Digital Research

DR DOS® 6.0

Operating System for Personal Computers

User Guide

Digital Research®

WE MAKE COMPUTERS WORK

DR DOS® 6.0

User Guide

DISCLAIMER

DIGITAL RESEARCH INC. PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE. Further, Digital Research Inc. reserves the right to revise this publication and to make changes from time to time to the content hereof without obligation of Digital Research Inc. to notify any person of such revision or changes. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This publication could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. Digital Research Inc. may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time.

NOTICE TO USER

This manual should not be construed as any representation or warranty with respect to the software named herein. Occasionally changes or variations exist in the software that are not reflected in the manual. Generally, if such changes or variations are known to exist and to affect the product significantly, a release note or README file accompanies the manual and the distribution disks. In that event, be sure to read the release note or README file before using the product.

TRADEMARKS

Digital Research, the Digital Research logo, DR DOS, MemoryMAX, TaskMAX, ViewMAX, DOSBOOK, SID, and FileLink are trademarks or registered trademarks of Digital Research Inc.

Adobe and PostScript are registered trademarks of Adobe Systems Inc. Hercules is a trademark of Hercules Computer Technology Inc.

IBM, IBM PC, and IBM PC/XT are registered trademarks and PS/2 and AT are trademarks of International Business Machines Corporation. Intel, 8086, 8087, 80286, 386, 386SX, 486 and i486 are registered trademarks of Intel Corporation.

SCAT, CHIPSet, LeAPSet, LeAPSetsx, NeAT and NeATsx are trademarks of Chips and Technologies Inc.

Lotus 1-2-3 is a registered trademark of Lotus Development Corporation. WordStar is a trademark of MicroPro International Corporation.

Microsoft and MS-DOS are registered trademarks and MS-Windows is a trademark of Microsoft Corporation.

Novell, IPX, NetWare, and NET3 are trademarks or registered trademarks of Novell Incorporated.

Compaq is a registered trademark of Compaq Computer Corporation.

PC-Kwik is a registered trademark and Super PC-Kwik is a trademark of Multisoft Corporation. SuperStor and AddStor are trademarks of AddStor, Inc.

All other trademarks are the property of their respective holders and are hereby acknowledged.

COPYRIGHT

Second edition: August 1991

Preface

About the manual

The product described in this manual is DR DOS®, a disk operating system (DOS) which is designed to run on Intel® 8088®, 80286®, 386®, i486®, and compatible microprocessors.

This manual is intended for people who are using DR DOS for the first time, and for those people who are already familiar with DOS but want information on this version.

Before you start using DR DOS, read your manufacturer's handbook, so that you are reasonably familiar with your computer and how it works.

Finding your way around

If you are new to personal computer operating systems, you should read the first four chapters of this manual in order of content. They are written with the new user in mind, and build upon each other. The remaining chapters contain more detailed information on specific uses of, and commands within DR DOS. If you are more experienced, after reading Chapter 1, "Getting started with DR DOS" you may find you want to skip to Chapter 7, "The EDITOR" and continue from there.

Below is a summary of the individual chapters and appendices. Those chapters that people new to PC operating systems should read are indicated with this symbol †.

Structure of the manual

Chapter 1 Getting started with DR DOS†

Chapter 1 begins by explaining certain concepts you should be familiar with before installing and using the DR DOS operating system. It then describes how to install DR DOS and how to take backup copies of your installation diskettes.

Chapter 2 Basic command concepts†

Chapter 2 describes the basic things you need to know about DR DOS commands to get started using them. It explains command structure and how to enter commands.

Chapter 3 Managing your information†

In Chapter 3 you will learn how to organize your data into files and directories for ease of access, how to name files and directories, and how to access them using paths. It also lists some DR DOS commands used to create and work with files and directories.

Chapter 4 Protecting your information†

This chapter contains important information about protecting your data from unauthorized access or accidental loss by using passwords or by backing up information onto diskettes for future reference. It also describes how to secure your entire system.

Chapter 5 Recovering your information

Chapter 5 decribes how you can recover deleted data in DR DOS, and how the "undelete" commands work together.

Chapter 6 Switching tasks

Chapter 6 explains how to install, configure and use TaskMAXTM, a utility that allows you to switch quickly and easily from one application to another. The applications you switch from are suspended exactly as you left them until you return.

Chapter 7 The EDITOR

Chapter 7 explains about the easy-to-use editor within DR DOS that you can use to create text files.

Chapter 8 Redirecting input and output

Instead of your keyboard input being sent to your screen, you can send it directly to a printer or file. You can also filter the output from your computer using a mechanism called piping. For example you may want the output of a program to be the input for another program. Chapter 8 explains how you can do this.

Chapter 9 Grouping commands in batch files

This chapter explains how to create and work with batch files, which enable you to store frequently-used sets of commands in one file.

Chapter 10 DR DOS commands

Chapter 10 contains an alphabetical list of all the DR DOS commands including their full command structure and what they are used for, with examples of usage.

Chapter 11 Customizing your system

At some point in the future you may want to change your system configuration, for example if you add a new printer. Chapter 11 explains how to use the SETUP program to do this. It also describes the two main configuration files: **config.sys** and **autoexec.bat**.

Chapter 12 Using MemoryMAXTM

The term *MemoryMAX* describes the collection of device drivers and commands that are used for memory management under DR DOS. This chapter describes the features offered by the DR DOS memory management tools, and explains how to use them to maximize the memory available to your applications.

Chapter 13 Optimizing your disk

Chapter 13 describes three utilities you can use in DR DOS to get maximum performance from your hard disk: a disk "optimizer", a cache program (Super PC-Kwik®), and a file compression program (SuperStor).

Chapter 14 Setting up code page switching

DR DOS allows you to change between different character sets, so that you can work with documents in different languages using a mechanism called code page switching. This chapter explains how to set up code page switching on your system.

Appendix A Error messages and troubleshooting

Appendix A contains a list of error and system messages generated by DR DOS and its associated utilities. Each message has a simple explanation and where possible a solution. There is also a section on what to do if things go wrong.

Appendix B Character sets for code pages supported by DR DOS

In this appendix is a list of all the different code page character sets supported by DR DOS.

Appendix C Memory

This appendix explains the different types of memory, and how you can use DR DOS to make better use of memory on your computer.

Appendix D The SID debugging tool

This appendix explains the programmers' symbolic debugging tool "SID".

Appendix E Using Microsoft® Windows™

This appendix contains notes about using Microsoft Windows with DR DOS.

Glossary

The glossary contains descriptions of terms found within the manual.

Quick Reference Table

To learn about	Look in
ANSI escape sequences	Chapter 11
Assigning passwords	Chapter 4
Autoexec.bat file	Chapter 11
Batch processing	Chapter 9
Changing your configuration	Chapter 11
Code page switching	Chapter 14
Command syntax and usage	Chapters 2 & 10
Config.sys file	Chapter 11
Copying disks	Chapters 1 & 4
Debugging programs	Appendix D
Directories	Chapter 3
Disk cache	Chapter 13
Disk optimizing	Chapter 13
Disks	Chapter 1
Editing the command line	Chapter 10
Editing text files	Chapter 7
Error messages	Appendix A
File compression	Chapter 13
File recovery	Chapter 5
Files	Chapter 3
Filters and pipes	Chapter 8
Formatting diskettes	Chapters 1 & 4
Installing DR DOS	Chapter 1
Installable device drivers	Chapter 11
Memory management	Chapter 12
Online help and documentation	Chapter 10
Paths	Chapter 3
Redirecting input and output	Chapter 8
SETUP program	Chapter 11
SID debugging program	Appendix D

Quick Reference Table (continued)

To learn about	Look in
Storing your data	Chapter 4
Switching between applications	Chapter 6
System security	Chapter 4
Wildcard characters	Chapter 3

New features of this version of DR DOS

If you are familiar with an earlier version of DR DOS you will find that this version includes some significant advances in DOS technology. The main additions are listed in this section.

Batch file subcommands

The following commands are now supported in batch files as well as **config.sys**: GOSUB, RETURN, and SWITCH. See Chapter 9, "Grouping commands in batch files".

Code page support (Hungarian, Russian, and Turkish)

DR DOS now supports the Hungarian, Russian, and Turkish code pages, and country codes. See the CHCP, KEYB, and MODE commands in Chapter 10, "DR DOS commands". Also see COUNTRY in Chapter 11, "Customizing your system", Chapter 14, "Setting up code page switching", and Appendix B, "Character sets for code pages supported by DR DOS".

Config.sys commands

The following commands have now been added: CLS, CPOS, EXIT, GOSUB, HIBUFFERS, RETURN, SWITCH, and TIMEOUT. See Chapter 11, "Customizing your system".

Disk optimizer (DISKOPT)

The DISKOPT command provides a menu-driven utility to change the arrangement of data on a disk, optimizing the disk's performance. See Chapter 13, "Optimizing your disk".

Help and DOSBOOK™

The following types of online help and online documentation are now available with DR DOS:

- ❑ Help text on all DR DOS commands (both internal and external) by typing the command followed by /? or /H.
- ❑ Online documentation about DR DOS by using the DOSBOOK command. See Chapter 10, "DR DOS commands".

File compare (FC)

The FC command allows you to compare the contents of ASCII text or binary files and display the differences found. If you compare ASCII text files, FC displays the lines you need to insert, delete or replace to make the old file match the new file. If you compare binary files, the differences are displayed in hexadecimal form. See the FC command in Chapter 10, "DR DOS commands".

Security (LOGIN and LOCK)

DR DOS now provides a "secure system" option during installation or configuration that protects your system from unauthorized access, using passwords and a LOGIN screen. Also, *temporary* locking of the system is provided by the LOCK command, so that you can leave your compter unattended without exiting applications or switching off. See Chapter 4, "Protecting your information", and LOCK in Chapter 10, "DR DOS commands".

Moving files and directories (MOVE)

The MOVE command allows you to move files or entire subdirectories to another location on the same disk or another disk. See MOVE in Chapter 10, "DR DOS commands".

Renaming directories (RENDIR)

DR DOS provides the RENDIR command to rename a subdirectory, in addition to the RENAME command for files. See RENDIR in Chapter 10, "DR DOS commands".

PostScript support (SCRIPT)

The SCRIPT command provides PostScript® support for DR DOS by translating all characters in a text file, including those conforming to the Hewlett-Packard LaserJet II standard, to PostScript and sending it to the destination you specify. See SCRIPT in Chapter 10, "DR DOS commands".

Disk cache (Super PC-Kwik)

Super PC-Kwik is a disk cache that also increases the speed at which your system reads information from diskettes and hard disks. See Chapter 13, "Optimizing your disk".

File compression (SuperStor)

The SuperStor program allows you to convert your disk drives into SuperStor drives, compressing data for more efficient storage. Data is then automatically compressed when you write to the drive and decompressed when you read from the drive. See Chapter 13, "Optimizing your disk". See also the descriptions of the SuperStor device drivers SSTORDRV.SYS and DEVSWAP.COM in Chapter 11, "Customizing your system".

Switching between applications (TaskMAXTM)

TaskMAX makes changing between applications quicker and easier. Using it you can switch from one application to another, while the application you leave is *suspended* until you return. You can keep up to 20 applications in the background, and return to them as you require. You can also copy information from one application to another. See Chapter 6, "Switching tasks", and the TASKMAX command in Chapter 10, "DR DOS commands".

File recovery utilities

DR DOS provides a set of commands to help you recover files that you have previously deleted.

- ☐ DELWATCH saves deleted files on the disk even though they appear to be deleted.
- ☐ DELPURGE allows you to remove files saved by DELWATCH.
- ☐ DISKMAP copies and saves the disk's File Allocation Table (FAT).
- ☐ UNDELETE attempts to recover deleted files, using the best method available (DELWATCH or DISKMAP).

See Chapter 5, "Recovering your information", and the UNDELETE command in Chapter 10, DR DOS commands".

UNFORMAT

The UNFORMAT command allows you to recover files from a disk that has been "safe formatted" using the DR DOS FORMAT command. See UNFORMAT in Chapter 10, "DR DOS commands".

Enhanced commands in this version of DR DOS

This section summarizes the enhancements to existing utilities in this version of DR DOS.

CHKDSK

CHKDSK now has a simplified command format. See CHKDSK in Chapter 10, "DR DOS commands".

COMMAND

You can now fix a **command.com** copy permanently in memory and specify a file other than **autoexec.bat** to execute, using the /P switch. You can also force the **command.com** to load in a particluar part of memory (lower, upper, or high) using the new /ML, /MU, and /MH switches. See Chapter 10, "DR DOS commands".

Config.sys commands

The HIDEVICE statement now supports a SIZE option, to specify the amount of memory required for the driver to install in upper memory.

You can now specify a file other than **autoexec.bat** to execute with the /P option, which is used to fix a **command.com** copy in memory. SHELL also supports the new options /ML, /MU, and /MH, to specify the part of memory in which to load the **command.com**.

The ECHO and SET commands have also been enhanced.

See Chapter 11, "Customizing your system".

DIR

DIR now has a switch to display a list in two-column format, /2. See Chapter 10, "DR DOS commands".

DISKCOMP and DISKCOPY

DISKCOMP and DISKCOPY now automatically make use of expanded memory, extended memory and temporary files (in that order) to hold more diskette contents in memory, so that they do not have to access the diskette so many times. This is particulary useful if you have a single disk-drive, because you will not need to change diskettes so often.

You can now use DISKCOPY to store an *image* of a diskette in a file. This *image file* can then be copied to, or compared against (using DISKCOMP), another diskette.

DISKCOMP and DISKCOPY both support two new switches, /A and /M. The /A switch, makes the computer sound a beep when a copy or comparison is complete, or when you need to change diskettes. /M allows you to make multiple copies or comparisons of a disk or image file.

DISPLAY.SYS

DISPLAY.SYS has a new option, [,m], which allows you to configure the number of subfonts per code page that are stored in memory. In previous versions, by default all font sizes were held in memory. Using this new option, you can free up conventional memory so that you have more memory available for your applications. See Chapter 11, "Customizing your system".

EMM386.SYS

EMM386.SYS has a new option, /WINSTD, which forces compatibility with Windows 3.0 in Standard mode. See Chapter 11, "Customizing your system", and Appendix E, "Using Microsoft Windows".

FORMAT

The FORMAT command will now format all types of disk including any device with a suitable device driver. The new /X switch is used to specify that you are formatting a hard disk. Also, FORMAT now performs a "safe format" if the format required matches the existing format of the disk. This means that if by accident you format a disk with data on it, you can use the UNFORMAT command to recover the disk. See Chapter 10, "DR DOS commands".

