAY.SYS code page driver`
The DEVICE=DISPLAY.SYS command in your CONFIG.SYS file includes incorrect syntax. Restart your computer after editing your CONFIG.SYS file.

`Invalid syntax on PRINTER.SYS code page driver`
The DEVICE=PRINTER.SYS command in your CONFIG.SYS file includes incorrect syntax. Restart your computer after editing your CONFIG.SYS file.

`Invalid syntax on PRINTER.SYS code page switching device drivers`
The DEVICE=PRINTER.SYS command in your CONFIG.SYS file includes incorrect syntax. Restart your computer after editing your CONFIG.SYS file.

`Invalid time`
You specified an invalid time in a TIME command.

`Invalid Volume ID`
FORMAT displays this message if you try to format a hard disk and specify a volume label that does not match the current volume label of the hard disk. Use the VOL command, if necessary, to determine the current volume label, then retry FORMAT.

`Is cross linked on allocation unit x`
CHKDSK found two files using the same cluster (allocation unit). You can correct the problem by using CHKDSK with /F, but this may truncate a file.

`x is not a choice, Please enter y-z`
FDISK displays this message if you try to select an invalid option. Choose an option from the range indicated.

`KEYB has not been installed`
You must use the KEYB command before changing the default keyboard layout.

`Label not found`
A GOTO command in the batch file you ran refers to a label that does not exist.

`Last backup diskette not inserted`
`Insert last backup diskette in drive x:`
`Strike any key when ready`
BACKUP uses this message to prompt you to insert the final backup diskette.

`*** Last file not backed up ***`
BACKUP could not back up the last file on the source disk. This message may appear if there is not enough room on the target disk or if there is an error in the source file or on the target disk. You may have to back up this file separately on another disk.

`Line too long`
EDLIN displays this message if the replacement string you give in an R command causes the line to exceed 253 characters.

`List output is not assigned to a device`
PRINT displays this message if you specify a device that does not exist.

`Lock violation reading (or writing) drive x:`
A program tried to access a file or record that was in use by another program. Press **A** (Abort) or wait a while and press **R** (Retry).

`LPTx: not rerouted`
MODE could not redirect parallel printer output to the port you specified.

`Memory allocation error.`
`Cannot load MS-DOS, system halted`
Restart MS-DOS. If the error persists, restart using the backup copy of your Startup diskette. Then make a new backup copy.

`MORE: Incorrect DOS version`
You tried to run MORE on an earlier version of MS-DOS.

`Must enter both /T and /N parameters`
FORMAT displays this message if you use either /T or /N without the other.

`Must specify COM1, COM2, COM3 or COM4`
You must specify a serial port when using MODE to set serial parameters.

`Must specify destination line number`
You did not specify a destination line number for an EDLIN C (copy) or M (move) command.

`Must specify ON or OFF`
The command you typed requires an ON or OFF parameter.

`NLSFUNC already installed`
You have already loaded NLSFUNC and cannot load it again.

`No drive specified`
The DEVICE=DRIVER.SYS command in your CONFIG.SYS file must specify a physical drive number. Restart your computer after editing your CONFIG.SYS file.

`No Extended DOS Partition to delete`
You tried to use FDISK to delete an extended MS-DOS partition, but there is no such partition on the current hard disk.

`No fixed disks present`
You tried to run FDISK, but your system does not recognize any hard disks. If you have a hard disk and see this error message, you may need to perform a physical format of the hard disk.

`No free file handles.`
`Cannot start COMMAND.COM, exiting`
MS-DOS is unable to reload the transient part of COMMAND.COM. Restart your computer. Then edit the FILES= command in your CONFIG.SYS file to increase the number of files. Then restart your computer again.

`No Logical DOS Drive(s) to delete`
FDISK displays this message if you try to delete a logical drive from an extended MS-DOS partition when no logical drive exists.

`No paper error writing device` *`dev`*
The printer is either out of paper or not turned on.

`No partitions to delete`
FDISK displays this message if you try to delete an MS-DOS partition before any partitions have been created.

`No partitions to make active`
FDISK displays this message if you try to activate a primary partition before any partitions have been created.

`No primary DOS partition to delete`
FDISK displays this message if you try to delete a primary MS-DOS partition before any partitions have been created.

`No room for system on destination disk`
SYS displays this message when the target diskette does not have enough free space or enough available root directory entries for the MS-DOS system files. Delete some files from the target diskette, then reuse the SYS command.

`No room in directory for file`
EDLIN displays this message if you try to create a file in a root directory that is already full or if you specify an invalid disk drive or filename. The root directory is limited to 512 files. All other directories may have any number of files.

`No source drive specified`
BACKUP displays this message if you fail to specify a source drive.

`No space left on device`
BACKUP and RESTORE display this message if the drive to which you are copying files is full. FC displays this message if the default drive has no room for output from a file comparison. Delete some files and try again.

`No space to create a DOS partition`
FDISK displays this message if you try to create an MS-DOS partition when the current partitions fill all available space.

`No space to create logical drive`
FDISK displays this message if you try to create a logical drive in an MS-DOS extended partition when the current partitions fill all available space.

`No sub-directories exist`
TREE displays this message when the current directory has no subdirectories.

`No such file or directory`
BACKUP, FC, and RESTORE display this message when one or more of the files or directories that you specified does not exist.

`No system on default drive`
SYS displays this message when the current drive does not contain the MS-DOS system files. Insert an MS-DOS system diskette in the current drive and try again.

`No target drive specified`
BACKUP displays this message if you do not specify a target drive.

`Non-DOS error reading (or writing) drive x:`
This message appears when MS-DOS does not recognize the format of the disk you are trying to read from or write to. Try to correct the problem with CHKDSK /F. If that fails, the disk is not usable unless you reformat it.

`Non-system disk or disk error`
`Replace and strike any key when ready`
This message appears if you try to start MS-DOS from a disk or diskette that does not contain the MS-DOS system files. Insert a system diskette in drive A and press a key.

`*** Not able to back up (or restore) file ***`
BACKUP and RESTORE display this message if there is an error on the source disk.

`Not enough memory`
JOIN, SHARE, and SUBST display this message if there is not enough memory available to perform the function you requested.

`Not enough room to merge the entire file`
EDLIN displays this message if there was not enough memory to hold the file during a T (transfer) command.

`Not ready error reading (or writing) drive x:`
MS-DOS was unable to read from or write to the specified drive. Check to be sure the drive door is closed and retry.

`One or more CON code pages invalid for given keyboard code`

KEYB displays this message if at least one prepared code page is incompatible with your screen console device (CON). This is a warning to let you know that your keyboard and screen console device are working from different code pages.

`Only partitions on drive 1 can be made active`

FDISK displays this message if you try to make a partition on a second or subsequent hard disk active.

`Out of environment space`

You have tried to add environment data, but the environment table is full. Use the COMMAND command with the /E switch to increase the environment space, or use the SET command to remove some of the existing environment variables.

`Parameter syntax or value error`

XMA2EMS.SYS displays this message if you use incorrect syntax in the DEVICE=XMA2EMS.SYS command in your CONFIG.SYS file. Restart your computer after editing the CONFIG.SYS file.

`Parameters not compatible`

FORMAT and REPLACE display this message if you specify a combination of parameters that cannot be used.

`Parameters not compatible with fixed disk`

FORMAT displays this message if you try to format a hard disk and include switches that apply only to diskettes.

`Parameters not supported`

You have specified parameters that MS-DOS does not support.

`Partition selected (x) is not bootable, active partition not changed`

FDISK displays this message if you try to change active partitions but MS-DOS cannot be started from the partition you select.

`Path (name) too long`
PRINT, REPLACE, and XCOPY display this message if you specify a pathname longer than 63 characters.

`Path not found`
CHKDSK, REPLACE, SUBST, and XCOPY display this message if you specify an invalid pathname.

`Primary DOS partition already exists`
FDISK displays this message if you try to create a primary MS-DOS partition when one already exists.

`Printbox ID not in GRAPHICS profile`
GRAPHICS displays this message if you specify a *PRINTBOX (PB)* parameter that does not match the first operand of a Printbox command in the graphics profile file.

`Printer error`
MODE displays this message if you try to set parameters for a printer that is turned off, disconnected, or off line.

`PRINT queue is full`
PRINT displays this message if you try to put more than the maximum number of files in the print queue. The default maximum is 10. You can increase it to as many as 32 by using the /Q switch.

`Printer type not in GRAPHICS profile`
GRAPHICS displays this message if your *type* parameter does not specify a printer that is identified in the graphics profile file.

`Probable non-DOS disk`
`Continue (Y/N)?`
CHKDSK displays this message if it does not recognize the format of the disk you are checking.

`Processing cannot continue`
CHKDSK displays this message if there is not enough memory in your computer to continue running CHKDSK.

`Profile statement out of sequence on line x`
GRAPHICS displays this message if there are statements out of order in the graphics profile file.

`Program too big to fit in memory`
You need more memory to run your application. Try restarting MS-DOS. If the problem persists, you may need to remove one or more device drivers or add memory to your computer.

`RATE and DELAY must be specified together`
`RATE = x`
`DELAY = x`
MODE displays this message if you try to set the keyboard typematic rate without specifying both parameters.

`Read error, COUNTRY.SYS`
MS-DOS cannot read the COUNTRY.SYS file. Restore the file from a backup copy.

`Read error in filename`
EDLIN and FIND display this message if they cannot read all of the specified file.

`Read error, KEYBOARD.SYS`
MS-DOS cannot read the KEYBOARD.SYS file. Restore the file from a backup copy.

`Read fault error reading drive x:`
MS-DOS is unable to read the disk in the specified drive. Make sure the drive door is closed, then press **R** (Retry).

`Requested logical drive size exceeds the maximum available space`
FDISK displays this message if you try to create a logical drive that is larger than the space available in the extended partition.

`Requested partition size exceeds the maximum available space`
FDISK displays this message if you try to create a partition that is larger than the available disk space.

`Requested screen shift out of range`
MODE displays this message if you try to shift the display too far to the left or right.

`Required font not loaded`
MODE displays this message if the DISPLAY.SYS file does not have the font size to do what you asked. Increase the *m* parameter in the DEVICE=DISPLAY.SYS command in your CONFIG.SYS file. Then restart your computer.

`Required profile statement missing before line x`
GRAPHICS displays this message if there is a statement missing in the graphics profile file.

`Restore file sequence error`
RESTORE displays this message if you insert backup diskettes in the wrong order while restoring files.

`Resynch failed. Files are too different`
FC displays this message if no lines match in the portion of the files currently in memory.

`Same drive specified more than once`
This message appears if you try to activate FASTOPEN more than once for the same drive.

`Same parameter entered twice`
FORMAT displays this message if you enter /T, /N, /F, or /V twice.

`SECOND diskette bad or incompatible`
DISKCOMP displays this message if the second diskette you are comparing does not have the same format as the first, or if DISKCOMP does not recognize the format of the second diskette.

`Sector not found error reading (or writing) drive x:`
MS-DOS cannot find the information it is looking for, usually because the disk has a defective spot. Copy all files from the disk to another disk, then try to reformat the affected disk.

`Sector size too large in file` *`filename`*

The named device driver uses a sector size larger than any other device driver on the system. You cannot run this device driver.

`Seek error reading (or writing) drive` *`x`*`:`

MS-DOS is unable to find the information it is looking for on the diskette. Be sure the diskette is properly inserted in the drive.

`SHARE already installed`

SHARE displays this message if you try to install it a second time.

`Sharing violation reading drive` *`x`*`:`

A program tried to access a file in use by another program. Press **A** (Abort) or wait a while and press **R** (Retry).

`SMARTDrive: Computer must be PC-AT or PC-AT compatible`

You tried to install SMARTDRV.SYS in extended memory, but your computer does not have extended memory. If you have expanded memory, you can add the /A switch to the DEVICE=SMARTDRV.SYS command in your CONFIG.SYS file. Then restart your computer. If you do not have extended memory or expanded memory, you cannot use SMARTDRV.SYS.

`SMARTDrive: Expanded Memory Manager not present`

You tried to install SMARTDRV.SYS with the /A switch, but SMARTDRV.SYS did not find your expanded memory manager. Be sure your CONFIG.SYS file includes a command installing an expanded memory manager. Edit your CONFIG.SYS file, then restart your computer.

`SMARTDrive: Expanded Memory Manager Status shows error`

MS-DOS detected an error while trying to set up SMARTDRV.SYS in expanded memory. Run your expanded memory diagnostics to check your expanded memory, and take any necessary corrective action.

`SMARTDrive: I/O error accessing drive memory`
MS-DOS detected an error while trying to set up SMARTDRV.SYS in expanded memory. You need to add memory to your computer to use SMARTDRV.SYS.

`SMARTDrive: Incorrect DOS version`
SMARTDRV.SYS was not installed because you are using an earlier version of MS-DOS.

`SMARTDrive: Insufficient memory`
SMARTDRV.SYS was not installed because your computer does not have enough memory.

`SMARTDrive: Invalid parameter`
SMARTDRV.SYS was not installed because the INSTALL=SMARTDRV.SYS command in your CONFIG.SYS file includes an invalid parameter. Restart your computer after editing your CONFIG.SYS file.

`SMARTDrive: No extended memory available`
You tried to install SMARTDRV.SYS in extended memory, but your computer does not have extended memory. If you have expanded memory, add the /A switch to the DEVICE=SMARTDRV.SYS command in your CONFIG.SYS file. Then restart your computer. If you do not have extended memory or expanded memory, you cannot use SMARTDRV.SYS.

`SMARTDrive: No hard drives in system`
SMARTDRV.SYS was not installed because your computer does not have a hard disk. You must have a hard disk to use SMARTDRV.SYS.

`SMARTDrive: Too many bytes per track on hard disk`
SMARTDRV.SYS was not installed because your hard disk uses a format that SMARTDRV.SYS does not understand.

`SORT: Incorrect DOS version`
You tried to use SORT with an earlier version of MS-DOS.

`SORT: Insufficient disk space`
SORT uses disk space while it works. Your disk does not have enough free space. Delete some files and try again.

`SORT: Insufficient memory`
Your computer does not have enough memory to run SORT.

`Source and target drives are the same`
BACKUP and RESTORE display this message if you specify the same disk as both source and target.

`SOURCE diskette bad or incompatible`
DISKCOPY displays this message if it cannot read your source diskette.

`Source does not contain backup files`
RESTORE displays this message if you attempt restore files from a disk that was not used with BACKUP.

`Source path required`
You did not specify a source path for the REPLACE command.

`Specified COMMAND search directory bad`
MS-DOS cannot find the command processor in the directory specified by the SHELL command in your CONFIG.SYS file. Restart your computer after editing your CONFIG.SYS file.

`Specified drive does not exist,`
`or is non-removable`
DISKCOMP and DISKCOPY display this message if you use these commands with a hard disk. You specify a valid diskette drive for both source and target.

`Syntax error`
ATTRIB and FIND display this message if you enter a command incorrectly.

`Syntax errors in GRAPHICS profile`
GRAPHICS displays this message if the GRAPHICS profile file contains an invalid command.

`Target cannot be used for backup`
BACKUP displays this message if doesn't recognize the format of the target disk you specify.

`Target diskette bad or incompatible`
DISKCOPY displays this message if doesn't recognize the format of the target disk you specify.

`Target diskette may be unusable`
DISKCOPY displays this message after completing a copy if it is unable to duplicate the source diskette correctly. A copy-protection format used by the source diskette may be the problem.

`The last file was not restored`
RESTORE displays this message if it is unable to restore the last file on the backup disk. Make sure there is enough room on the target disk.

`The only bootable partition on drive 1 is already marked active`
FDISK displays this message if you try to make an MS-DOS partition active when an active partition already exists.

`Too many drive entries`
You tried to use FASTOPEN with more than four hard disks.

`Too many extent entries`
FASTOPEN displays this message if you specify an *m* parameter larger than 999.

`Too many file/directory entries`
FASTOPEN displays this message if you specify an *n* parameter larger than 999.

`Too many files open`
EDLIN and LABEL display this message if they cannot complete a task because not enough file handles are available. Increase the number of files specified in the FILES= command in your CONFIG.SYS file. Then restart your computer.

`Too many name entries`
FASTOPEN displays this message if the total number of entries specified for a drive is greater than 999.

`Too many open files`
BACKUP, FC, RESTORE, and XCOPY display this message if they cannot complete a task because not enough file handles are available. Increase the number of files specified in the FILES= command in your CONFIG.SYS file. Then restart your computer.