PROMPT

You can now specify an additional special character ($x) with the PROMPT command to run a command (defined by the PEXEC variable in a SET statement) automatically, every time you return to the DR DOS command prompt. See Chapter 10, "DR DOS commands".

RESTORE

RESTORE now supports all previous versions of DOS BACKUP commands. See RESTORE in Chapter 10, "DR DOS comands".

VDISK.SYS

VDISK.SYS has a new option, /X, which allows you to configure a memory disk in expanded memory. The maximum size of the memory disk you can create has been increased to 32 Mbytes (depending on the type of memory you use for the memory disk). See Chapter 11, "Customizing your system".

XCOPY

The XCOPY command includes a new switch, /L, which allows you to copy a disk label from one disk to another, as well as the files specified.

XDIR

XDIR now supports the /X switch which sorts the directory output by filename extension and the /Y switch which sorts by SuperStor compression ratio. See XDIR in Chapter 10, "DR DOS commands".

For users of other DOS operating systems

If you are new to DR DOS, but have used another DOS operating system, you will find many commands and utilities are familiar. However, DR DOS enhances many of the commands, and eases the use of your computer. Listed here are important and useful features that you may not be aware of.

Built-in help

With all DR DOS commands you can display information about their options by typing the command name followed by /? or /H, for example CHKDSK /?.

Protecting files

With DR DOS you can assign passwords to your files and directories to prevent unauthorized access, using the PASSWORD command.

Displaying and searching for files

You can display files in multiple subdirectories, using the XDIR command.

You can search for a file or multiple occurrences of a file, and display the file's location on your disk using the TREE command.

Transferring files between computers with FileLink™

The FileLink file transfer utility allows programs and data to be easily transferred from one computer to another, using a serial cable; see Chapter 10, "DR DOS commands".

Editing the command line

DR DOS allows you to recall previously entered command lines for re-use. The number of lines that may be recalled can be chosen at installation or reconfigured later using the SETUP program.

Full screen editing

Instead of single line editing (like that provided in other DOS operating systems), DR DOS provides the EDITOR, which enables you to display and edit a full screen of text at a time.

Deleting files

You can delete files in multiple subdirectories and remove empty directories with a single command, using the XDEL command.

When you wish to delete files and be extra sure you do not delete the wrong ones, you can use the ERAQ or DELQ command to cause DR DOS to prompt you for confirmation before deleting each file.

Debugging program

DR DOS has a debugging program called SIDTM which is similar to debugging utilities in other DOS operating systems. It is described in Appendix D, "The SID debugging tool".

Expanded Memory Specification

DR DOS supports the LIM 4.0 Expanded Memory Specification on all Intel 386 and i486-based systems and on IBM® Expanded Memory Adaptors (XMA) or compatible memory boards.

Large hard disk support

DR DOS supports hard disk partitions greater than 32 Mbytes. Disk partitions use an industry-standard partitioning scheme, and support up to 512 Mbyte per partition.

Installing DR DOS

The installation program for DR DOS (INSTALL) provides a series of menu screens to enable you to install DR DOS quickly and easily. Also changing your system configuration later is easy using the SETUP program.

Uninstalling DOS

The DR DOS installation program also provides an option for you to save your old operating system. Then you can remove DR DOS at any time and restore your old system, using the DR DOS UNINSTAL command.

Running DR DOS from ROM

Your computer may have DR DOS in ROM. In this case you will find that DR DOS is already installed on your computer and starts automatically when you power on. Running DR DOS from ROM also increases the amount of memory available for running applications.

Advanced memory management

EMM386.SYS

This is the DR DOS LIM 4.0 driver which converts extended memory into expanded memory on Intel 386, and i486-based computers. EMM386.SYS allows the DR DOS kernel to be located in upper memory, or above 1 Mbyte, to increase the memory available to applications. It also supports the use of upper memory between 640 Kbyte and 1 Mbyte to load device drivers, TSRs (Terminate-and-Stay-Resident programs), and networking software. This can be achieved using the HIDEVICE, HIINSTALL and HILOAD commands. By supporting the Virtual Control Program Interface (VCPI), other 386 applications, like Lotus 1-2-3®, can also run when this driver is loaded. EMM386.SYS is compatible with Microsoft Windows 3.0 in all three modes. See Chapter 11, "Customizing your system". See also Appendix E, "Using Microsoft Windows".

HIDEVICE

HIDEVICE allows you to load device drivers into upper memory, if available, during startup. The EMM386.SYS or HIDOS.SYS device driver must be loaded first; see Chapter 11, "Customizing your system".

HIDOS

This command enables you to relocate operating system data structures into upper memory, if available, during startup. The EMM386.SYS or HIDOS.SYS drivers must be loaded first; see Chapter 11, "Customizing your system".

HIDOS.SYS

This device driver allows users of Intel 80286-based computers to load the DR DOS kernel into the high memory above 1 Mbyte (if the computer has more than 1 Mbyte of memory). It also allows users with computers based on chip sets that support mappable Shadow RAM to access the upper memory between 640 Kbyte and 1 Mbyte in order to load drivers, TSRs (Terminate-and-Stay-Resident programs), and networking software. This can be achieved using the HIDEVICE, HIINSTALL and HILOAD commands; see Chapter 11, "Customizing your system".

HIINSTALL

This command allows utilities, such as CURSOR and PRINT, to be loaded into upper memory, if available, at startup. The EMM386.SYS or HIDOS.SYS drivers must be loaded first; see Chapter 11, "Customizing your system".

HILOAD

HILOAD allows utilities and network drivers such as Novell® IPX™ and NET3™ to be loaded from batch files, or the command line, into upper memory, if available. The EMM386.SYS or HIDOS.SYS device driver must be loaded before this can be used; see Chapters 10, "DR DOS commands" and 11, "Customizing your system".

MEM

The MEM command displays the amount of used and free conventional, extended and expanded memory. It also displays where in memory programs, data and drivers are located, in both text and graphical form; see Chapter 10, "DR DOS commands".

MEMMAX

This command disables newly mapped areas of RAM if they confuse an application; see Chapter 10, "DR DOS commands".

Extending your computer system

You may wish to extend your computer system by adding extra hard disks, memory boards or 3½" diskette drives. You should consult your dealer about what you can use with DR DOS if you are in any doubt.

Assumptions and conventions

What you type in

What you should type in on the screen is shown as follows:

```
DISKCOMP a: b: /V
```

The commands and switches are shown in uppercase, and the remainder of the command line in lowercase for clarity.

Screen messages

Information on screen, like a system message, is shown like this:

```
    Insert diskette to format in drive A:
    and strike a key when ready
```

File entries

Example entries in files, for example an extract from the **autoexec.bat**, are shown as follows:

```
keyb uk
date
time
```

Callouts

Look for callouts!

Callouts in the margins direct your attention to specific items of information.

Warnings, cautions and notes

Warnings, cautions and notes incorporate information separated from the text. Warnings and cautions are used to advise of the risk of loss of data or other problems; warnings are more serious than cautions. Notes contain additional information.

Examples

Note

If your keyboard has separate cursor control keys, marked with arrow keys pointing right, left, up and down, you can use these keys to move the cursor instead.

Caution Use this option with care, as it can lead to problems when used with certain application programs.

WARNING Use the FORMAT command with care! It will erase everything on the diskette to be formatted.

Keys you will use

When you work with DR DOS, certain keys on your keyboard have specific meanings.

The Enter key

The key you use to make the computer carry out an instruction is the "**Enter**" key, sometimes called the "**Return**" key or "**Carriage Return**" key. It often has this symbol ↵ on the keytop. In this manual it is always referred to as the **Enter** key.

The Ctrl and Alt keys

The **Ctrl** (control) and **Alt** (alternative) keys are used to alter the meaning of another key. If you see **Ctrl+A**, you hold down the **Ctrl** key and then type **A** or whatever letter follows the + sign. You use the **Alt** key in the same way.

If you see **Ctrl+Alt+Del,** hold down **Ctrl** and **Alt**, and then press the **Del** key. This restarts DR DOS.

Commands

For clarity this manual shows DR DOS commands in uppercase letters, and filenames and options in lowercase, but DR DOS itself ignores case when it reads the command line.

Every command that you type in DR DOS needs to be followed by the **Enter** key. This is assumed in all chapters after Chapter 1, "Getting started with DR DOS". Sometimes when commands are being carried out, you may be prompted to respond to a query by typing a single letter, Y for YES or N for NO.

For example if you are copying diskettes using the command DISKCOPY, when the diskette has been copied you are asked:

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

Disks and diskettes

Unless otherwise stated in the text, *disk* refers to any kind of disk, hard, floppy or memory.

Floppy disks are called *diskettes* and the term diskette refers to either 5¼" or 3½" diskettes.

Most of the examples in this manual assume that your computer has one hard disk (labeled “drive C”) and one diskette drive (labeled “drive A”). DR DOS works equally well on a computer with twin-diskette drives, computers with a hard disk and one diskette drive, and on computers with more than one hard disk.

Contents

1

Getting started with DR DOS

THIS chapter is your starting point with DR DOS. If you are an experienced personal computer user, you can go straight to section 1.4, "How to install DR DOS". The DR DOS installation procedure is very easy to follow and has full context-sensitive help available at all stages; simply press the **F1** function key for assistance.

If you have no experience of personal computer operating systems, you may find it helpful to read sections 1.1, "Essential concepts", 1.2, "Different disk configurations", and 1.3, "Preparing disks for use" first.

1.1 Essential concepts

This section explains some concepts which you need to be familiar with before you begin using DR DOS.

Memory

Every computer must have memory to store programs and information, and to load and run programs. The type of memory used to hold programs and data while the computer is switched off, is called *storage* memory. The type of memory used to load and run programs, is called *Random Access Memory* (or RAM for short). RAM can be thought of as the computer's workspace. The more complex a task you want to do, the more memory your computer needs.

Types of RAM

There are several types of RAM, including:

- ☐ Conventional (or base).
- ☐ Upper memory.
- ☐ High memory.
- ☐ Extended.
- ☐ Expanded.

Conventional memory

Conventional memory is where you run most of your programs and utilities, and is usually where your operating system is loaded. The maximum amount of conventional memory available is normally 640 Kbytes. However, this barrier can be overcome by using *Extended* memory which is memory above 1 Mbyte.

Expanded memory

Another way of accessing more memory is to use *expanded memory,* otherwise known as EMS or LIM memory after the specifications that define it. Expanded memory enables you to run programs which manipulate large amounts of data, and which would otherwise not run efficiently in conventional memory. Programs have to be designed specifically to access it.

Upper memory

Any memory addressed between 640 Kbytes and 1 Mbyte is known as *upper* memory. Upper memory is 384 Kbytes in size. Some sections of upper memory are used by various parts of your computer's hardware.

High memory

High memory is the first 64 Kbytes of extended memory. Within DR DOS there are several commands that can make use of upper and high memory.

Appendix C, "Memory", has more information on all these memory types.

Disks

In order to hold all the information that you build up after working on a personal computer for some time, as well as the programs you use to help you in your work, you need some form of permanent storage medium. Disks provide that medium.

It is important that you understand the difference between disk and memory storage. The memory in your computer is used for *temporary* storage. This means “in most cases” when you switch your computer off, the data stored in it is lost. Disk drives are designed for *permanent* information storage.

Types of disk

When you start working with DR DOS you will be using two or three types of disk to store information:

❑ Hard disks.

❑ Diskettes.

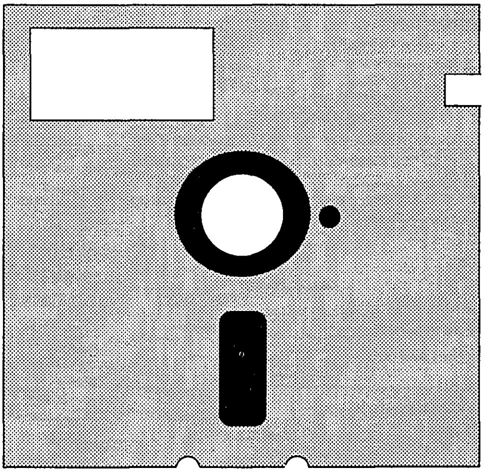
❑ Memory disks.

Hard disks

Hard disks are permanently fixed inside your computer (they are sometimes referred to as *fixed* disks because of this). These disks can store a lot of information, and can access the information from memory much faster than diskettes. Hard disks come in many sizes, typically allowing between 20 and 300 Mbytes to be stored. Hard disks have to be partitioned before they can be used, that is, divided into sections. You then load your operating system into a partition. You can have a partition up to 512 Mbytes in size, and you may have several partitions on one hard disk.

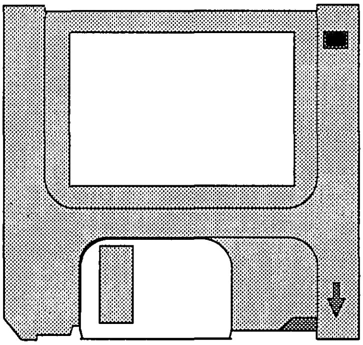
Diskettes

Diskettes (or “floppy disks") are *removable* disks which come in various sizes and capacities. The two main sizes are: 5¼" and 3½". The physical capacity of diskettes ranges from 360 Kbytes to 1.2 Mbytes for 5¼" diskettes, and for 3½" diskettes from 720 Kbytes to 2.88 Mbytes. New blank diskettes need to be formatted before you can use them.



5¼" diskette front view

The 3½" diskettes are more compact than 5¼" diskettes. They can store as much as a 5¼" diskette or even more. They have solid plastic covers and a protective metal shield to prevent dust getting to the diskette surface.



3½" diskette front view

The do's and don'ts of handling diskettes

Do's:

Do Hold diskettes at the corner.

Do Use a felt tip pen to write the diskette label before sticking it onto the diskette, and if necessary remove the old label first.

Do Store them in a suitable container in a safe place.

Do Label diskettes clearly with what is recorded on them, including the name of the owner (if it is important) and the date it was recorded.

Do Store diskettes in their protective envelopes.

Do Take care when inserting diskettes into the diskette-drive. Slide them in without catching or bending them.

Do Store at a temperature between 50 degrees and 125 degrees Fahrenheit (10-52^0 Centigrade).

Don'ts:

Don't Touch the inner diskette surface (the outer diskette cover can be touched).

Don't Keep diskettes near magnetic fields, radiators, heating or cooling ducts, or in direct sunlight.

Don't Place diskettes on top of your monitor.

Don't Let the diskette label adhesive near the exposed inner diskette.

Don't Bend or fold diskettes.

Don't Punch holes in the diskette using staplers or paper clips.

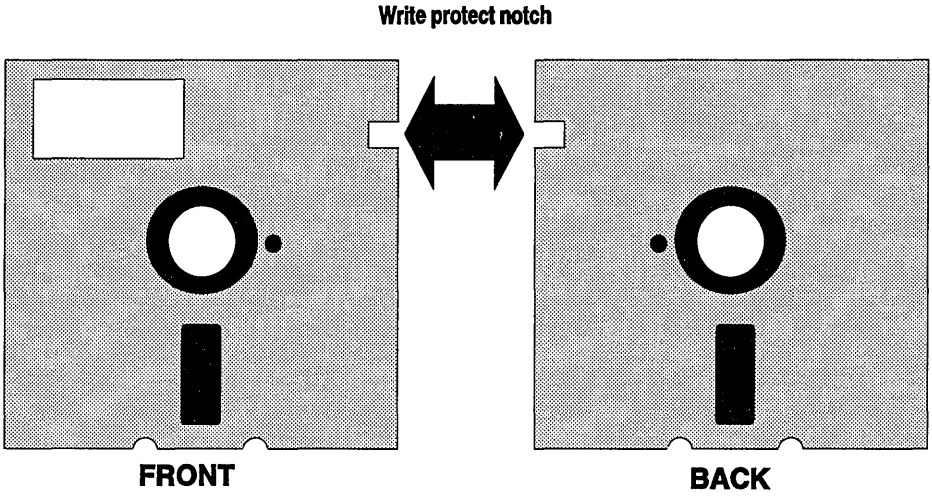
Don't Clean diskettes.

Don't Expose the inner diskette.

Write-protecting diskettes

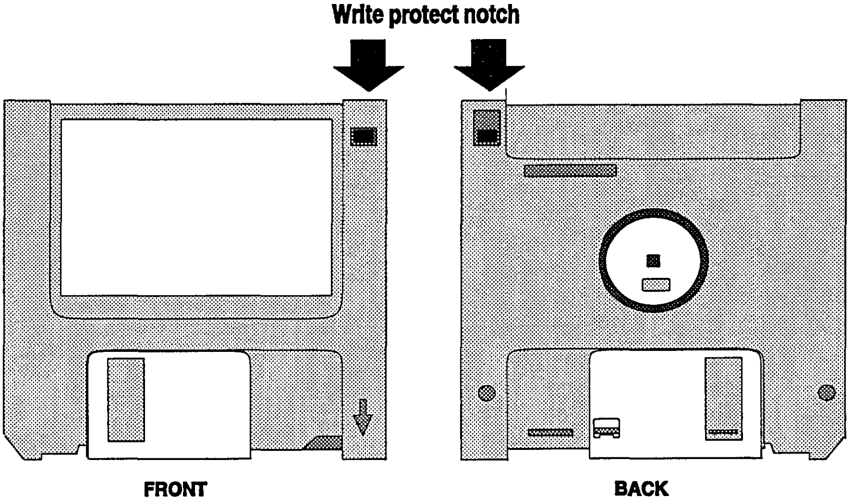
Diskettes can be physically write-protected to prevent anyone copying information onto them, or in any way changing their contents.

On 5¼" diskettes, there is a notch in the top right-hand side of the diskette. To write-protect the diskette, you place a write-protect tab (usually supplied with your diskettes) over this notch. If the tab is removed, the diskette is no longer protected. Write-protecting diskettes is a good way of ensuring that people realize a diskette has important contents.



Some 5¼" diskettes do not have the write-protect notch and these are permanently write-protected. Usually application diskettes come in this format to prevent you from erasing their contents.

With 3½" diskettes, the write-protect notch is in the same place as a 5¼" diskette but the tab is built into the diskette. You slide the plastic tab up to protect the diskette; when you do this you can see a hole through the notch.



Memory disks

There is a third type of disk, the *memory* disk (sometimes called a RAM disk or virtual disk). It is not a physical disk but is part of the computer's memory that DR DOS treats as a very fast disk. Files can be copied to or from a memory disk, and programs run from it. However, a memory disk is only a temporary means of storage, so when you switch off or reset your computer anything stored in the *memory* disk is lost. For permanent storage you must always use a hard disk or diskettes. Memory disks are not usually configured as standard. You can read more about setting up and using a memory disk in Chapter 11, "Customizing your system".

Drives

Disks are accessed by the slots or *drives* into which they fit on your personal computer. The various drives within your computer are designated by letters followed by a colon. The first diskette drive is usually the A: drive, and a second diskette drive the B: drive. The main hard disk is usually referred to as the C: drive. If you have multiple hard disk partitions, each partition is given its own drive letter, so it can be accessed like a separate drive. A memory disk would be given the first available letter *after* the physical drives, for example D: or E:.

Files and directories

Information that is to be held on a permanent basis on your computer is stored on disks, and is organized into *files*. A file can contain an application program, text like a letter or report, or computer data. A lot of your work within DR DOS will involve files.

Files always have names to identify them (up to eight characters in length), and the filename can have an optional extension of up to three characters, separated from the filename by a period (.), for example:

letter.txt

The filename extension helps to identify the class of file, for example **.txt** means the file is a *text* file.

DR DOS organizes files by storing them in directories. Directories contain certain information on the files they have stored in them, including their size, time of creation, and when they were last changed. Directories can be arranged in a hierarchical fashion, and so contain subdirectories as well as files. Files and directories are further explained in Chapter 3, "Managing your information".

DR DOS: the operating system

DR DOS links all the different parts of your computer together and makes them work as a complete system. It is the link between your computer hardware and software. (Hardware means the physical parts of your computer like the screen and keyboard, and software is the programs and applications you run on your computer.)

DR DOS creates the environment in which you interact with the computer, and within which your programs work. The operating system is a whole *series* of programs stored in the most part on disk, but called into the computer's memory when required. DR DOS runs in the background basically "operating" your computer, hence the name Disk Operating System (or DOS for short).

You work directly with DR DOS in many ways. You use it to run programs like a word processor, spreadsheet, or business application. You work directly with DR DOS to organize the information you create in your applications into files and those files into directories for ease of access. You use DR DOS to manage your information, for example removing files and directories you no longer need. You also use DR DOS to customize your system, for example to make your programs run faster and more efficiently.

There are many things you can do with DR DOS. As your knowledge develops, you will find more and more applications of, and features within, DR DOS to enhance your work.

1.2 Different disk configurations

Your personal computer is likely to have one of three basic disk configurations:

❑ A hard disk system with a single diskette drive.

❑ A twin-diskette system.

❑ A hard disk system with two diskette drives.

With the twin-diskette system, you keep your DR DOS diskette in one drive and perform most of your operations on a diskette with your own data on it, in the other drive. With the hard disk systems most of your work will be performed on the hard disk.

1.3 Preparing disks for use

All *physical* disks (hard disks and diskettes) need to be prepared using DR DOS before you can work with them. This process is called *formatting* for diskettes and *partitioning* for hard disks. In both cases it involves dividing the disk electronically into sections so that DR DOS can store and retrieve files more easily.

WARNING The preparation process erases everything stored on a disk, whether it is a hard disk or diskette, so great care must be taken when using it.

Preparing a hard disk

When you install DR DOS, the INSTALL program detects whether your hard disk needs preparing. You should only need to prepare a hard disk once. In most cases your dealer will have already done this for you. Remember you should only prepare your hard disk when it is new and blank or if you want to erase *everything* from it and start again. You prepare a hard disk using the utility FDISK, which is described in Chapter 10, "DR DOS commands".

Preparing diskettes

New, blank diskettes need to be prepared using DR DOS before you can store information on them. To prepare a diskette, you can use the DR DOS command FORMAT. This command is explained in Chapter 10, "DR DOS commands". For FORMAT to work, your diskettes must not be write-protected (see section 1.1, "Essential concepts").

1.4 How to install DR DOS

Introduction

Installing DR DOS is straightforward. When the INSTALL program is run, you work through a series of screens which give clear instructions on what you should do.

Getting online help

The INSTALL program provides context-sensitive help. To get help, move to the field on the screen for which you want help and press the **F1** key.

Cancelling installation

You can quit the INSTALL program and discard any changes made by pressing the **F10** key. You will see a screen asking you if you really want to leave the program, select the `Exit to DR DOS` option, then press the **Enter** key.

Restoring your old system

By answering Yes to a prompt during the INSTALL program that asks you if you want to save your old operating system, you will be able to restore it and remove DR DOS at any time in the future. See UNINSTAL in Chapter 10, "DR DOS commands".

Reconfiguring your system later on

Once you have installed DR DOS you can go back at any time and change the way your system is configured by running the DR DOS SETUP program, for example if you wanted to set up a memory disk. See Chapter 11, "Customizing your system".

Securing your system

By answering Yes to a prompt during INSTALL that asks you whether you want to install system security, you can protect your entire operating system from unauthorized access, using passwords. You can also install system security from the SETUP program. See Chapter 4, "Protecting your information" for information about security.

Starting the INSTALL program

You can either install onto your first hard disk partition (normally C:) or onto diskettes.

1. Insert the diskette labeled *Startup* into diskette-drive A: and close the door.
2. Switch on the computer, or if it is already on, press the **Ctrl+Alt+Del** keys at the same time (this is called "rebooting" your computer).
3. Wait for DR DOS to display the first screen of the INSTALL program, the Welcome screen.
4. Follow the instructions shown. They explain which keys you can use to move between the various screens and fields and what each key does. The keys you use to move through the INSTALL program vary. The bottom line of each screen lists all the keys that pertain to this screen. The bottom line is updated as you move from field to field, or between screens.
5. **Where to install**

   The next screen asks you where you want to install DR DOS, on your hard disk or on a diskette.

   Preparing a hard disk

   The INSTALL program detects whether your hard disk has been prepared for you. If it has not been prepared, you need to prepare your hard disk using a program called FDISK which INSTALL automatically starts for you at this point.

WARNING FDISK should be used with care when preparing a hard disk. It is normally only used to prepare new disks.

For detailed instructions on how to use FDISK, refer to Chapter 10, "DR DOS commands". When FDISK has finished, you are prompted to reboot your computer by pressing **Ctrl+Alt+Del** (ensure your *Startup* diskette is in diskette-drive A:), and start again from step 3.

6 **Selecting** configuration size

If your hard disk already contains configuration information from a previous installation of DR DOS, you will be asked if you want the INSTALL program to use this configuration information instead of its own defaults (where applicable).

If you choose to have your previous configuration files ignored, or you have not installed DR DOS before, you are presented with a screen which asks you to select the default size of your configuration.

There are three options in this screen. You can choose between increased application memory at the expense of operating system performance, a balance between application memory and operating system performance, or increased operating system performance and functionality at the expense of application memory.

If you are not sure what to choose, select the middle option as it provides the best balance between system performance and application memory space.

7 **Selecting country, keyboard and keyboard type**

In this screen the INSTALL program asks you to define the country you live in and the national keyboard you wish to use. You are prompted as to whether you want to accept the default installation values or modify them.

Note

At this point, if your supplier of DR DOS has opted to include ViewMAX, you will be asked what mouse and display you want to use.

8 **Replacing existing files**

If you are installing onto a hard disk and updating to the new version of DR DOS, the INSTALL program gives you the option of replacing any existing DR DOS files.

WARNING When you select the option to replace DR DOS files, all files on your hard disk in all subdirectories that have the same name as DR DOS files are replaced. This will happen even if the files were created in a different operating system from DR DOS.

9 Select installation or SETUP

At this screen you can either accept default installation values, as selected in step 6, and go straight to installing DR DOS, or run the SETUP program and modify the default values.

Using SETUP

In SETUP a series of screens are displayed for you to supply the new values. Each screen has at least one option, and may also have one or more fields which you can edit. To move between fields and options, use the **Up** and **Down** arrow keys or the **Tab** key. There is always an option to continue without changing the settings.

In the next two screens you are asked where you want DR DOS to be installed, and you can change your default path and append statements. For more information on PATH and APPEND see Chapter 10, "DR DOS commands".

Moving within the SETUP screens

Each option is identified by a *box* shape on the left or right of the text. You will see that one option on the screen is highlighted. This is the *current option.* To choose another option, use the arrow keys or the **Tab** key to move to it. When you move to the option, it is highlighted on the screen. Press **Enter** to select it and then move to the next screen.

Fields that you can edit are displayed in reverse video. You can get help information on a field at any time by pressing the **F1** function key. As you move to each field, you see a flashing cursor next to the current field. Also, the menu line at the bottom of the screen is updated to show which keys you can use to edit the field.

For example, if you are in a field from which you can choose `YES/NO` or `ON/OFF`, press the **Spacebar** to switch between the values. If you are in a numeric field, you can change the number in the field using the numeric keypad. To discard changes you have made to a field, press the **Esc** key.

When you have finished, press **Enter** to move to the next field, or to the continue option to move to the next screen.

Retracing your steps

You can go back over the screens (and modify anything you have changed) by pressing the **Esc** key until you reach the screen you want.

Installing DR DOS

When you have finished configuring your system, or if you accepted the default values, you start the installation process.

Enabling UNINSTAL

You are asked whether or not you want to save your old operating system. Saving your old system means that you can use the DR DOS UNINSTAL command to restore it, and remove DR DOS from your hard disk. Refer to the UNINSTAL command description in Chapter 10, "DR DOS commands".

Press **Enter** when you are prompted to start DR DOS installation. You will be prompted to insert the remaining installation diskettes during the procedure.

DR DOS starts to copy over various files to your disk:

- ❑ Two special *system* files that contain the main operating system are copied to your root directory. If files of the same name already exist, DR DOS overwrites them.
- ❑ The file **command.com** is copied to your root directory.
- ❑ Two files, **autoexec.bat** and **config.sys**, are *created* in your root directory. These files contain configuration information that DR DOS needs to access every time it is started up. For more information, see Chapter 11, "Customizing your system". If you already have these files on your disk, they are overwritten.
- ❑ All other DR DOS files are copied to a directory of your choice. If you do not specify a directory, one is created for you from the root directory called DRDOS. If DR DOS files already exist on the chosen directory, they will be overwritten.

Note

To learn about files and directories, see Chapter 3, "Managing your information".