`Top level process aborted, cannot continue`
You chose **A** (Abort), and the error is unrecoverable.

`Track 0 bad - disk unusable`
FORMAT displays this message if it encounters bad sectors on the first track of a diskette. Use a different diskette.

`Unable to access Drive x`
FDISK cannot access the specified drive. Restart your computer.

`Unable to create directory`
MKDIR and XCOPY display this message if they cannot create the directory you specify. This can happen if the directory name is the same as an existing file or directory name, or if you are trying to add a directory to a root directory that already has 512 entries.

`Unable to create KEYB table in resident memory`
KEYB was unable to create a country-specific table for the keyboard code specified. You may not have enough memory available.

`Unable to perform refresh operation`
PRINTER.SYS does not have a copy of the code page in memory to download to the printer.

`Unable to reload with profile supplied`
You already loaded GRAPHICS and did not allocate enough memory to load a second profile. Start your computer again.

`Unable to shift Screen`
MODE is unable to shift the display further.

`Unable to write BOOT`
FORMAT is unable to write the boot record on the specified disk. Retry with a different disk.

`Unexpected DOS Error` *n*
REPLACE encountered an unexpected error. The MS-DOS error number is *n*.

`Unrecognized command in CONFIG.SYS`
Your CONFIG.SYS file contains an invalid command. Restart your computer after editing CONFIG.SYS.

`Unrecoverable error in directory`
`Convert directory to file (Y/N)?`
CHKDSK displays this prompt if a directory is no longer usable because it contains too much invalid information. If you press **Y**, CHKDSK converts the directory to a file, allowing you to fix or delete it.

`Unrecoverable read error on drive` *x*
`side` *n*`, track` *n*
DISKCOMP displays this message if it cannot read the indicated diskette track after four attempts.

`Unrecoverable read (or write) error on drive` *x*`:`
MS-DOS is unable to read or write to the specified device. Make sure the diskette is properly inserted. Then press **R** (Retry). If the error persists, press **A** (Abort).

`usage: fc [/a] [/b] [/c] [/l] [/lb n] [/w]`
`[/t] [/n] [/NNNN] file1 file2`
You entered an incorrect switch in an FC command.

`Volume label does not match`
FDISK displays this message if you try to delete a logical drive and do not enter the current volume label correctly.

`Warning - directory full`

RECOVER cannot complete a task because your root directory is full. Delete some files and retry.

`Warning! Diskette is out of sequence`
`Replace diskette or continue if okay`
`Strike any key when ready`

RESTORE displays this message if you do not restore diskettes in the sequence in which they were backed up.

`Write failure, diskette unusable`

An MS-DOS critical error occurred when SYS was writing data to the disk. Rerun SYS with a different disk.

`Write fault error writing drive x:`

MS-DOS is unable to write to the specified device. Be sure the disk is properly inserted. Then press **R** (Retry). If the error persists, press **A** (Abort).

`Write protect error`
`Format terminated`

You tried to format a write-protected diskette. Remove the write protection and retry.

Appendix B

International Keyboard Layouts

Introduction B-3
Belgium B-4
Canada B-4
Denmark B-5
France B-5
Germany B-6
Italy B-6
Latin America B-7
Netherlands B-7
Norway B-8
Portugal B-8
Spain B-9
Sweden/Finland B-9
Switzerland B-10
United Kingdom B-10
United States B-11
Generating Special Characters B-12
Entering Characters Using Alt Gr B-13

Introduction

MS-DOS includes commands that allow you to select international keyboard layouts for your U.S. keyboard.

After loading an international keyboard command, you can use your United States keyboard to generate most of the corresponding international characters. See the KEYB command in Chapter 4 for details on the correct use of these commands to change the keyboard layout.

The layouts for the individual international keyboards are illustrated on the following pages. Use these illustrations to locate equivalent keys on your U.S. keyboard.

To reload the United States keyboard layout after using another one, press **Ctrl Alt F1**. To reselect the international keyboard layout you were using, press **Ctrl Alt F2**.

Belgium

Canada

Denmark

France

Germany

Italy

Latin America

Netherlands

Norway

Portugal

Spain

Sweden/Finland

Switzerland

United Kingdom

United States

Generating Special Characters

With some of the international keyboards, you can generate special characters with accents or other special marks above or below certain letters. To produce the characters in the following lists first press the key for the special mark; then press the letter key. After you press the letter key, the letter with the special mark appears on your screen.

For example, after you use the KEYB command to select the French keyboard, if you press the ^ key and then **a**, the character â appears on your screen.

There are no special character sequences for the United Kingdom keyboard or the Italian keyboard.

> **Note**
> These sequences are sometimes called dead key combinations.

Use this method to produce the following special characters:

Belgium (AT Keyboards: See France)
Belgium (Enhanced Keyboard)
850: áÁéÉíÍóÓúÚ àÀèÈìÌòÒùÙ äÄëËïÏöÖüÜÿ âÂêÊîÎôÔûÛ ñÑãÃõÕ

Canadian French (AT and Enhanced Keyboards)
863: éÉóú àÀèÈùÙ ëËïÏüÜ âÂêÊîÎôÔûÛ çÇ
850: áÁéÉíÍóÓúÚýÝ àÀèÈìÌòÒùÙ äÄëËïÏöÖüÜÿ âÂêÊîÎôÔûÛ çÇ

Finland and Sweden (AT Keyboard)
437: áéÉíóú àèìòù äÄëïöÖüÜÿ âêîôû
850: áÁéÉíÍóÓúÚýÝ àÀèÈìÌòÒùÙ äÄëËïÏöÖüÜÿ âÂêÊîÎôÔûÛ

Finland and Sweden (Enhanced Keyboard)
437: Same combinations as AT keyboard plus ñÑ
850: Same combinations as AT keyboard plus ñÑ

France (AT and Enhanced Keyboards)
437: äÄëïöÖüÜÿ âéîôû
850: äÄëËïÏöÖüÜÿ âÂîÎôÔûÛ

Germany (AT and Enhanced Keyboards)
437: áéÉíóú àèìòù
850: áÁéÉíÍóÓúÚ àÀèÈìÌòÒùÙ

Latin America (AT keyboard: See Spain)
Latin America (Enhanced Keyboard)
437: áéÉíóú àèìòù äÄëïöÖüÜÿ âêîôû
850: áÁéÉíÍóÓúÚýÝ àÀèÈìÌòÒùÙ äÄëËïÏöÖüÜÿ âÂêÊîÎôÔûÛ

Netherlands (AT keyboard: Uses U.S. AT Keyboard)
Netherlands (Enhanced Keyboard)
437: áéÉíóú àèìòù äÄïöÖüÜÿ âêîôû ñÑ çÇ
850: áÁéÉíÍóÓúÚýÝ àÀèÈìÌòÒùÙ äÄëËïÏöÖüÜÿ âÂêÊîÎôÔûÛ
ãÃõÕñÑ çÇ

Norway and Denmark (AT Keyboard)
865: áéÉíóú àèìòù äÄëïöÖüÜÿ âêîôÔ
850: áÁéÉíÍóÓúÚýÝ àÀèÈìÌòÒùÙ äÄëËïÏöÖüÜÿ âÂêÊîÎôÔüÜ

Norway and Denmark (Enhanced Keyboard)
865: Same combinations as AT keyboard plus ñÑ
850: Same combinations as AT keyboard plus ñÑãÃõÕ

Portugal (AT keyboard: Uses U.S. AT Keyboard)
Portugal (Enhanced Keyboard)
860: áÁéÉíÍóÓúÚ àÀèÈìÌòÒùÙ üÜ âÂêÊôÔ ãÃõÕñÑ
850: áÁéÉíÍóÓúÚýÝ àÀèÈìÌòÒùÙ äÄëËïÏöÖüÜÿ âÂêÊîÎôÔûÛ
ãÃõÕñÑ

Spain (AT and Enhanced Keyboards)
850: áÁéÉíÍóÓúÚ àÀèÈìÌòÒùÙ äÄëËïÏöÖüÜÿ âÂêÊîÎôÔûÛ

Swiss French and Swiss German (AT and Enhanced Keyboards)
437: áéÉíóú àèìòù äÄëïöÖüÿ âêîôû ñÑ
850: áÁéÉíÍóÓúÚýÝ àÀèÈìÌòÒùÙ äÄëËïÏöÖüÜÿ ãÃõÕñÑ

Entering Characters Using Alt Gr

Certain keys on the international keyboards are marked with one or two extra characters. See the illustrations in Appendix C for the location of these keys. You can produce the extra characters by pressing the **Alt Gr** key (the **Alt** key located on the right of the space bar) with the selected character key when you are using a foreign keyboard command.

For example, to input the @ character on a French keyboard, press **Alt Gr** and **0**.

Appendix C

Code Page Tables

Introduction C-3
Code Page 437 (United States-English) C-4
Code Page 850 (Multilingual) C-5
Code Page 860 (Portuguese) C-6
Code Page 863 (Canada-French) C-7
Code Page 865 (Nordic—Norwegian and Danish) C-8

Introduction

The tables in this appendix list the code pages used by MS-DOS. Code page switching is described in Chapter 9.

By using the tables you can determine the hexadecimal value of each character. The first hex digits are in the top row, and the second hex digits are in the first column.

Code Page 437 (United States—English)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0		▲		0	@	P	`	p	Ç	É	á	▒	└	╨	α	≡
1	☺	▼	!	1	A	Q	a	q	ü	æ	í	▓	┴	╤	β	±
2	☻	↕	"	2	B	R	b	r	é	Æ	ó	▓	┬	╥	Γ	≥
3	♥	!!	#	3	C	S	c	s	â	ô	ú	─	├	╙	π	≤
4	♦	¶	$	4	D	T	d	t	ä	ö	ñ	┤	─	╘	Σ	⌠
5	♣	§	%	5	E	U	e	u	à	ò	Ñ	╡	┼	╒	σ	⌡
6	♠	▬	&	6	F	V	f	v	å	û	ª	╢	╞	╓	μ	÷
7	•	↨	'	7	G	W	g	w	ç	ù	º	╖	╟	╫	τ	≈
8	◘	↑	(	8	H	X	h	x	ê	ÿ	¿	╕	╚	╪	Φ	°
9	○	↓	)	9	I	Y	i	y	ë	Ö	⌐	╣	╔	┘	Θ	∙
A	◙	→	*	:	J	Z	j	z	è	Ü	¬	║	╩	┌	Ω	·
B	♂	←	+	;	K	[	k	{	ï	¢	½	╗	╦	█	δ	√
C	♀	∟	,	<	L	\	l	|	î	£	¼	╝	╠	▄	∞	ⁿ
D	♪	↔	-	=	M	]	m	}	ì	¥	¡	╜	═	▌	ϕ	²
E	♫	▲	.	>	N	^	n	~	Ä	Pt	«	╛	╬	▐	ε	■
F	☼	▼	/	?	O	_	o	⌂	Å	ƒ	»	┐	╧	▀	∩	

Code Page 850 (Multilingual)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0		►		0	@	P	`	p	Ç	É	á	▒	└	ð	Ó	-
1	☺	◄	!	1	A	Q	a	q	ü	æ	í	▒	┴	Ð	ß	±
2	☻	↕	"	2	B	R	b	r	é	Æ	ó	▒	┬	Ê	Ô	=
3	♥	!!	#	3	C	S	c	s	â	ô	ú	│	├	Ë	Ò	¾
4	♦	¶	$	4	D	T	d	t	ä	ö	ñ	┤	─	È	õ	¶
5	♣	§	%	5	E	U	e	u	à	ò	Ñ	Á	┼	ı	Õ	§
6	♠	▬	&	6	F	V	f	v	å	û	ª	Â	ã	Í	µ	÷
7	•	↨	'	7	G	W	g	w	ç	ù	º	À	Ã	Î	þ	¸
8	◘	↑	(	8	H	X	h	x	ê	ÿ	¿	©	╚	Ï	Þ	°
9	○	↓	)	9	I	Y	i	y	ë	Ö	®	╣	╔	┘	Ú	••
A	◙	→	*	:	J	Z	j	z	è	Ü	¬	║	╩	┌	Û	•
B	♂	←	+	;	K	[	k	{	ï	ø	½	╗	╦	▉	Ù	¹
C	♀	∟	,	<	L	\	l	|	î	£	¼	╝	╠	▬	ý	³
D	♪	↔	-	=	M	]	m	}	ì	Ø	¨	¡	¢	═	¦	Ý	²
E	♫	▲	.	>	N	^	n	~	Ä	×	«	¥	╬	Ì	¯	▪
F	☼	▼	/	?	O	_	o	⌂	Å	ƒ	»	┐	¤	▬	´	

Code Page 860 (Portuguese)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0		►		0	@	P	`	p	Ç	É	á	▒	└	╨	α	≡
1	☺	◄	!	1	A	Q	a	q	ü	À	í	▒	┴	╤	β	±
2	☻	↕	"	2	B	R	b	r	é	È	ó	▒	┬	╥	Γ	≥
3	♥	!!	#	3	C	S	c	s	â	ô	ú	─	├	╙	π	≤
4	♦	¶	$	4	D	T	d	t	ã	õ	ñ	┤	─	╘	Σ	⌠
5	♣	§	%	5	E	U	e	u	à	ò	Ñ	╡	┼	╒	σ	⌡
6	♠	▬	&	6	F	V	f	v	Á	Ú	ª	╢	╞	╓	μ	÷
7	•	↨	'	7	G	W	g	w	ç	ù	º	╖	╟	╫	τ	≈
8	◘	↑	(	8	H	X	h	x	ê	Ì	¿	╕	╚	╪	Φ	°
9	○	↓	)	9	I	Y	i	y	Ê	Õ	Ò	╣	╔	┘	Θ	•
A	◙	→	*	:	J	Z	j	z	è	Ü	¬	║	╩	┌	Ω	•
B	♂	←	+	;	K	[	k	{	Ì	¢	½	╗	╦	█	δ	√
C	♀	∟	,	<	L	\	l	|	Ô	£	¼	╝	╠	▄	∞	ⁿ
D	♪	↔	-	=	M	]	m	}	ì	Ù	¡	╜	═	▌	ϕ	²
E	♫	▲	.	>	N	^	n	~	Ã	Pt	«	╛	╬	▐	ε	■
F	☼	▼	/	?	O	_	o	⌂	Â	Ó	»	┐	╧	▀	∩	

Code Page 863 (Canada—French)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0		▲		0	@	P	`	p	Ç	É	¦	▒	└		α	≡
1	☺	▼	!	1	A	Q	a	q	ü	È	‗	▒	┴		ß	±
2	☻	↕	"	2	B	R	b	r	é	Ê	´	▒	┬		Γ	≥
3	♥	‼	#	3	C	S	c	s	â	ô	ó	▒	├		π	≤
4	♦	¶	$	4	D	T	d	t	Â	Ë	ú	─	─	╙	Σ	⌠
5	♣	§	%	5'	E	U	e	u	à	Ï	¨	┼	╞	╓	σ	⌡
6	♠	▬	&	6	F	V	f	v	¶	û	¸	╟	╚	╒	µ	÷
7	•	↨	'	7	G	W	g	w	ç	ú	ª	╫	╔	╪	τ	≈
8	◘	↑	(	8	H	X	h	x	ê	¤	î	╢	╩	┘	Φ	°
9	○	→	)	9	I	Y	i	y	ë	Ô	⌐	╖	╦	┌	Θ	∙
A	◙	↑	*	:	J	Z	j	z	è	Ü	¬	╕	╠	▄	Ω	·
B	♂	↓	+	;	K	[	k	{	ï	¢	½	╣	═	▌	δ	√
C	♀	∟	,	<	L	\	l	|	î	£	¼	╗	╬	▐	∞	n
D	♪	↔	-	=	M	]	m	}	ì	¥	¾	╝	╧	▀	φ	2
E	♫	◄	.	>	N	^	n	~	Ä	Û	«	╜	╨	α	ε	■
F	☼	►	/	?	O	_	o	⌂	§	ƒ	»	┐	╤	∩	∩	

Code Page 865 (Nordic—Norwegian and Danish)