When installation is complete, the INSTALL program prompts you to restart your computer by pressing the **Ctrl+Alt+Del** keys at the same time. After your computer completes its startup tests, you will see the DR DOS prompt, for example:

```
C>
```

You can always go back any time in the future to alter any aspect of your computer's configuration by using the SETUP program, see Chapter 11, "Customizing your system".

1.5 Getting started at the command line

The system prompt

When you begin working with DR DOS you start at the command line or DR DOS prompt. Whenever you see the system prompt you know the computer is waiting for you to enter instructions at the keyboard.

If DR DOS is installed on a hard disk, the prompt is:

```
C>
```

This means that drive C: is the *current* disk drive. When you enter an instruction, DR DOS always looks by default to the current disk drive.

If DR DOS is installed on diskette, the prompt is:

```
A>
```

Changing between drives

When you enter an instruction, DR DOS always looks to the current disk drive first. If you want a command to be directed at a different drive, you change to it by typing the new drive letter plus a colon and pressing **Enter**. For example:

```
C>A: ↵
```

changes the current drive (C:) to diskette drive A:, and produces the prompt:

```
A>
```

Any instruction you type now is directed at the A: drive. For example:

```
A>DIR ↵
```

gives a listing of all files and directories on a diskette inserted in drive A:.

You can tell DR DOS to look at another drive without changing the current drive. To do this include the drive letter and colon when entering the command. So, instead of the example above, you could type:

```
C>DIR a: ↵
```

When giving instructions to your computer, for example starting a program, you must always direct them at the drive where the program or information is stored.

Entering instructions

After you enter an instruction at the command line, you must always press **Enter** to make DR DOS carry out the instruction.

If you make a mistake when typing a command, you see a message like this:

```
    Command or filename not recognized
```

It may be that you have merely mistyped the instruction or directed the command at the wrong drive.

If you make a typing error and you have not yet pressed **Enter,** you can correct it by pressing the **Backspace** key (←) which erases the command line one character at a time. You can then retype the command. DR DOS also has a powerful feature that stores commands entered previously and lets you recall them by pressing the up and down arrow keys. You can then edit and re-issue them. You may have set this up at installation by setting the HISTORY value to ON. If it was not set up at installation, it is very easy to set it up using the SETUP program (see Chapter 11, "Customizing your system").

This and other ways of editing the command line are explained in Chapter 10, "DR DOS commands", section 10.2.

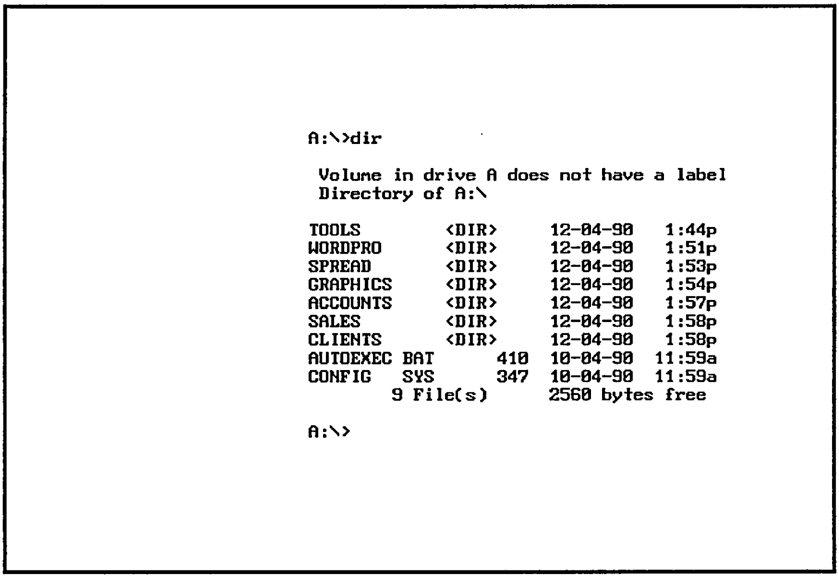
Some simple commands

DR DOS contains an extensive set of commands. A command is simply a method of getting your computer to carry out a particular task like copying a file. Most commands act upon whole disks, or a specified file or group of files. Command names are up to eight characters in length and are usually followed by options which contain further information about them.

One useful command which you can get to know now is DIR, which provides a DIRectory listing of files and directories. Try typing the letters DIR at the command prompt, and press **Enter**:

```
DIR ↵
```

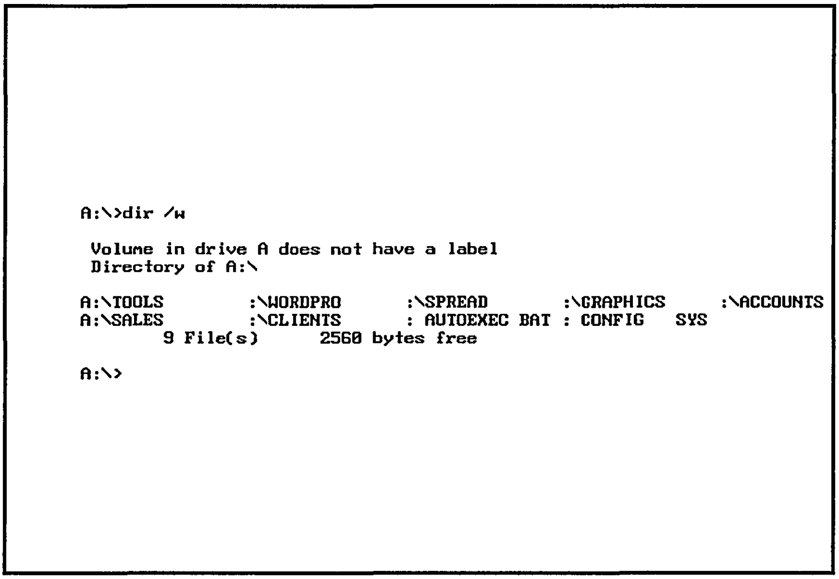
You should see a listing of all the files and subdirectories in the *root* directory of the disk in the current drive, with their size in bytes (characters) and the date and time they were last modified, plus how much space is left on the disk, as follows.



```
A:\>dir

 Volume in drive A does not have a label
 Directory of A:\

TOOLS        <DIR>      12-04-90   1:44p
WORDPRO      <DIR>      12-04-90   1:51p
SPREAD       <DIR>      12-04-90   1:53p
GRAPHICS     <DIR>      12-04-90   1:54p
ACCOUNTS     <DIR>      12-04-90   1:57p
SALES        <DIR>      12-04-90   1:58p
CLIENTS      <DIR>      12-04-90   1:58p
AUTOEXEC BAT       410  10-04-90  11:59a
CONFIG   SYS       347  10-04-90  11:59a
        9 File(s)      2560 bytes free

A:\>
```

The root directory is the first directory on your disk (read Chapter 3, "Managing your information" for more information on directories). You can add an option to the command, for example /W which shows the listing in a wide format, giving space for more files on the screen:

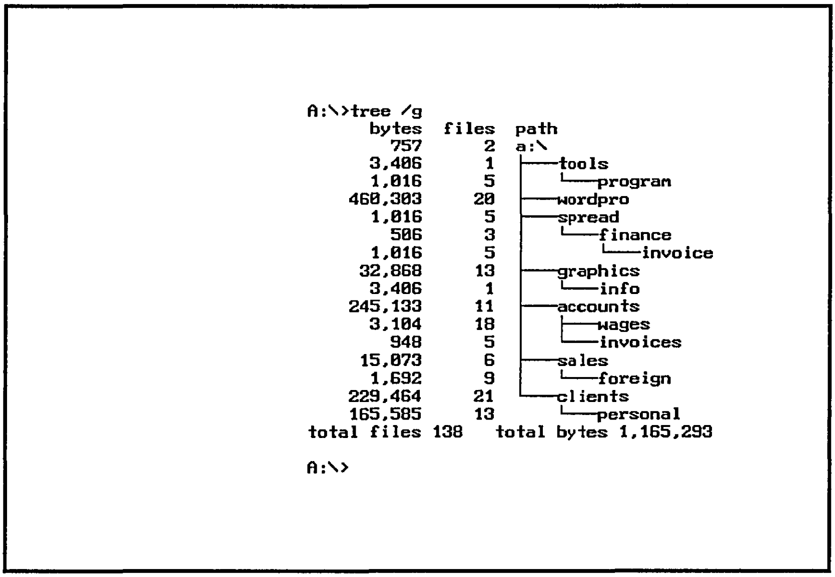
```
DIR /w ↵
```



```
A:\>dir /w

 Volume in drive A does not have a label
 Directory of A:\

A:\TOOLS        :\WORDPRO      :\SPREAD       :\GRAPHICS      :\ACCOUNTS
A:\SALES        :\CLIENTS      : AUTOEXEC BAT : CONFIG   SYS
       9 File(s)      2560 bytes free

A:\>
```

Notice that the sizes and modification times of the files are not displayed, but many more files can be listed on the screen.

Another useful command is the TREE command. It is similar to DIR in that it displays a listing of directories, but it shows all the directories on a disk wherever you type it from. For example it could display as follows:



```
A:\>tree /g
      bytes  files  path
        757      2  a:\
      3,406      1  ├───tools
      1,016      5  │   └───program
    460,303     20  ├───wordpro
      1,016      5  ├───spread
        506      3  │   └───finance
      1,016      5  │       └───invoice
     32,868     13  ├───graphics
      3,406      1  │   └───info
    245,133     11  ├───accounts
      3,104     18  │   ├───wages
        948      5  │   └───invoices
     15,073      6  ├───sales
      1,692      9  │   └───foreign
    229,464     21  └───clients
    165,585     13      └───personal
total files 138   total bytes 1,165,293

A:\>
```

For more information on the TREE command, see Chapter 3, "Managing your information".

Getting help on commands

You can display help text on any DR DOS command by typing the command followed by /? or /H. For more detailed help information, use the DOSBOOK command. See Chapter 10, "DR DOS commands".

For more information on commands, see Chapter 2, "Basic command concepts". Chapter 10, "DR DOS commands" provides a detailed list of the DR DOS commands.

1.6 Copying your master diskettes

Before going any further make a copy of your DR DOS master diskettes using the DISKCOPY command. The DISKCOPY command automatically formats a diskette for you if it is not already formatted.

WARNING DISKCOPY erases any information already stored on the diskette you are copying to. Make sure you use a blank diskette or one whose contents are not important to you. You need to know your drive and diskette types before you copy diskettes. You cannot copy from one type of physical drive to another, for example from a 3½" to a 5¼" drive, or between different density disks, for example from a 1.2 Mbyte disk to a 360 Kbyte disk.

To copy your master diskettes:

1. From the DR DOS prompt type:

   ```
   DISKCOPY a: a: ↵
   ```

   You are prompted to insert your source diskette. Insert the diskette labeled *Startup* into diskette-drive A:. This is your *source* diskette - the diskette you want to copy *from*. Press any key to start copying.

2. A message appears tracking the copying process. You are prompted to insert your destination diskette. Remove the *Startup* diskette and insert a blank diskette to copy *to*. Press any key to continue.

3. Follow the instructions as they advise you, to insert the source or destination diskettes in turn. Use the *same* source and destination diskettes until the DISKCOPY process is completed.

4. When the copying is complete, you are asked:

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

   Remove the diskette already in the drive, press Y and insert diskette 2. Repeat the process until all the installation diskettes are copied.

5. Label your copied diskettes clearly, write-protect them, and store them and your master diskettes in a secure place.

Twin diskette drive systems

If you have a twin diskette system, insert a blank diskette into drive B: and type:

```
DISKCOPY a: b: ↵
```

Follow the instructions on the screen.

1.7 Using other applications

To start other programs and applications, like a word processor or spreadsheet, generally all you need to do is type the program name at the command prompt, and press the **Enter** key. For full details refer to your application's documentation.

2

Basic command concepts

THIS chapter is an introduction to working with DR DOS commands. A command is simply a method of getting your computer to carry out a particular task like copying a file. Most commands act upon whole disks, or a specified file or group of files.

The *command line* is the command followed by any other relevant information given at one time to the computer to enable it to carry out your instructions. The command line ends when you press the **Enter** key. DR DOS has a powerful feature that enables you to recall previously-entered commands; this is described in "Editing commands at the command line" in Chapter 10, "DR DOS commands".

2.1 Command types

Internal commands

Certain DR DOS commands that are very useful, and therefore needed often, are stored in memory for quick access. They are loaded from a special file called **command.com**. This makes it easier for DR DOS to carry them out. These commands are called internal or resident commands (because they are kept in memory). Not many commands are stored in this way as it would take up too much space. These commands are available to you when you have the system prompt in front of you on the screen.

You do not need to change to a particular drive or directory to use internal commands.

The internal commands are listed in Table 2-1. For a summary of what each command does, see the beginning of Chapter 10, "DR DOS commands".

Table 2-1 DR DOS Internal Commands

BREAK	DELQ	PATH	VERIFY
CHCP	DIR	PROMPT	VOL
CHDIR/CD	ERAQ	RENAME/REN	
CLS	ERASE/ERA	RMDIR/RD	
COPY	EXIT	SET	
CTTY	HILOAD	TIME	
DATE	MKDIR/MD	TYPE	
DEL	MORE	VER	

External commands

Some commands that are used less often are not automatically loaded into memory. To save space they are stored on disk instead. These are *external* commands. They have the file extension **.com** or **.exe**. When you enter an external command, DR DOS retrieves the “command file” from disk and runs it.

The memory used by the command is freed when it has finished its job. BACKUP, RESTORE and DISKCOPY are examples of external commands, and will not work unless DR DOS has access to a directory containing the command files.

DR DOS provides a command called PATH to make sure that external commands can always be found, even if they are stored somewhere other than the current directory. The PATH command is described in Chapter 3, "Managing your information". The DR DOS external commands are listed in Table 2-2.

Table 2-2 DR DOS External Commands

APPEND
ASSIGN
ATTRIB
BACKUP
CHKDSK
COMMAND
COMP
CURSOR
DELPURGE
DELWATCH
DISKCOMP
DISKCOPY
DISKMAP
DISKOPT
DOSBOOK
EDITOR
EXE2BIN
FASTOPEN
FC
FDISK
FILELINK
FIND
FORMAT
GRAFTABL
GRAPHICS
JOIN
KEYB
LABEL
LOCK
MEM
MEMMAX
MODE
MOVE
NLSFUNC
PASSWORD
PRINT
RECOVER
RENDIR
REPLACE
RESTORE
SETUP
SHARE
SID
SORT
SUBST
SUPERPCK
SYS
TASKMAX
TOUCH
TREE
UNDELETE
UNFORMAT
UNINSTAL
VIEWMAX
XCOPY
XDEL
XDIR

Getting online help on commands

All commands followed by /? or /H will display help text.