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0		▲		0	@	P	`	p	Ç	É	á	▒	└	╨	α	≡
1	☺	▼	!	1	A	Q	a	q	ü	æ	í	▓	┴	╤	β	±
2	☻	↕	"	2	B	R	b	r	é	Æ	ó	▓	┬	╥	Γ	≥
3	♥	‼	#	3	C	S	c	s	â	ô	ú	▓	├	╙	π	≤
4	♦	¶	$	4	D	T	d	t	ä	ö	ñ	─	─	╘	Σ	⌠
5	♣	§	%	5	E	U	e	u	à	ò	Ñ	┼	╞	╒	σ	⌡
6	♠	▌	&	6	F	V	f	v	å	û	ª	╟	╚	╓	µ	÷
7	•	↨	'	7	G	W	g	w	ç	ù	º	╫	╔	╫	τ	≈
8	◘	↑	(	8	H	X	h	x	ê	ÿ	¿	╪	╩	╪	Φ	°
9	○	↓	)	9	I	Y	i	y	ë	Ö	⌐	┘	╦	┘	Θ	∙
A	◙	→	*	:	J	Z	j	z	è	Ü	¬	┌	╠	┌	Ω	·
B	♂	←	+	;	K	[	k	{	ï	¢	½	█	═	▄	δ	√
C	♀	∟	,	<	L	\	l	|	î	£	¼	▌	╬	▌	∞	ⁿ
D	♪	↔	-	=	M	]	m	}	ì	Ø	¡	▐	╧	▐	φ	²
E	♫	▲	.	>	N	^	n	~	Ä	Pt	«	▀	╨	▀	ε	■
F	☼	▼	/	?	O	_	o	⌂	Å	ƒ	¤	┐	╫	┐	∩	

Appendix D

ANSI Escape Sequences

Introduction D-3
Cursor Functions D-4
Modes of Operation D-5
Keyboard Reassignment D-7

Introduction

This appendix explains how the ANSI escape sequences are defined for MS-DOS. This appendix also includes examples of how to use ANSI escape sequences.

An ANSI escape sequence is a series of characters (beginning with an escape character or keystroke) that you can use to define functions for MS-DOS. Specifically, you can reassign keys, change graphics functions, and affect cursor movement.

The following conventions apply to the ANSI escape sequences described here:

- The default value is used when you do not specify an explicit value or a value of zero.
- *Pn* represents a numeric parameter. This is a decimal number specified with ASCII digits. Do not type the P.
- *Ps* represents a selective parameter. This is any decimal number that identifies a subfunction. You may select multiple subfunctions by separating the parameters with semicolons. Do not type the P.
- Be sure to include the bracket because it is a necessary part of the escape sequence.
- Letters included in ANSI escape sequences are case-specific. That is, the ANSI.SYS driver distinguishes lowercase letters from uppercase letters. Be sure to enter letters exactly as they are listed in the following tables.

Cursor Functions

The following table lists the escape sequences that affect the cursor position on the screen.

Name/Format	Description
Cursor Position (CUP) **Esc**[*Pn*;*Pn*H Horizontal and Vertical Position (HVP) **Esc**[*Pn*;*Pn*F	CUP and HVP move the cursor to the position specified by the parameters. The first parameter specifies the line number, and the second parameter specifies the column number. All default values are 1. When you do not specify the parameters, the cursor moves to the home position.
Cursor Up (CUU) **Esc**[*Pn*A	This sequence moves the cursor up one line without changing columns. The value of *Pn* determines the number of lines to move. The default value for *Pn* is 1. There is no effect if the cursor is already on the top line.
Cursor Down (CUD) **Esc**[*Pn*B	This sequence moves the cursor down one line without changing columns. The value of *Pn* determines the number of lines to move. The default value for *Pn* is 1. There is no effect if the cursor is already on the bottom line.
Cursor Forward (CUF) **Esc**[*Pn*C	This sequence moves the cursor forward one column without changing lines. The value of *Pn* determines the number of columns to move. The default value for *Pn* is 1. There is no effect if the cursor is already in the far right column.
Cursor Backward (CUB) **Esc**[*Pn*D	This sequence moves the cursor back one column without changing lines. The value of *Pn* determines the number of columns to move. The default value for *Pn* is 1. There is no effect if the cursor is already in the far left column.
Device Status Report (DSR) **Esc**[6n	The console driver outputs a CPR sequence (see below) on receipt of the DSR escape sequence.

Name/Format	Description
Cursor Position Report (CPR) **Esc[***Pn*;*Pn*R (from console driver to system)	This sequence reports current cursor position via standard input. The first parameter specifies the current line and the second parameter specifies the current column.
Save Cursor Position (SCP) **Esc[s**	This sequence saves the current cursor position. The RCP sequence (see below) restores this cursor position.
Restore Cursor Position (RCP) **Esc[u**	This sequence restores the cursor position to the value it had when the console driver received the SCP sequence.
Erase Display (ED) **Esc[2J**	This sequence erases the screen, and the cursor goes to the home position.
Erase Line (EL) **Esc[K**	This sequence erases from the cursor to the end of the line (including the cursor position).

Modes of Operation

The following table lists the escape sequences that affect screen graphics.

Name/Format	Description
Set Graphics Rendition (SGR) **Esc[***Ps*;...;*Ps*m	This sequence invokes the graphic functions specified by the SGR parameters. The graphic functions remain in effect until the next occurrence of an SGR escape sequence.
Set Mode (SM) **Esc[=***Ps*h **Esc[=h** **Esc[=Øh** **Esc[?7h**	This sequence changes the screen width or type.
Reset Mode (RM) **Esc[=***Ps*l **Esc[=l** **Esc[=Øl** **Esc[?7l**	Parameters for RM are the same as for SM, except that parameter 7 resets the wrap at the end of line mode.

The following table describes the parameters and functions for the Set Graphics Rendition (SGR) escape sequences. Your display adapter may not support all these functions.

Parameter	Function
0	All attributes off (normal white on black)
1	Bold on
2	Faint on
3	Italic on
4	Underscore on (monochrome displays only)
5	Blink on
6	Rapid blink on
7	Reverse video on
8	Concealed on (ISO 6429 standard)
30	Black foreground (ISO 6429 standard)
31	Red foreground (ISO 6429 standard)
32	Green foreground (ISO 6429 standard)
33	Yellow foreground (ISO 6429 standard)
34	Blue foreground (ISO 6429 standard)
35	Magenta foreground (ISO 6429 standard)
36	Cyan foreground (ISO 6429 standard)
37	White foreground (ISO 6429 standard)
40	Black background (ISO 6429 standard)
41	Red background (ISO 6429 standard)
42	Green background (ISO 6429 standard)
43	Yellow background (ISO 6429 standard)
44	Blue background (ISO 6429 standard)
45	Magenta background (ISO 6429 standard)
46	Cyan background (ISO 6429 standard)
47	White background (ISO 6429 standard)
48	Subscript
49	Superscript

The following table lists the parameters and functions for the Set Mode (SM) and Reset Mode (RM) escape sequences.

Your display adapter may not support all these modes.

Parameter	Function
0	40 x 25 black and white
1	40 x 25 color
2	80 x 25 black and white
3	80 x 25 color
4	320 x 200 color
5	320 x 200 black and white
6	640 x 200 black and white
7	Wrap at end of line
14	640 x 200 color
15	640 x 350 mono
16	640 x 350 color
17	640 x 480 color
18	640 x 480 color
19	320 x 200 color

Keyboard Reassignment

Although not part of the ANSI 3.64-1979 or ISO 6429 standard, the following keyboard reassignments are compatible with these standards.

The control sequence is one of the following:

- **Esc**`[`*Pn*`;`*Pn*`;...`*Pn*`p`
- **Esc**`["`*string*`";p`
- **Esc**`[`*Pn*`;"`*string*`";`*Pn*`;`*Pn*`;"`*string*`";`*Pn*`p`
- any other combination of strings and decimal numbers.

The final code in the control sequence (p) is reserved for private use by the ANSI 3.64-1979 standard.

The first ASCII code in the control sequence defines which code is mapped. The remaining numbers define the sequence of ASCII codes generated when this key is intercepted. Note that there is one exception: if the first code in the sequence is zero (NUL), then the first and second codes make up an extended ASCII redefinition.

To disable the reassignment of extended key values on enhanced keyboards, use the following escape sequence:

```
Esc[Øq
```

To enable the reassignment of extended key values on enhanced keyboards (which is equivalent to using the /X switch when installing the ANSI.SYS driver), use the following syntax:

```
Esc[1q
```

Examples

To reassign the Q and q key to the A and a key (and vice versa), use the control sequence:

`Esc[65;81P`	A becomes Q
`Esc[97;113p`	a becomes q
`Esc[81;65p`	Q becomes A
`Esc[113;97p`	q becomes a

To reassign the **F10** key to a DIR command followed by a carriage return, use the control sequence:

```
Esc[Ø;68;"dir";13p
```

The 0;68 is the extended ASCII code for the **F10** key; 13 decimal is a carriage return.

Index

A

Accented characters, B-12
Active partition, 4-78
Allocation unit, 1-10, 4-93
ANSI escape sequences, D-3–8
ANSI.SYS device driver, 3-8, 6-15, 6-31, D-3
Append, EDLIN, 8-31, 8-34
APPEND command, 2-11, 3-6, 4-5–9
Application programs,2-3, 6-5–6, 6-10, 6-15, 6-19
Archive bit, 5-20
ASCII
  characters, 4-94–95
  files, comparing, 5-17–20
  mode, 4-36
Assemble, DEBUG, 8-6, 8-9
Assembly language programs, 8-52
ASSIGN command, 3-7, 4-10–11
ATTRIB command, 3-5, 4-12–14, 5-4, 5-20
Attributes, changing, 5-20–22, 5-31–34
AUTOEXEC.BAT file, 3-17, 7-5–6
AUX, 2-4, 4-39

B

Backing up
  disks, 4-15–19
  files, 4-15–20, 5-8–11
  hard disk files, 4-18
  log file, 4-17
BACKUP command, 3-5, 4-15–20
Backup files, 4-15–20, 5-8–11
BAK files, 8-30–31
Batch file(s), 3-15–18
  calling, 7-10
  comments, 7-4
  creating, 3-16–18, 7-3–5
  executing, 3-16
  jumping, 7-15
  labels, 7-15
  naming, 7-3
Baud rates, 4-119
BIN files, 8-48
Binary
  files, comparing, 5-17–20
  format, 8-48
  mode, 4-36
BIOS, 6-5
Block devices, identifying, 6-7
Booting system, 1-3–4, 4-72
Boot-up drive, 4-158
BREAK command, 3-3, 3-9, 4-20, 6-8
Buffers, allocating, 6-3
BUFFERS command, 6-3, 6-9–11

C

CALL command, 3-3, 3-10, 7-10
CD command, 2-13, 3-3, 3-7, 4-23–24
Central processing unit (CPU), 1-5
Changing file attributes, 5-4, 5-20–21
Character
  codes, 1-15
  sets, 4-21
CHCP command, 3-3, 3-9, 4-21–22, 9-5, 9-9
CHDIR command, 2-13, 3-3, 3-7, 4-23–24
Check disk, 3-4, 4-25–27, 5-25–27

CHKDSK command, 3-4, 4-25–27, 5-25
Clear monitor screen, 3-3, 3-9, 4-28
CLOCK$ device, 2-4
CLS command, 3-3, 3-9, 4-28
Clusters, 1-10
CMOS clock, setting, 4-41
Code page(s), 4-21–22, 4-105, 4-124–126, 6-12–16
  4201.CPI file, 4-126, 9-5
  4208.CPI file 4-126, 9-5
  5202.CPI file 4-126, 9-5
  changing configuration, 3-9
  character sets, 4-126
  commands, 4-126, 5-38–42
  information (CPI) files, 4-126
  list, 5-39
  preparing, 5-39
  refreshing, 4-124, 4-127, 5-41
  selecting, 4-127, 5-38, 5-40
  status, 4-127, 5-38, 5-41
  switching, 4-124–125, 5-39, 6-12, 9-3
  tables, 9-4, C-3–9
CODEPAGE /STATUS, 9-6
CODEPAGE PREPARE, 9-6, 9-8
CODEPAGE REFRESH, 9-6, 9-9
CODEPAGE SELECT, 9-6, 9-9
Color/graphics adapter, 4-94–99
COM files, 8-5, 8-48
COM1 (2, 3, 4), 2-4, 4-39, 4-121, 5-36
Command
  echoes, 7-8, 7-11–12
  filter, 3-20
  format, 3-11–14
  parameters, 3-12
  piping, 3-20–21
COMMAND command, 4-29–31
Command processor
  exiting, 4-57–58
  loading, 4-29–31
  prompt, 1-3
  secondary, 4-29
  switches, 3-14
  top-level, 4-29
Commands
  batch, 3-15–18, 7-10–20
  configuration, 6-3–53
  entering, 3-11, 3-14–15
  external, 3-3
  format, 3-11–12
  internal, 3-3
  programming, 8-3–69
Comment lines, 7-4, 7-19
Compare, DEBUG, 8-6, 8-10
Compare files, 5-17–20
Comparing
  disks, 4-47–50, 5-27–28
  files, 4-32–34, 5-17–20
COMP command, 3-5, 4-32–34
Computer reset, 1-5–6
COMSPEC variable, 4-29
CON, 2-4, 3-16, 4-39, 5-39
Conditional
  parameters, 7-16
  processing, 7-16
CONFIG.SYS file, 4-60, 4-130, 4-162, 5-13, 6-3–5
Configuration
  commands, 6-3–52
  file, 6-3–52
Console device, 2-4, 3-16
Ctrl Alt Delete, 1-5, 1-13
Ctrl Break, 1-5, 1-13
Ctrl C, 1-5, 1-13
Control keys
  functions, 1-12

COPY command, 2-6, 3-3, 3-6, 4-35–38
see also BACKUP, XCOPY
Copy, EDLIN, 8-31, 8-35
Copying
directories, 5-22–24
diskettes, 3-4, 4-51–53, 5-26–27
files, 2-6, 3-3, 3-6, 4-35–38, 5-22–24
Country codes, 4-146, 6-12–13
Country information file, 6-12–14
COUNTRY command, 4-162, 5-13, 5-24, 6-3, 6-12–14, 9-6–7
COUNTRY.SYS, 6-13–14, 4-21
CPU (central processing unit), 1-5
registers, 8-6, 8-17–19
CTTY command, 3-3, 3-9, 4-39–40
Cursor keys
functions, D-4
see also Editing keys
see also EDLIN

D

Data
files, 4-122
displaying, 4-129
protecting, 4-12
sorting, 4-156
redirecting, 3-18–20
DATE command, 3-3, 3-9, 4-41–42
Date, format, 4-41–42
DEBUG command, 3-10, 8-4–28
address, 8-7
command format, 8-5–9
commands, 8-6, 8-9–28
double-precision option, 8-5
hyphen prompt, 8-4
parameters, 8-6
range, 8-8
registers, 8-18
starting, 8-4
TESTBUG.COM, 8-24–28
(A)ssemble, 8-6, 8-9
(C)ompare, 8-6, 8-10
(D)ump, 8-6, 8-10–11
(E)nter, 8-6, 8-11
(F)ill, 8-6, 8-11
(G)o, 8-6, 8-12
(H)ex, 8-6, 8-12
(I)nput, 8-6, 8-13
(L)oad, 8-6, 8-13
(M)ove, 8-6, 8-13
(N)ame, 8-4, 8-6, 8-14–15
(O)utput, 8-6, 8-15
(P)roceed, 8-6, 8-16
(Q)uit, 8-6, 8-16
(R)egister, 8-6, 8-17–19
(S)earch, 8-6, 8-19
(T)race, 8-6, 8-19–20
(U)nassemble, 8-6, 8-20–21
(W)rite, 8-6, 8-21
(XA) Allocate expanded memory, 8-6, 8-22
(XD) Deallocate expanded memory, 8-6, 8-22
(XM) Map expanded memory, 8-6, 8-23
(XS) Get expanded memory status, 8-6, 8-23
Debugging tools, 8-4–28
Default drive, 1-4
changing, 1-4
DEL command, 2-6, 3-3, 3-6, 4-43–44
Delete, EDLIN, 8-31, 8-36–37
Deleting
directories, 2-12, 3-7
files, 2-6, 3-3, 3-6, 4-43–44
partitions, 4-79–82
Delimiters, 3-14