For more detailed online help, use the DOSBook command. See Chapter 10, "DR DOS commands".

Batch processing subcommands

Batch files are files in which you store a set of often-used commands. When you type the name of the batch file at the command line and press **Enter**, the commands are carried out as if you had typed them in one after the other. For example, you can create a batch file that stores the commands you use to start an application such as a spreadsheet.

Batch files can be very powerful and DR DOS has a special set of subcommands for use in them. These subcommands are stored in the same way as internal commands, and are listed in Table 2-3. They are also described in more detail in Chapter 9, "Grouping commands in batch files".

All batch processing subcommands are internal, and are listed in Table 2-3.

Table 2-3 Batch Processing Subcommands

@	FOR	PAUSE	SWITCH
:LABEL	GOSUB	REM	
CALL	GOTO	RETURN	
ECHO	IF	SHIFT	

Config.sys commands

Apart from the commands we have already discussed, there is a special set of system configuration commands that are not entered at the keyboard, but are put into a file called **config.sys**. The **config.sys** file is looked at by DR DOS every time it is started. It then carries out any of the commands it finds there. These commands are set up for you automatically when you install DR DOS. For more information see Chapter 11, "Customizing your system".

2.2 Entering commands

As mentioned in Chapter 1, "Getting started with DR DOS", DR DOS command names are up to eight characters in length and are usually followed by options (or parameters) that contain further information about the command, for example:

```
CHKDSK /F
```

Command switches

Command parameters shown with a forward slash, for example /F, are referred to as switches. This is because they are used to tell DR DOS to switch to a particular variation on a command. /F tells CHKDSK to fix any errors detected. If you did not include the /F switch, errors would not be automatically fixed.

Punctuation is usually needed to separate command names and their options. Generally you put a space between a command name and an option, and between options themselves.

Sometimes other symbols must be used, for example commas or semi-colons; this is true of the MODE command:

```
MODE COM1:9600,N,8,1
```

Command syntax

The syntax or format of a command is made up of the command name itself and all its possible parameters and associated punctuation marks. In Chapter 10, "DR DOS commands", each command description begins by setting out the full syntax. For example, the syntax for CHKDSK, a useful command for getting a status report on the condition of a disk, is shown as:

```
CHKDSK[/?|/H][d:|wildspec][/V]
```

Square brackets mean that the parameters they contain are optional. The vertical bar (|) means choose one or the other. You can enter commands and their options in upper or lower case or a combination of both. Commands only take effect after you press the **Enter** key.

A command specification can be lengthy. You will find a complete explanation of them in Chapter 10, "DR DOS commands".

Specifying drives

Often, when entering a command, you need to state a *source* drive and a *destination* drive. The term *drive* refers to your hard disk and diskette drives, or it could refer to other removable storage disks and temporary memory disks created using the VDISK.SYS utility. You should have already encountered this in Chapter 1, "Getting started with DR DOS", when you copied your master diskettes. Remember that DR DOS is a *disk* operating system, and commands need to be directed at the correct disk drive for DR DOS to be able to carry out the instruction. For example, with the BACKUP command (see Chapter 4, "Protecting your information") you specify where you are backing up **from** and **to**:

```
BACKUP source_drive [filename] destination_drive
```

The source drive is generally your hard disk (for example C:) and the destination drive a diskette in drive A:.

2.3 Displaying data a screenful at a time

When a command displays a lot of information, it may move down the screen (scrolling), so you miss the first part of the display. This will happen if you enter a DIR command and the listing is so long it cannot all be shown on the screen at once.

You can suspend scrolling by pressing **Pause** or **Ctrl+S**. To start scrolling again press any key. Some commands have a /P switch which, if specified as part of the command line, causes listings to display one page at a time.

You can also use MORE to display data a screenful at a time. See the MORE explanation in Chapter 10, "DR DOS commands".

2.4 Stopping commands

You can stop commands when they are running by pressing **Ctrl+C** or **Ctrl+Break**. The processing of the command is stopped, and the command prompt returned.

3

Managing your information

AS you work with your applications, you create your own information which needs to be organized for ease of access and security. You also need to learn how to manage your information, for example to copy, delete, rename and print it. This chapter explains how you can achieve this with DR DOS using:

- ❑ Files.
- ❑ Directories.
- ❑ Directory tree structures.
- ❑ Paths.
- ❑ Various commands that handle files and directories.
- ❑ Wildcard characters.

3.1 Storing data in files

Information on your computer is stored in files, and a lot of your work with DR DOS will involve them. A file is a collection of information that has something in common, for example it can consist of an application program, text like a letter or memo, or computer data.

Naming files

Files always have a name to identify them (up to eight characters long) and filenames often have an extension (up to three characters long, preceded by a period), for example:

myfile.doc

Your filenames can be any combination of A-Z and 0-9. The following special characters are reserved by DR DOS and must not be used in filenames:

< > = , ; : . * ? [ ] ( ) / \+ ! ~

Try to develop a logical system for naming your files, and make use of filename extensions. For example, you can identify all your correspondence by giving each file an extension like **.let** for "letter" or **.doc** for "document". Naming related files in this way becomes particularly useful when you want to handle groups of files using *wildcards*, which are discussed later in this chapter.

Some extensions are automatically assigned by programs, for example Lotus 1-2-3 assigns the extension **.wks**.

Some common filename extensions are explained in Table 3-1, you should not use these filename extensions for any other purposes.

Table 3-1 Common Filename Extensions

Extension	Description
.$$$	Temporary or incorrectly closed file. Many programs write temporary files on the disk or diskette during their operation.
.bak	Backup file created by a text editor.
.bas	File containing a BASIC program.
.bat	A file containing one or more commands that DR DOS executes sequentially.
.com	DOS program file.
.cpi	Code Page Information file containing specific national character sets.
.exe	DOS program file.
.sys	Device drivers and the **config** file.

Filenames you should not use

DR DOS uses special names to identify peripherals (devices added to your computer, for example a printer). These are called reserved device names. You should not call your files by these names; they are listed in Table 3-2.

Table 3-2 Reserved Device Names

Device name	Explanation
CON	User console (the keyboard and the screen).
AUX or COM*n*	Serial communications port, where *n* is 1, 2, 3, or 4, depending on the particular computer.
PRN or LPT1	First parallel printer LPT1.
LPT2	Second parallel printer.
LPT3	Third parallel printer.
NUL	All output to this device is discarded. No input is returned from the device.
CLOCK$	Provides data and time support.

Handling groups of files (using wildcards)

To help you handle groups of files with ease and flexibility, there are two special characters, the question mark (?) and asterisk (*), known as *wildcards*.

Wildcards provide a means of referring to more than one file at a time in a command specification. Using wildcards you partially specify a filename so that several files can match it. You can have wildcards in both the filename and the extension.

The ? wildcard

The question mark wildcard matches with any character in the exact position the question mark occupies.

For example:

```
chapter?.doc
```

matches with:

```
chapter1.doc
chapter2.doc
chapter3.doc
```

So, to copy all three chapters to a diskette, you only need the one command:

COPY chapter?.doc a:

The * wildcard

The asterisk wildcard is shorthand for several question mark wildcards. Where the question mark wildcard is only valid for the one position, the asterisk wildcard is valid from the position in which it appears to the end of the filename or extension.

For example:

```
chap*.*
```

matches with the chapter files:

```
chapter1.doc
chapter2.doc
chapter3.doc
```

However, it also matches with other filenames like:

```
chapmem.fax
chaplist.doc
chapfax.bat
```

Commands and wildcards

The commands most often used with wildcards in their specification are COPY, DEL, ERASE/ERAQ/DELQ, DIR, MOVE, RENAME, and TREE.

WARNING Wildcards are very powerful tools. They are extremely useful, but they can also be very destructive. For example *.* means *all* files. This is called th*e global* wildcard. So if you type DEL *.* it erases ALL files in the directory in which it was carried out. When deleting files using *.*, you are always prompted before DR DOS carries out the operation, but you should still take great care when using the global wildcard. If you want to be prompted before deleting *any* file, use the ERAQ or DELQ commands.

Examples of using wildcards

If you have a diskette with six files:

b.let
b.pic
a.let
aa.let
a.pic
aaa.let

and you want to delete the **.let** files *only* using the ERASE command, type:

```
C>ERASE *.let
```

This erases all files with the extension **.let**.

If you want to erase **a.let** and **b.let**, type:

```
C>ERASE ?.let
```

This erases the **a.let** and **b.let** files, whereas:

```
C>ERASE ?.*
```

erases **a.let**, **b.let**, **a.pic** and **b.pic**.

Handling groups of files (using filelists)

If a filename is preceded by the @ character, the operating system looks in the file for a list of filenames (a filelist). Some commands that usually operate on only one file per command line can thus be made to operate on several.

You create a filelist as a text file (using EDITOR or a word processor). You can enter as many filenames as you wish; you can even use wildcard characters in the names.

Note that each filename in your text file must be placed on a new line, or be separated by a space, tab or comma. Note also that you cannot use wildcard characters in the filelist name.

For example, you could create a file called **source.fl**, containing the names of the following files:

***.doc**
sample.txt
henry.bak

If you then wanted to give one password to this group of related files, instead of issuing the command afresh for each file, you could type:

```
PASSWORD @source.fl /R:mycode
```

This assigns the password **mycode** to the files ***.doc, sample.txt** and **henry.bak**, in the subdirectory from which you gave the command. The @ symbol tells the PASSWORD command that the filename specified contains a filelist.

3.2 Using directories to organize your files

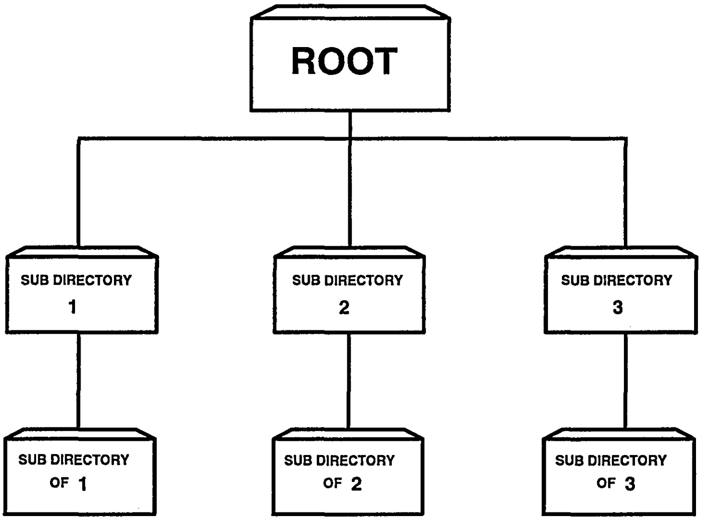
The more files you create, the harder it is to keep track of them. You will find you need to organize them into groups according to their content, by project names, user names or classes. You can do this by using *directories* to order your files.

The root directory

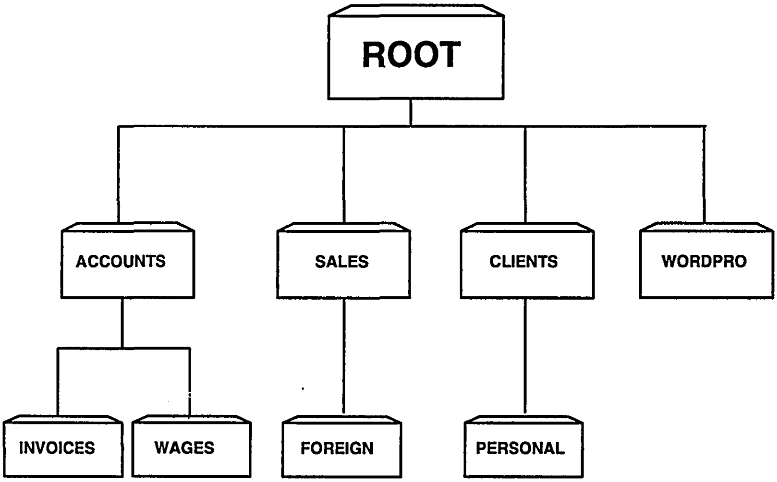
When you prepare a disk or diskette and start storing files on it, a directory is automatically created called the *root* directory. On diskettes, the root directory is often the only directory you will need. On a hard disk, the root directory is poorly equipped to contain all the files stored on it. Root directories can contain only a limited number of files, and as the number of files grow , locating them in a flat directory structure becomes well-nigh impossible. For these reasons, DR DOS allows you to build up hierarchical directory structures, one for each disk (or disk partition) in the system.

Subdirectories

Directories created from the root directory are called subdirectories. Subdirectories can also contain further subdirectories. As you add more directories to the structure it will begin to grow into an inverted tree, that is the root at the top and the branches growing downwards. This is also called a multi-level directory structure:



In the following tree, four subdirectories were made from the root directory, **accounts**, **sales**, **clients**, and **wordpro**:



In the subdirectory **accounts**, two more subdirectories were created called **invoices** and **wages**. The **sales** and **clients** subdirectories also have subdirectories, **foreign** and **personal**.

Files can be created anywhere in the structure. You can create many levels, but it is not practical to make too many, as it then becomes difficult to locate your files.

Naming directories

Naming directories follows the same conventions as naming files: a one-to-eight character name plus an optional extension of up to three characters. Normally it makes sense to keep directory names short.

The current directory

DR DOS considers the directory that you are working in at any one time as the current directory. When you carry out a task like creating a file, DR DOS assumes you mean it should be created in the current directory (that is the directory in which you are *currently working*), unless you specify otherwise.

You can travel around the directory structure from the root downwards, or from anywhere within the structure back up to the root. You can also go straight to a directory without going through intervening directories. You move between directories using the CHDIR (change directory) command, described later in this chapter. Each time you change directories using CHDIR, the destination directory becomes the new current directory.

Usually, your system prompt tells you what your current directory is. However, if your system prompt is not set up to do this, you can always type CHDIR, without any parameter, to find out.

Special directory symbols

All directories (except the root) are contained within other directories. A directory which contains a subdirectory is known as the parent of the subdirectory. DR DOS uses special symbols to indicate the current directory and the parent of (or directory above) the current directory:

.	Means the name of the current directory.
..	Means the directory above the current one.

These entries are automatically created by DR DOS when you make a directory. When you issue a DIR command, the entries **.** and **..** appear at the top of the listing.