DEVICE command, 6-15
Devices, 3-19
  block, 6-7
  code pages, 4-107
  drivers, 6-30–53
  installing, 6-3, 6-15–16
  identifiers, 6-7
  names, 2-4
  preparing, 4-114–128
DIR command, 2-6, 2-11–12, 3-3, 3-7, 4-45–46
Directories, 2-7–14
  changing, 2-13, 3-3, 3-7, 4-23–24
  creating, 2-12
  default, 2-9
  deleting, 2-12
  displaying, 2-13, 4-163–164
  joining, 4-101–103
  maximum number, 2-8
  naming, 2-8
  pathnames, 2-8–11
  removing, 2-12
  root, 2-7–8
  subdirectories, 2-7
Disk
  buffers, 6-9–11
  drive descriptor, 1-5, 2-5
  status, 4-25–27
Disk Compare, 3-4
DISKCOMP command, 3-4, 4-47–50
DISKCOPY command, 3-4, 4-51–53, 5-4, 5-26
Diskette(s)
  backing up, 5-8–11
  checking, 5-25–27
  comparing, 4-47–50, 5-27–28
  compatibility, 1-7
  copying, 3-4, 4-51–53, 5-26–27
  double-sided, double-density, 1-6
  formatting, 1-9–12, 3-4, 4-87–93, 5-29–30
  fragmented, 4-52
  high density, 1-6
  incompatibility, 1-7
  initializing, 4-87–93
  labelling, 4-108–9
  recovery, 4-140–141
  tracks, 1-6
  utilities, 3-4–5
Diskette drives, 1-6
  diskette incompatibilities, 1-7
  drive incompatibilities, 1-7
  installing, 6-30
  using single, 1-8
  using two, 1-7
Disk Utilities, 3-4–5, 5-4, 5-25–30
Display
  echoes, 3-3, 3-10, 7-8, 7-11–12
  monitor, *see Monitor*
  parameters, 4-142
Displaying
  files, 4-165
  version number, 4-166
  volume label, 4-168
Displaymode parameter, 4-122
DISPLAY.SYS device driver, 6-15, 6-32–33, 9-6
DOSSHELL command, 3-8, 4-54
Drive(s)
  assignments, 4-10–11
  changing, 1-4
  compatibility with diskettes, 1-6
  default, 1-4
  descriptor, 1-5, 2-5
  incompatibility, 1-7
  joining, 4-101–103
  letter, 2-10
  substituting, 3-7, 4-158–159
  single, 1-8–9
  two, 1-7–8

DRIVER.SYS, 3-9, 6-15, 6-34–36
DRIVPARM command, 6-17–18
DU, 5-4
Dump, DEBUG, 8-6, 8-10–11

E

ECHO command, 3-3, 3-10, 7-8, 7-11–12
Editing
  keys, 1-13–15
  text, 8-3–47
EDLIN command, 3-11, 8-29–47
  asterisk (*), 8-30
  commands, 8-31, 8-34–47
  options, 8-31–32
  period (.), 8-32
  pound sign (#), 8-32
  prompt, 8-30
  question mark (?), 8-32
  special features, 8-33
  starting, 8-30
  switches, 8-29
  (A)ppend, 8-31, 8-34
  (C)opy, 8-31, 8-35
  (D)elete, 8-31, 8-36
  (E)nd, 8-31, 8-37–38
  (I)nsert, 8-31, 8-38–39
  (L)ist, 8-31, 8-39–41
  (M)ove, 8-31, 8-41
  (P)age, 8-31, 8-41
  (Q)uit, 8-31, 8-42
  (R)eplace, 8-31, 8-42–45
  (S)earch, 8-31, 8-45–46
  (T)ransfer, 8-31, 8-46–47
  (W)rite, 8-31, 8-47
EEMM286.SYS device driver, 6-15, 6-37–39
EEMM386.SYS device driver, 6-15, 6-40–41
EGA monitor, 9-5
EGA.CPI file, 4-126, 9-5
End, EDLIN, 8-31, 8-37–38
Enter, DEBUG, 8-6, 8-11
Environment, 4-29–30
  current, 3-8
  setting, 4-152–153
  size, 4-30
  strings, 7-5
  variables, 7-7–8
ERASE command, 3-3, 3-6, 4-55–56
Error messages, A-3–42
ERRORLEVEL, 7-16
Escape codes, D-3–8
EXE files, 8-5, 8-48–49, 8-50
EXE2BIN command, 3-11, 8-48–49
Execution, suspending, 1-13
EXIST, 7-16
EXIT command, 3-3, 3-9, 4-57–58
Expanded
  memory, 6-6, 6-37–41
  memory specification (EMS), 6-37–41
Extended memory, 6-5–6, 6-37–41
Extended partition, 4-75–78
  creating, 4-75
  deleting, 4-79–82
  logical drives, 4-75–78
External commands, 3-3, 4-3

F

FASTOPEN command, 3-5, 4-59–61, 6-4
FCBS command, 6-19–20
FC command, 3-5, 4-62–70, 5-18
FDISK command, 3-5, 4-71–82, 5-18
File attributes, 5-20–22

File(s)
ASCII mode, 4-36
AUTOEXEC.BAT, 3-17, 7-5–6
backing up, 5-8–11
batch, 3-15–18, 7-3–5
binary, 8-48
comparing, 5-17–20
control blocks (FCBs), 6-3, 6-6
copying, 2-6, 3-3, 3-6, 4-35–38, 5-22–24
deleting, 2-6, 3-3, 3-6, 4-43–44
display attributes, 3-5
displaying, 2-6, 2-11–12, 3-3, 3-7, 4-165
editing, 3-11
erasing, 2-6, 3-3, 3-6, 4-43–44
executable, 4-4, 8-5, 8-48–49, 8-50
extension, 2-3
handles, 6-6
listing, 2-6, 2-11–12, 3-3, 3-7, 4-45–46
locating, 4-4
maintaining, 2-6
managing, 2-3–7
merging, 4-37
naming, 2-3
opening, 6-6
printing, 3-6, 4-134–136
protecting, 2-6
recovering, 4-140–141
renaming, 3-6
replacing, 4-142–144, 5-15–17
restoring, 4-145–147, 5-11–15
sharing, 3-6
sort, 3-6
verifying, 3-10
FILES, 6-3, 6-21
Filename(s), 2-3–4
File Utilities, 5-4, 5-8–24
Fill, DEBUG, 8-6, 8-12
FIND command, 3-5, 4-85–86
FOR command, 3-3, 3-10, 7-13–14
Format
binary, 8-48
diskettes, 1-9–12, 5-29–30
hard disk, 3-5, 5-29–30
partitioning, 4-71–82
FORMAT command, 1-9–12, 3-4, 4-87–93, 5-29
Function keys, 1-13–15

G

Go, DEBUG, 8-6, 8-12
GOTO command, 3-3, 3-10, 7-15
GRAFTABL command, 1-16, 3-8, 4-94–95, 9-7
GRAPH24 command, 3-8, 4-96
Graphics
mode, 3-8
printing, 3-8
table, 3-8
GRAPHICS command, 3-8, 4-97–99

H

Hard disk, 1-3
backing up files, 3-5
checking, 4-25–27
cylinders, 4-25–27
FDISK, 3-5
formatting, 3-5, 4-87–93
initializing, 4-71–72
partitioning, 3-5
restoring files, 3-5
Hardware interrupts, 6-27
Help, 7
HELP command, 3-8, 4-100, 5-43
Hex, DEBUG, 8-6, 8-12
HIMEM.SYS, 6-15, 6-42–43

I

IF command, 3-3, 3-10, 7-16–17
Input
  DEBUG, 8-6, 8-13
  redirecting, 3-18
Input/output device, 3-18
  changing, 3-18
Insert, EDLIN, 8-31, 8-38–39
Internal commands, 3-3, 4-3
International keyboards, 1-16
  selecting, 1-16
  special characters, generating, 1-15–16, B-11–14

J

JOIN command, 3-7, 4-101–103

K

KEYB command, 1-16, 3-8, 4-104–107, 6-4, 9-5, 9-8
Keyboard, 1-12–16
  code, 1-16, 4-105
  control, 6-31
  control key functions, 1-13
  country, 1-16
  file, 4-104
  function keys, 1-13–15
  generating special characters, 1-16–17
  layout, 4-105, B-3–11
  reassignment, D-7–8
  replacing, 4-105
  setting international, 1-16, B-3–11
  special characters, B-11–14

L

LABEL command, 3-5, 4-108–109
Labels
  batch files, 7-15
  volume, 1-10–12
LASTDRIVE command, 6-3, 6-23
LCD.CPI, 4-126, 9-5
LIM-EMS, 6-6, 6-37–41
LINK command, 3-11, 8-50–69
  aborting, 8-55
  aligning segments, 8-63
  assembly language, 8-52
  combining segments, 8-64
  external references, 8-51
  files, 8-53–54
  fixups, 8-65–67
  groups, 8-52, 8-65
  libraries, 8-57
  list file, 8-57
  object module, 8-56
  parameters, 8-56–57
  run file, 8-50, 8-53, 8-56
  segments, 8-52
  starting, 8-54–56
  switches, 8-57–62
  /CPARMAXALLOC, 8-58
  /DOSSEG, 8-58
  /DSALLOCATE, 8-59
  /EXEPACK, 8-59
  /HELP, 8-59
  /HIGH, 8-60
  /LINENUMBERS, 8-60
  /MAP, 8-60
  /NODEFAULT, 8-60
  /NOGROUP, 8-60
  /NOIGNORE, 8-61
  /OVERLAYINTERRUPT, 8-61
  /PAUSE, 8-61–62
  /SEGMENTS, 8-62
  /STACK, 8-62
  VM.TMP, 8-53–54
List, EDLIN, 8-31, 8-39–41
Load, DEBUG, 8-6, 8-13
Loading MS-DOS, 1-3
Log file, 5-9, 5-11

Logical drives, 4-75–76
deleting, 4-80
printer devices, 4-121
LPT1 (2, 3), 2-4, 4-39, 5-35, 5-37

M

MD command, 2-12, 3-7, 4-112–113
MEM command 3-8, 4-110–111
Memory
allocating, 6-5
expanded, 6-6, 6-37–41
extended, 6-5–6
nonvolatile, 6-5
RAM, 6-5–6
size, 3-8
volatile, 6-5
MENU command, 3-7, 5-3–44
Menu utilities, 4, 3-7, 5-3–44
Merging files, 4-37
Messages, A-3–42
MKDIR command, 2-12, 3-3, 3-7, 4-112–113
MODE command, 3-8, 4-114–126, 5-31, 9-8–9
MODE settings, 5-31–43
Monitor
clearing the screen, 4-28
display format, 5-31–34, 4-114, 4-122
preparing, 4-122-124
MORE command, 3-6, 4-129
Move, DEBUG, 8-6, 8-13
Move, EDLIN, 6-31, 8-41

N

Name, DEBUG, 8-4, 8-6, 8-14–15
NLSFUNC command, 3-8, 4-130–131, 6-4, 9-5, 9-8
NUL, 2-4, 4-39
Numeric keypad, 1-15

O

Object code modules, 3-11
Object modules, LINK, 8-50–53, 8-56
Output, 3-18
Output, DEBUG, 8-6, 8-15
Output, redirecting, 3-18

P

Page, EDLIN, 8-31, 8-41
Parallel printers, 4-116
Parameters
alignment, MODE, 4-122
condition, 7-16
DEBUG, 8-6
display, 6-12
displaymode, MODE, 4-122
LINK, 8-56–57
replaceable, 7-6–7, 7-20
SGR, D-6
SM/RM, D-7
Parity, 4-118–119, 5-36
Partitions, hard disk
active, 4-78
creating, 4-75
deleting, 4-79–82
displaying, 4-83
extended, 4-75–78
logical drives, 4-75–78
primary, 4-75–78
PATH command, 2-11, 3-3, 3-9, 4-132–133
Pathnames, 2-8–11
absolute, 2-9
relative, 2-9
substituting, 4-158–159
PAUSE command, 3-3, 3-10, 7-18
Pipe separator, 3-20–21
Port number, 5-36–37
PRINT command, 3-6, 4-134–136

Printer(s)
buffer, 4-134
change configuration, 3-8, 5-35
device name, 2-4
parallel, 6-29
preparing, 4-116–118, 5-35
PRINTER.SYS, 6-15, 6-44–45
Printing
file, 3-6
output redirection, 4-121–122
PRN, 2-4, 4-39
Proceed, DEBUG, 8-6, 8-16
Programming tools, 8-3–69
PROMPT command, 3-3, 3-9, 4-137–139

Q

Quit, DEBUG, 8-6, 8-16
Quit, EDLIN, 8-31, 8-42

R

RAM, 6-5
RAM disk, *see Virtual disk*
Random access memory, see RAM
RD command, 2-12, 3-7, 4-150
Read-only, 2-7
attribute, set or reset, 4-12–14, 5-20
memory (ROM), 6-5
Reboot, operating system, 1-5
RECOVER command, 3-6, 4-140–141
Redirecting I/O, 3-18–20, 5-37–38, 7-4
Redirecting LPTn, 5-37–38
Redirection symbols, 3-18–20, 5-37–38, 7-4
Refreshing, code pages, 4-127
Register, DEBUG, 8-6, 8-17–19
Registers, 8-18
REM command, 3-3, 3-10, 6-4, 6-24, 7-19
RENAME command, 3-3, 3-6, 4-141
REPLACE command, 3-3, 4-145,–147
Replace Files, 5-15–17
Replace, EDLIN, 8-31, 8-42–45
Replaceable parameters, 7-6–7
Replacing files, 3-3
RESET button, 1-6
Resetting, computer, 1-5
Restart, hardware level, 1-6
RESTORE command, 3-5, 4-145–147
Restore files, 4-145–147, 5-11–15
RMDIR command, 2-12, 3-3, 3-7, 4-150
Root directory, 2-7–10

S

Screen
aligning, 4-122
clearing, 3-3, 3-9
graphics, 4-94–99
Search, DEBUG, 8-6, 8-19
Search, EDLIN, 8-31, 8-45–46
Search path, 2-8–11
Searching for text, 4-85–86
SELECT command, 4-151, 6-4
Serial port
changing configuration, 5-36–37
preparing, 4-118–121
SET command, 3-3, 3-9, 4-152–153
SETATTR, 5-4
SETBCKUP, 5-4
SETFC, 5-4
SETMODE, 5-4

SETPRINT command, 4, 3-7, 5-45–48
SETRPLCE, 5-4
SETRSTOR, 5-4
SETXCOPY, 5-4
SHARE command, 3-6, 4-154–155, 6-4
SHELL, CONFIG.SYS, 6-3, 6-25–26
SHIFT command, 3-3, 3-10, 7-20
SMARTDRV.SYS, 6-6
SORT command, 3-6, 4-156–157, 9-7
Special characters, generating, 1-15–16, B-11–14
Stack frames, 6-27–28
STACKS command, 6-3, 6-27–28
Stop bits, 5-36
Subdirectories, see Directories
SUBST command, 2-11, 3-7, 4-158–159
Substitute, drive letter, 2-11, 3-7, 4-158–159
Switches, 3-14
SYS
  command, 3-5, 4-160
  files, 3-8
System
  booting, 1-3
  startup, 6-4

T

Text
  editor, *see EDLIN*
  files, 4-36
  searching for, 4-85–86
TIME command, 3-3, 3-9, 4-161–162
Time format, 1-4, 4-161
Trace, DEBUG, 8-6, 8-19–20
Transfer, EDLIN, 8-31, 8-47
TREE command, 2-14, 3-7, 4-163–164
TYPE command, 3-3, 3-6, 4-165
Typematic rate, 5-43

U

Unassemble, 8-6, 8-20–21

V

VDISK.SYS device driver, 3-9, 6-5, 6-16, 6-48–50
VER command, 3-3, 3-10, 4-166
VERIFY command, 3-3, 3-10, 4-166
Version number, 3-3, 3-10, 4-167
Video Display, 5-31–34
Virtual disks, 6-5–6
VOL command, 3-3, 3-10, 4-168
Volume
  identification, 1-10
  labels, 1-10–12
  serial number, 1-11

W

Wildcard characters, 2-5
Write, DEBUG, 8-6, 8-21
Write, EDLIN, 8-31, 8-47
Write-protect diskette, 2-6–7

X

XCOPY command, 3-6, 4-169–171, 5-22
XMAEM.SYS, 6-16, 6-51
XMA2EMS.SYS, 6-16, 6-52
XMS, 6-42

MS-DOS® 4.01 Reference Manual

MS-DOS® 4.01 Reference Manual

EPSON®

Epson America, Inc.
2780 Lomita Boulevard
Torrance, CA 90505