If you type the command:

```
DIR ..
```

you get a directory listing of the files in the *parent* directory of your current directory.

Note

You cannot delete the **.** and **..** entries.

3.3 Specifying paths

Within a tree directory structure, it is quite acceptable for more than one file or directory to have the same name. For example, there may be a file called **france.doc** in the directory **\sales** and in the directory **\accounts**. You need a way of referring to files and directories which allows them to be specified uniquely, so you are able to access each of the two **france.doc** files in the above example without confusion.

DR DOS provides such a mechanism. Every time you want to access a file or directory you specify a *path* to it. A path is like a roadmap, it tells the operating system the route to follow to get to a particular file or directory.

You specify a path with a series of directory names, separated by backslashes (\). This traces a route through the tree structure to the desired point.

If the first directory in the path is preceded by a backslash, then the path begins at the root directory. Paths which begin with a directory name, are taken as starting from the current directory.

For example:

```
c:\sales
```

In this path, the search is to start on the C: drive.

In the following path, **sales** and **foreign** indicate the names of the directories DR DOS has to pass through to locate the file **france.doc**:

```
c:\sales\foreign\france.doc
```

When you add a filename to a path it becomes a file specification (or filespec), in other words a way of specifying where a particular file is. See Chapter 10, "DR DOS commands", section 10.1 for more information on defining a file specification.

The path in a file specification can contain any number of directories, and can be anything up to 63 characters in length.

The PATH command

In Chapter 2, "Basic command concepts" we talked about external commands, the commands that are stored on disk rather than in memory. They include all files with extensions **.exe** and **.com**. DR DOS needs to be told where these external commands are and you do this by using the PATH command. The PATH command can be used to specify a series of paths that DR DOS should search to find command program files. For example:

```
PATH=c:\drdos;c:\program;
```

With this command, you are telling DR DOS to first look in the subdirectory **c:\drdos** and if this proves unsuccessful, to look in the subdirectory **c:\program**. Notice that the individual paths are separated by semi-colons (;).

Usually you would keep your most-often used directories at the beginning of the path for speed of access.

Looking for external commands

It is good practice to keep your external commands in one directory. This is done automatically for you during installation; the external commands are placed in the directory **\drdos**. Therefore it is useful for you to have the directory **\drdos** at the beginning of the PATH:

```
PATH=c:\drdos
```

Once a path is set up, you can type commands without having to worry about being in the wrong directory. The path remains set up until you restart your computer or issue another PATH command.

You can always display the current path by typing:

```
C>PATH
```

You can clear the path by typing:

```
C>PATH ;
```

As well as looking for external commands you also use PATH to search for batch files (see Chapter 9, "Grouping commands in batch files").

Looking on different disks

Using paths you can make DR DOS search on different disks or diskettes for files or programs. This is useful if you have a twin diskette drive system and no hard disk. You may have a program diskette in each drive, and you want DR DOS to switch between them to search for different program files. You can set up a path like this:

```
PATH=a:\;b:\;
```

DR DOS knows to look first in the root directory of the A: drive, and if the program is not there, to switch to the B: drive.

Note

Always specify the complete path including the drive for all the subdirectories in the path.

PATH and the autoexec.bat file

The best time to issue the PATH command is when DR DOS starts up. You can do this by including a PATH command in a file called **autoexec.bat**, which is always loaded as part of system startup. You can use the SETUP program to set (or modify) your path, as described in Chapter 11, "Customizing your system".

Note

Another command that works in a similar way to the PATH command is APPEND. Instead of **.exe**, **.com** and **.bat** files, it works with data files. See the explanation for APPEND in Chapter 10, "DR DOS commands".

3.4 Working with directories

There are four commands that relate specifically to handling directories: TREE, MKDIR, CHDIR, RENDIR and RMDIR.

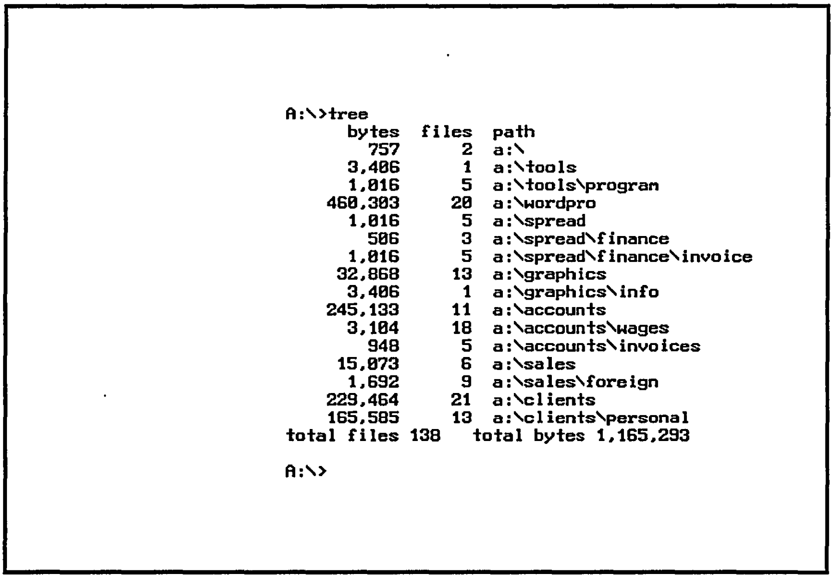
Listing directories (the TREE command)

The TREE command allows you to list all the directories on a given drive and keep a track of your "tree" structure, for example:

A>**TREE**

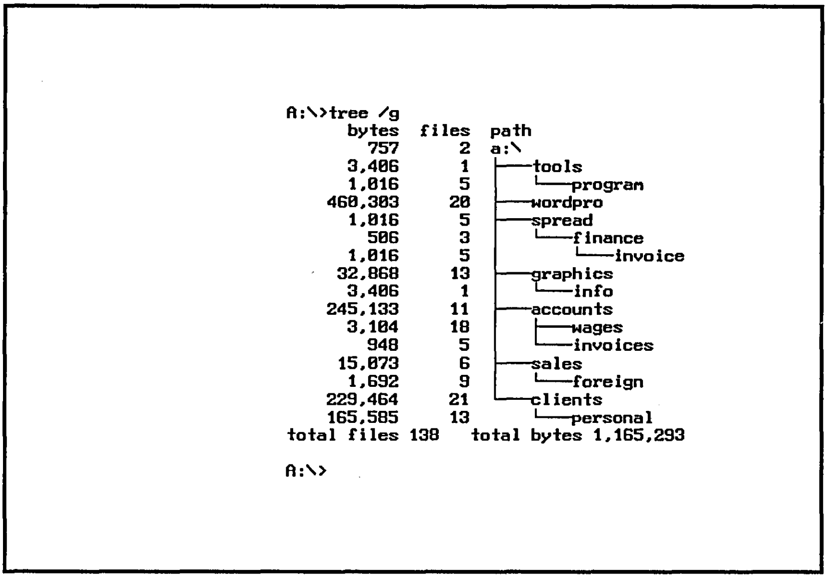
gives you a listing of the directories on the A: drive.

The list gives the space used and the number of files belonging to each directory, as well as the path to each directory like this:



```
A:\>tree
      bytes  files  path
        757      2  a:\
      3,406      1  a:\tools
      1,016      5  a:\tools\program
    460,303     20  a:\wordpro
      1,016      5  a:\spread
        506      3  a:\spread\finance
      1,016      5  a:\spread\finance\invoice
     32,868     13  a:\graphics
      3,406      1  a:\graphics\info
    245,133     11  a:\accounts
      3,104     18  a:\accounts\wages
        948      5  a:\accounts\invoices
     15,073      6  a:\sales
      1,692      9  a:\sales\foreign
    229,464     21  a:\clients
    165,585     13  a:\clients\personal
total files 138   total bytes 1,165,293

A:\>
```

The TREE command has some useful parameters such as /F, which lists all the files in each directory, as well as the directories themselves. You can also produce a graphical display by specifying the /G switch, which generates a display similar to the one on the next page.



```
A:\>tree /g
    bytes  files  path
      757      2  a:\
    3,406      1  ├──tools
    1,016      5  │  └──program
  460,303     20  ├──wordpro
    1,016      5  ├──spread
      506      3  │  └──finance
    1,016      5  │     └──invoice
   32,868     13  ├──graphics
    3,406      1  │  └──info
  245,133     11  ├──accounts
    3,104     18  │  ├──wages
      948      5  │  └──invoices
   15,073      6  ├──sales
    1,692      9  │  └──foreign
  229,464     21  └──clients
  165,585     13     └──personal
total files 138   total bytes 1,165,293

A:\>
```

Note

A command that lists the contents of directories and the subdirectories of those directories is the XDIR command, or *Extended Directory* command, which is described in Chapter 10, "DR DOS commands".

Making a directory (the MKDIR command)

To create a directory, use the MKDIR command (which can be abbreviated to MD). The directory above the one you wish to create must already exist.

To create the directory **accounts**, for example, type:

```
C>MKDIR \accounts
```

To make a directory below **accounts**, type:

```
C>MKDIR \accounts\invoices
```

or you can create a directory from the **accounts** directory itself.

The directories, **accounts** and **invoices**, are now ready to receive files, and to be made the *current* directory. You can now create files with a word processor to store in these directories.

Note

The . and .. entries in a directory are automatically created by DR DOS, which uses these symbols to indicate the current directory, and the parent of the current directory; see section 3.2, "Using directories to organize your files".

Displaying and changing directories (the CHDIR command)

To change between directories in DR DOS type CHDIR (or CD for short) followed by a path. The directory you change to becomes your current directory.

For example, to make the **accounts** directory the current directory type:

```
C>CHDIR accounts
```

To return to the root directory type:

```
C>CHDIR \
```

(Remember that the backslash represents the root directory).

Also useful is the .. symbol. When you type:

```
C>CHDIR ..
```

you are moved up one level, for example from the **accounts** directory to the root directory.

Note

If you type CHDIR without a path, you can find out the name of the directory you are in.

Renaming directories (the RENDIR command)

Use the RENDIR command to change the name of an existing subdirectory.

For example, to change the name of the current directory **mydir** to **accounts**, type:

```
C>RENDIR mydir accounts
```

See Chapter 10, "DR DOS commands" for more information about the RENDIR command.

Removing a directory (the RMDIR command)

The RMDIR (or RD) command removes an empty directory.

For example to remove the subdirectory **\accounts\invoices**, where the current directory is **\accounts**, type:

```
C>RMDIR invoices
```

To remove the same directory when the current directory is the root, type:

```
C>RMDIR \accounts\invoices
```

Notes

The directory to be deleted must be empty of files. The DIR command displays only the entries **.** and **..** when a directory is empty. If DIR lists any more entries, the directory is not empty.

The directory to be deleted must not have any subdirectories below it. Remove them first.

You cannot delete a directory while it is the current directory.

3.5 General actions on files

This section touches on other useful commands you can use with files. Note that these are overviews; you should go on to read Chapter 10, "DR DOS commands" for more detailed descriptions.

Copying files (the COPY command)

A command that you will need to use often is COPY. Using it, you can copy one or more files between disks (for example from a hard disk to a diskette), between directories, or from one file to another.

For example:

```
C>COPY wages.txt a:
```

When you copy from disk to disk, you can use the same name for a file. However you cannot do this on the *same* disk, unless the file is in a different directory. For example you could type:

```
C>COPY report.doc c:\sales\report.doc
```

Note

The XCOPY command is another useful command for copying. It can copy groups of files including a whole tree structure of files (for example files and directories) and is faster than COPY. See Chapter 10, "DR DOS commands".

Moving files (the MOVE command)

You can move one or more files, or the contents of entire subdirectories, by copying files to new locations and then deleting the originals. The MOVE command lets you move files on the same disk or between different disks.

The following command, for example, moves a file permanently from a diskette to the hard disk:

```
A>MOVE intro.txt c:\report
```

You can also specify a subdirectory to move, rather than moving files individually.

See Chapter 10, "DR DOS commands", for more information about the MOVE command.

Deleting files

Eventually you will want to delete files from your disks or diskettes to keep them from becoming too cluttered. You can use one of the five delete commands to do this: DEL, ERASE, DELQ, ERAQ, or XDEL. Type the command followed by the filename you want to erase:

```
C>DEL myfile.txt
```

All the commands work in much the same way, except that DELQ and ERAQ always prompt you before deleting a file. XDEL enables you to delete files and subdirectories.

Caution Take care when using the delete commands, particularly if you use wildcard characters. If you want to use wildcard characters, it is safer to use the ERAQ or DELQ commands.

Displaying file contents (the TYPE command)

You can display the contents of a *text* file on your screen using the TYPE command like this:

```
C>TYPE myfile.txt
```

This command works only with text files. If you try to display a program file using TYPE, you get a lot of unintelligible characters and symbols.

If you are looking at a large file, to see a screenful at a time, use the /P (Pause or Page) parameter:

```
C>TYPE myfile.txt /P
```

Press any key to see the next screen of text.

Renaming files (the RENAME command)

Sometimes you need to change the name of a file or group of files. To rename a file enter the command RENAME (you can shorten it to REN), followed by the filename you want to change and then the name you want to change it to, for example:

```
C>RENAME myfile.txt newfile.txt
```

If the file you are renaming is not on the current drive, you need to state where it is:

```
C>RENAME a:myfile.txt a:newfile.txt
```

You can rename groups of files using the wildcard characters, for example:

```
C>REN *.bat *.old
```

Printing files (the PRINT command)

If you have a printer connected to your computer, you can use the PRINT command to send a file to it:

```
C>PRINT myfile.txt
```

You are prompted for the name of the *print device* connected to your computer:

```
    List device? [PRN]
```

The print device is the *printer port* the printer is connected to (a port is a socket in your computer which can be used to connect your computer to a printer or other device).

Press **Enter** to accept the default print device (in this case the printer port called PRN), or enter another print device (for example LPT1, LPT2, or LPT3) and then press **Enter**.

You can carry on working with DR DOS while your file is printing, although your work may be slowed down if you are printing a very large file.

3.6 Keeping track of where you are

Using the PROMPT command you can keep track of where you are in your tree structure. PROMPT lets you change the default DR DOS prompt, for example C>, to show the directory you are in. Do this by typing:

```
C>PROMPT $P$G
```

The symbol $P says show the current directory and the symbol $G means show the greater-than sign (>). This is the usual way in which the prompt is set up on hard disks.

If you enter this command and then change to a directory called **sales** from the root directory you would see:

```
C:\SALES>
```

If you changed again to a subdirectory of **sales** called *foreign*, you would see:

```
C:\SALES\FOREIGN>
```

so you would always know which directory you are in.

You can put the PROMPT command into your **autoexec.bat** file, so that it is invoked every time you start up your computer. How to do this is described in Chapter 11, "Customizing your system".

Note

This is just one of the many ways in which you can change your default prompt using the PROMPT command. See Chapter 10, "DR DOS commands" for more information.

4

Protecting your information

DR DOS offers two distinct methods of protecting the information in your files.

- □ You can protect your information by making copies of your files at regular intervals. Then, if a file is accidentally erased or becomes corrupt, you have a copy of the original to refer to. This is called *taking a backup*.
- □ You can also protect your information from other people reading or changing it, by applying passwords to the files.

This chapter describes these two methods.

DR DOS also offers a "secure system" option that prevents unauthorized access to your entire operating system, using passwords. This chapter also describes this system security.

4.1 Making backup copies of disks

It is extremely important that you take regular copies of your work to safeguard against loss of data. Make it a part of your working routine to take periodic backups of your files. Backups are copies of information taken from one disk and stored on another.

DR DOS provides three commands for this purpose:

- ❑ For the hard disk, the BACKUP and RESTORE commands. These two commands always work together.
- ❑ For diskettes, the DISKCOPY command.

Backing up your hard disk

The BACKUP command enables you to copy all or part of a hard disk onto one or more diskettes. The original copy remains on the hard disk. You can then use your backup copies if the original is lost or corrupted, by restoring them using the RESTORE command. Any files backed up using the BACKUP command can only be used after you have restored them using RESTORE.

Establish a backup routine

When backing up your hard disk, you should try to establish a routine. Usually a backup of the entire hard disk is taken once every month or few months. Backing up the whole hard disk can be a lengthy task and requires a large number of diskettes. In between backing up the entire hard disk, you can back up any *modified* files using the various BACKUP command switches. How often you need to back up modified files depends on your working routine. You may need to backup files daily, or you may find once a week is enough.

Before you make any backups, keep in mind these important points:

- ❑ Do not backup disks affected by an ASSIGN, JOIN or SUBST command as they affect the way a disk is viewed by the system; always cancel these commands first (see Chapter 10, "DR DOS commands").
- ❑ If your files and subdirectories have passwords, you need to give them first before taking a backup (see section 4.3, "File attributes and the ATTRIB command").

❑ When you backup files onto a diskette, any files already on the diskette are overwritten, even if the diskette is not full. To avoid this happening there is a switch, /A, which tells DR DOS to preserve the files already on the backup diskette.

❑ Make a note of the exact BACKUP command syntax you use to backup a disk, as you will need it when you restore your data (see the section "Restoring BACKUP files").

Backing up the entire hard disk

Before backing up a hard disk you need to make sure you have plenty of new, blank diskettes to hand. To backup a full 20 Mbyte hard disk you may need up to 60 formatted diskettes, though this number will vary according to the capacity of your diskette drive.

To backup your hard disk onto diskettes type:

```
C>BACKUP c: a: /S
```

The /S switch tells BACKUP to backup all subdirectories as well as the root directory (if you did not specify /S, only the root directory would be backed up).

You are asked to insert the first backup diskette into diskette drive A:

```
Backing up from C:\ to A:
Insert disk number 001 in drive A:
All files in the root directory of the
destination disk will be deleted
Strike a key when ready
```

The first diskette you insert is “backup diskette 001". Press any key to start the backup procedure.

When the first diskette is full, you are prompted to insert the next diskette (002). Remove the first diskette and label it with its backup number (001), the date and the exact BACKUP command syntax you used. (It is important that you label your backup diskettes in the sequence they are created; this is due to the way in which the RESTORE command works. Also you will need to use the same syntax you used to backup your files to restore them.)

Insert the next diskette, 002, and press any key to continue. When this diskette is full, remove it, label it, and then insert diskette 003. Continue doing this until the DR DOS system prompt is returned. This may take some time depending upon the size of your hard disk.

You now have a full set of backup diskettes for your hard disk. Now write-protect your backup diskettes, using write-protect tabs (see Chapter 1, "Getting started with DR DOS", section 1.1).

Backing up modified files

To backup files changed since your last backup was taken, you use the switch /M (M means modified) as follows:

```
C>BACKUP c: a: /M
```

If you intend to backup your modified files onto an existing backup diskette, you must specify the switch /A in the command line:

```
C>BACKUP c: a: /M /A
```

This is because backing up onto diskettes erases all the files that are already stored there. The /A switch tells BACKUP to preserve the files that already exist. You can also backup files modified since a specified date onto your backup diskettes using another switch, /D:date. For example:

```
C>BACKUP c: a: /D:8-3-90
```

backs up all files changed since March 8th 1990. You can only separate the items in the date with hyphens. The date format you use depends on what country your system is set up for.

Backing up directories

To backup a specified directory and all its associated subdirectories, use the /S switch in the same way you used it to backup the root directory and all its associated directories. For example to back up the directory c:\accounts and all its subdirectories, type:

```
C>BACKUP c:\accounts a: /S
```

To backup a single file, specify the filename in the command line. For example:

```
C>BACKUP c:\accounts\report.txt a:
```

backs up the file **report.txt** onto a diskette in drive A:.

Using wildcard characters

You can use wildcards in the command line. For example:

```
C>BACKUP *.doc a:
```

backs up all files in the current directory with the extension **.doc** onto a diskette in drive A:.

Restoring BACKUP files

The RESTORE command is always used to recover files stored using BACKUP. It works in a similar way to BACKUP. It copies backup files back into a usable form, usually onto the disk from which they originally came. If you labeled your diskettes correctly when you backed them up, you just insert them in the correct order. You need to use the equivalent syntax you used when backing up your files to retrieve them, so if you want to restore your complete hard disk backup type:

```
C>RESTORE a: c: /S
```

(The corresponding BACKUP command is BACKUP c: a: /S.)

You are prompted to insert the first RESTORE diskette, which would be the first diskette you backed up onto:

```
    Insert disk number 001 in drive A:
    Strike a key when ready
```

Press any key to continue. RESTORE displays the name of each file as it restores it to the disk. You are prompted as above until all the diskettes are restored, and then the DR DOS system prompt is returned.

To restore the directory **accounts** and its associated directories type:

```
C>RESTORE a:\accounts c: /S
```

The RESTORE command has many switches that are similar to those of the BACKUP command. See Chapter 10, "DR DOS commands", for a description of the full range of switches available with the RESTORE command.

Checking what is on a BACKUP disk

The RESTORE command has a helpful switch /R that reports on what files would be restored without taking any action. This is very useful for checking what files are on your backup diskettes.

More on BACKUP and RESTORE

The BACKUP and RESTORE commands are far more powerful and flexible than outlined here. For more information see Chapter 10, "DR DOS commands" which gives a listing of all the DR DOS commands their various switches, and their uses.

Copying diskettes

The DISKCOPY command duplicates entire diskettes. We have already described DISKCOPY in Chapter 1, "Getting started with DR DOS". The basic format of the command for single diskette drive systems is:

```
C>DISKCOPY a: a:
```

For twin diskette systems type:

```
C>DISKCOPY a: b:
```

In either case, follow the screen prompts. DISKCOPY will automatically format your diskette for you if it is not already formatted.

Caution DISKCOPY overwrites *anything* that exists already on the destination diskette, that is, on the diskette you are copying to. If you wish to add files to a diskette without necessarily overwriting existing data, use the COPY or XCOPY commands.

Always write-protect diskettes containing important information.

For more information on this command, see Chapter 10, "DR DOS commands".

4.2 Protecting files with the password command

What is a password?

A password is a set of characters you need to type in to gain access to information. In DR DOS you can apply three levels of password protection to a file:

- ❑ The highest level prevents anyone from even reading the file unless they issue the password.
- ❑ The next level allows other users to read the file but not to change it without the password.
- ❑ The lowest level allows users to read and change the file, but not to rename or delete it without first giving the password.

You can also password-protect directories. If you do this, all the files and directories within the protected directory are completely inaccessible without the correct password.

You set passwords using the command PASSWORD. Passwords can be up to eight characters long. The PASSWORD command has the following format:

```
PASSWORD filename switch:password
```

PASSWORD switches

The command has three switches that correspond to the levels of protection described above:

Command switches

/R	The highest level. /R sets the password attributes R, W and D. You need the password to even read the file.
/W	The next level down. /W sets the password attributes W and D. You need the password to edit, rename or delete the file.
/D	The lowest level. /D sets the password attribute D. You need the password to delete or rename the file or change the file's attributes (for example, read-only; see section 4.3, "File attributes and the ATTRIB command").

Note

A full description of all the PASSWORD switches can be found in Chapter 10, "DR DOS commands".

Setting passwords

To set the password **lucky** with the highest level of protection to the file **wages.txt**, you type:

```
C>PASSWORD wages.txt /R:lucky
```

To set the password so that users can read and write to **wages.txt**, but not delete or rename it unless they issue the password, type:

```
C>PASSWORD wages.txt /D:lucky
```

Be careful when choosing a password. Make sure it is not too obvious to anyone attempting to work it out, or alternatively one that is too difficult to remember!

Issuing passwords

To issue a password after you have allocated one to a file, you type a semi-colon immediately after the filename followed by the password.

For example, the TYPE command is used to display a text file on the screen. To display the password-protected file **wages.txt**, you would type:

```
C>TYPE wages.txt;lucky
```

To copy the file **wages.txt** to a new file **archives.txt** on drive A:, enter the command:

```
C>COPY wages.txt;lucky a:archives.txt
```

Note that the copy, **archives.txt**, is *not* password-protected.

Password-protecting directories

In the same way you give files passwords, you can also allocate them to directories using other PASSWORD switches. To assign a password to a path, you use the password switch /P:. For example, to set a password on the directory **c:\clients\personal** type:

```
C>PASSWORD c:\clients\personal /P:private
```

To access this path now you have to issue the password **private** as follows:

```
C>CD c:\clients\personal;private
```

Global system passwords

With PASSWORD you can also set a universal or "global" password, that is, a password that DR DOS automatically tries when you attempt to access a file or directory that is password-protected.

This saves you having to give the same password every time you try to access one of the files in the group.

This can be useful if you are working with an application that does not recognize passwords.

After you have issued the global password it is tried against each password-protected file you try to access. The global password remains in effect until it is switched off explicitly by you, or until the computer is restarted.

For example, if you had a number of files all protected with the password **leopard** and you wished to set the global password, you would type:

```
C>PASSWORD /G:leopard
```

Now you would not have to issue the password **leopard** for the files to which it applies.

Note

Remember if you leave your computer unattended, to switch off the global password by entering the command PASSWORD /NG.

Removing passwords

From a file

To remove a password from a file, use the format:

```
PASSWORD filename /N
```

For example, to remove the password **lucky** from the file **wages.txt**, enter:

```
C>PASSWORD wages.txt /N
```

You are prompted to give the password:

```
     C:\Wages.txt ...file password?
```

Type in the password **lucky** (it is not shown on the screen) and press **Enter**. The password is then deleted, and the system prompt returned.

Alternatively, you can specify the password in the command line, for example:

```
C>PASSWORD wages.txt;lucky /N
```

From a directory

To remove a password from a directory or path, use the format:

```
PASSWORD dirpath;password  /NP
```

For example to remove the password **private** from the path **c:\clients\personal**:

```
C>PASSWORD c:\clients\personal;private /NP
```

A note on password-protection

The DR DOS password protection scheme is an easy and convenient way of preventing unauthorized access to data, or accidental loss of important information. However, you should only use password protection as a supplement to the physical security provided by keeping your computer and diskettes securely locked away if they have sensitive information stored on them.

4.3 File attributes and the ATTRIB command

In addition to passwords, other pieces of information called *attributes* can be associated with a file. These attributes are described below. When you set an attribute it means it is *active* or switched on; when you reset it, it is *non-active* or switched off.

Attribute	Meaning
A	Archived. This attribute is set whenever a file is changed. It may be reset by, for example, BACKUP, FILELINK, REPLACE, XCOPY (and other commands) when a file has been "archived", as well as by the ATTRIB command.
H	Hidden. Setting up the hidden attribute makes the file invisible to the user. For example, it will not appear in standard directory listings.
R	Read-only. When set, you can only read a file. You cannot write to it, or delete it.
S	System. Files with the system attribute set are only displayed when the /S option is used with the DIR command.

To set an attribute, use the following format:

```
ATTRIB +attrib [@]wildspec
```

For example, to set the hidden attribute for all **.let** files in the DOC directory, you would type:

```
C>ATTRIB +H doc\*.let
```

To reset an attribute, use the format:

```
ATTRIB -attrib [@]wildspec
```

Attributes can provide a means of protecting your files. However, this method of protecting files is not as safe as using passwords, as several commands have switches that allow you to alter read-only files. The safest way to protect a file or directory is to use the PASSWORD command. Setting a password with a /D level of protection is sufficient to stop another user from removing or changing an attribute, for example changing a read-only file to not read-only.

4.4 Securing the system

DR DOS offers an option to secure your entire operating system. When you choose the system security option during INSTALL, or at any later time when you run SETUP, you can specify two levels of password-protection; that is, one of the passwords must be typed in by the user in order to gain access to DR DOS on your computer. This protects your data, and your fixed disks, by preventing unauthorized access to them.

DR DOS system security uses two different passwords:

- ❑ The *Us*er Key password should be used for day-to-day access to DR DOS. Note that you should change your User Key frequently to ensure security.
- ❑ The Master Key password should be used in addition to the User Key, for enabling and disabling the system security feature.

When you install system security, your **config.sys** file is automatically updated to run the LOGIN command every time you switch on your computer. LOGIN displays a screen that prompts you for the required password.

Note

You can change the contents of the LOGIN screen by editing the files **login.txt** and **login.err** (in directory **c:\drdos**) using the DR DOS EDITOR.

You can also use your User Key and Master Key passwords with the LOCK command, to *temporarily* lock and then unlock the system. Refer to Chapter 10, "DR DOS commands" for a description of LOCK.

LOGIN command

DR DOS runs LOGIN automatically when you install security. However, you can use the LOGIN command from the command line as follows:

- ❑ Type `LOGIN` to display the directory currently containing the system security files.

- ❑ Type `LOGIN [dirpath]` where *dirpath* specifies the directory to contain the system security files, including **login.txt** and **login.err**. Use this option to change the location from the default (**\drdos**).

LOGIN also has a /? or /H option.

Installing system security

When you run INSTALL or SETUP, you are asked whether or not you want to install system security. Answer **Yes** to install it; you are also prompted for the User Key and Master Key passwords.

Note

When you make your system secure, it is no longer possible to access your hard disk when you boot from diskette until the LOGIN program is run and you give the Master Key password. This prevents any other operating system from booting and gaining access to your files.

If you install system security *and* TaskMAX, the LOCK command is also loaded automatically. Refer to the LOCK command description in Chapter 10, "DR DOS commands" for information.

Disabling system security

You can disable the system security feature at any time by running SETUP. You should use the Master Key password for disabling (and enabling) security. This provides an additional level of system security.

5

Recovering your information

DR DOS provides several methods of recovering deleted data. The chances of successfully recovering a deleted file depend on which method DR DOS is able to use. This chapter provides an overview of file recovery and the recovery aids you can use to make this easier and more successful. For more information about how to use the DR DOS file recovery commands, see the command descriptions in Chapter 10, "DR DOS commands".

5.1 File Recovery Aids

Before you actually start deleting files with DR DOS, you should know how to use two utilities that DR DOS provides to aid file recovery after delete operations: DISKMAP and DELWATCH. File recovery in DR DOS without either of these two methods is referred to as *unaided.*

DISKMAP

DISKMAP is a utility that copies and saves the File Allocation Table (FAT) of a disk. The FAT is like an index to a disk that records the location and size of all files on the disk. Running DISKMAP on a regular basis means that you always have a recent copy of the FAT, and DR DOS can therefore retrieve information about a deleted file that it would not normally be able to retrieve. Refer to the explanation about DISKMAP in Chapter 10, "DR DOS commands".

DELWATCH

DELWATCH is another utility provided in DR DOS that actually saves deleted files on the disk. Instead of removing a deleted file, DELWATCH simply marks it as a *pe*nding delete file. The file appears to be deleted, however. Using DELWATCH means that you are certain of successfully recovering a file deleted in DR DOS. Refer to the explanation about DELWATCH in Chapter 10, "DR DOS commands".

5.2 Recovering deleted files (the UNDELETE command)

Use the UNDLETE command to recover deleted files, regardless of which method DR DOS is able to use: DELWATCH, DISKMAP, or unaided. UNDELETE automatically determines and displays the method available.

By typing UNDELETE on the command line without a file specification, you can use a full-screen menu version of the UNDELETE command. The menu contains a list of the deleted files in your current directory and a series of command options from which to choose. The list displays information about deleted files. You can select any of the following menu options:

- ❑ Start UNDELETE for a selected file.
- ❑ Change the current disk drive.
- ❑ Select a file or group of files.
- ❑ Exit from UNDELETE and return to the command prompt.
- ❑ Sort the files in the list by name, extension, date, time, size, or recovery method available.
- ❑ Display different kinds of online help.

By typing UNDELETE on the command line with a file specification, you start UNDELETE for the specified file.

For example, to recover a file called **report.txt** that you deleted from your current directory, you can do either of the following:

- ❑ Type UNDELETE to display the menu, select **report.txt** in the list of deleted files, and choose the `Undelete` option from the File menu.
- ❑ Type the following on the command line:

```
C>UNDELETE report.txt
```

You are prompted about proceeding before UNDELETE attempts to recover **report.txt**.

Refer to the explanation about UNDELETE in Chapter 10, "DR DOS commands" for more information about using the command and the menu.

6

Switching tasks

The DR DOS utility TaskMAX makes changing between applications faster and more convenient. It allows you to switch quickly and easily from within one application to another without exiting the first application. You do this by adding your applications as *tasks* to the TaskMAX menu, which you can display at any time. When you change to another application, the one you leave is frozen exactly as you left it and put to one side until you return; this way you do not lose your place. The current state of the application or task is swapped out to a swap file in memory, or to disk so that the same memory in your computer is available for you to run other applications which you can also put in the background.

TaskMAX has many advantages:

- ❑ You can switch between a maximum of 20 different applications quickly and easily.
- ❑ Applications you use most often can be made instantly available every time you use your computer.
- ❑ You can copy information between applications.
- ❑ You can run multiple copies of an application.
- ❑ TaskMAX makes the best possible use of your computer's memory by loading TSRs as tasks.

The following are examples of how TaskMAX can be useful:

- You may start preparing a document in the DR DOS EDITOR and find you need to check some information in a database. Normally you would exit EDITOR, start your database, check the information you need, exit your database, and restart EDITOR to update your document. Using TaskMAX you can speed up this process by adding the EDITOR and database as tasks to the TaskMAX menu, then you can switch between them with a few simple keystrokes.
- You may want to edit a table in a spreadsheet, but need information from a word processor file to complete it. Using TaskMAX, you can add the spreadsheet and word processor as tasks and, with a few easy steps, copy the information you need from the word processor file to the spreadsheet.

Terminate-Stay-Resident program

TaskMAX is a Terminate-and-Stay-Resident program (TSR); that is, a program which leaves some of its code in the computer's memory when you exit the program and return to the DR DOS command prompt. Therefore, unlike other utilities, it is not loaded from disk every time you use it; once loaded into memory it is always available, until you switch off your computer or remove it from memory using the `Remove TaskMAX` option.

6.1 Loading TaskMAX

You can install TaskMAX so that it is loaded automatically every time you switch on your computer, or you can load it directly from the command line.

Loading TaskMAX automatically

You may find it most convenient to install TaskMAX when you install DR DOS, or later using the SETUP program (see Chapter 11, "Customizing your system", section 11.1), so that TaskMAX is loaded automatically and available to you every time you use your computer. You can also use the SETUP program to modify certain TaskMAX defaults; see section 6.8, "Configuring TaskMAX", in this chapter.

Note

If you load TaskMAX this way, it may be necessary to load other programs before it. For more information see section 6.9, "TaskMAX and the DR DOS environment", in this chapter.

Loading TaskMAX from the command line

Note

If you load TaskMAX from the command line, you should run the SHARE command first. For more information about SHARE, see Chapter 10, "DR DOS commands".

To load TaskMAX from the command line type:

```
C>TASKMAX
```

A message is displayed saying that the DR DOS application switcher is loaded. You can now start using TaskMAX; see section 6.2, "Displaying the TaskMAX menu", in this chapter. Note the following:

- ❑ If TaskMAX is already resident in memory when you try to load it, you will see a message to this effect followed by a list of existing tasks.
- ❑ TaskMAX will remain resident until you switch off your computer, or remove it; see section 6.7, "Removing TaskMAX", in this chapter.
- ❑ There are several parameters you can use to configure TaskMAX from the command line; see section 6.8, "Configuring TaskMAX", in this chapter.

6.2 Displaying the TaskMAX menu

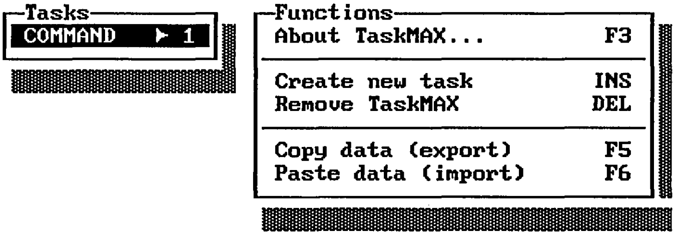
To display the TaskMAX menu, press **Ctrl+Esc** (these are known as the TaskMAX *activation* keys).

Note

The activation keys can be changed via SETUP; the default keys are referred to in this chapter.

The TaskMAX menu appears in the top right corner of the screen. If you are running a graphics application (for example a drawing package) when you press **Ctrl+Esc**, the screen is cleared and the menu is displayed. When you exit the menu you are returned to the application screen.

The TaskMAX menu has two parts: the left side is the *task* list where all the tasks you add are displayed and the right side contains the TaskMAX options. There are five options: `About TaskMAX`, `Create new task`, `Remove TaskMAX`, `Copy data (export)` and `Paste data (import)`.



Space for tasks

Selecting the `About TaskMAX` option shows you how much space the tasks you have added have used. This is useful when you are adding lots of tasks as you can see approximately how much space you have left.

When you first display the menu, the task list always contains one task, `COMMAND`, which represents your current DR DOS task. Remember that a task is simply something you want to switch to, for example an application.

Selecting an option

To select an option from the menu, press the "hotkey" shown to the right of the item; for example, to select `Remove TaskMAX` press **Del**.

Moving around the task list

To move around the task list, use one of the following:

- ❑ ↑ ↓ keys
- ❑ **Spacebar** (which moves down the menu, then jumps to the top)
- ❑ Tab and **Shift+Tab** keys (which move forward or back a task respectively).

Selecting a task

To select a task, do one of the following:

- ❑ Move the highlight bar to the task and press **Enter**.
- ❑ Type the number shown to the right of the task and press **Enter**. To select a number higher than 9, for example 14, type 1 then 4 to represent 14, and then press **Enter**.

Note

If you are using a non-enhanced keyboard, instead of pressing **Ins** and **Del**, you can press **F2** and **F4**, respectively.

Cancelling the menu

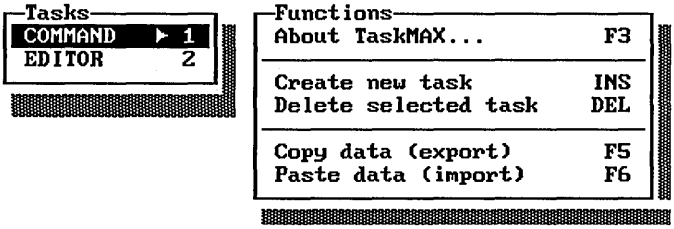
To cancel the TaskMAX menu, press **Esc**; note that this does not remove TaskMAX from your computer's memory. To display the menu again, press **Ctrl+Esc**.

6.3 Adding tasks

To add a task:

1. Display the TaskMAX menu by pressing **Ctrl+Esc**.
2. Select `Create new task` by pressing **Ins**. You are returned to the DR DOS command line.
3. If you now display the TaskMAX menu again, you will see that task 2 is shown as `COMMAND`. This, in effect, represents a second DR DOS task. Notice that the option `Delete selected task` now appears on the menu (replacing `Remove TaskMAX`). Press **Esc** to cancel the menu.
4. Start the application you want to add as a task in the way you normally start it. For example, you may need to change to a different drive or directory or you may start your applications using a batch file. For practise, start the DR DOS EDITOR by typing Editor and pressing **Enter**.
5. The application you have just loaded is now installed as a task. To check this, display the TaskMAX menu from within the application by pressing **Ctrl+Esc**; the name of the application should appear in the task list as task 2. The taskname can be up to 8 characters in length; if the application's name exceeds this, TaskMAX will use an abbreviated version. If TaskMAX cannot determine a

name, you see Task *n* as the taskname, where *n* is the task number. You can change the tasknames that TaskMAX allocates by using the /N switch; see section 6.8, Configuring TaskMAX, subsection "Configuring TaskMAX from the command line", in this chapter.



```
Tasks                 Functions
COMMAND   ► 1         About TaskMAX...          F3
EDITOR      2
                      Create new task           INS
                      Delete selected task      DEL

                      Copy data (export)        F5
                      Paste data (import)       F6
```

Now continue to add applications you want to switch between. The tasks are listed on the menu in the order in which you add them. Note that you can only switch between applications added as tasks.

Maximum number of tasks

You can add up to 20 tasks, but try to keep to those you use most frequently. If you add the maximum number of tasks allowed, the `Create new task` option disappears from the TaskMAX menu, so you will not be able to add any more. The current task, that is, the task you are, or were last, using is indicated by an arrow in the task list. The tasks remain installed until you delete them, unload TaskMAX, or switch off your computer.

Checking there is space for tasks

If you try to add a task and there isn't enough space, a warning message will be displayed. If this happens you will have to remove an existing task first. If you try to swap tasks and there is not enough space you will also see a warning message. Save any changes made in your application and exit from it. When you add tasks, keep a check on the amount of space you have used up as shown on the TaskMAX menu. There must be enough space available to store your tasks when you are not using them. Graphics-based applications can require up to 800 Kbytes of space, whereas text-based applications need a lot less.

Where background tasks are stored

If you have expanded or extended memory on your computer, you can configure TaskMAX so that background tasks are stored there; see section 6.8, "Configuring TaskMAX", in this chapter. This can speed up the swapping of tasks. If you do not have expanded or extended memory on your computer, or TaskMAX runs out of space in this memory, it will automatically store background tasks to a *swap file* on your hard disk.The swap file is called **taskmax.swp**. If necessary, you can specify where this swap file is stored; see section 6.8, "Configuring TaskMAX", in this chapter. The percentage figure on the TaskMAX menu, showing how much space your existing tasks have used, is a combination of how much memory and disk space has been allocated to your background tasks, plus the current size of the swap file.

Note

After a while you may find you use certain tasks more than others. You can configure TaskMAX so that these tasks are available all the time; see section 6.8, "Configuring TaskMAX", in this chapter.

6.4 Switching between tasks

You can switch from one task to another either from the menu or bypassing the menu.

From the menu

You can switch tasks from the menu in one of two ways:

❑ Display the TaskMAX menu by pressing **Ctrl+Esc**; move the highlight bar in the task list to the task you want; press **Enter**.

or

❑ Display the TaskMAX menu by pressing **Ctrl+Esc**; type the number shown to the right of the task in the task list; press **Enter**. To select a number higher than 9, for example 10, type 1 and then 0.

Bypassing the menu

There are two ways in which you can bypass the menu when you switch tasks:

❑ To switch to the next application listed on the menu without displaying the menu, press **Ctrl** and **+** on the numeric keypad. To move back a task, press **Ctrl** and **-** on the numeric keypad.

or

❑ If you know the number of the task in the task list, you can switch to it by pressing **Ctrl** and typing the corresponding number on the numeric keypad. For example, if you want task 12, press **Ctrl** and hold it down while you type 1 and then 2 to represent 12.

Remember you can only switch between applications that have been added as tasks.

Can't switch applications?

If TaskMAX cannot switch to an application, the computer sounds a beep; try again. If there is still a problem, section 6.10, "Important notes on using TaskMAX", may help. Otherwise, exit the application and then switch tasks using TaskMAX.

For a list and explanation of general system error messages, see Appendix A, "Error messages and troubleshooting".

6.5 Copying data

Using TaskMAX you can copy *a rectangular block* of data between, or within, applications that have been set up as tasks. You mark the corners of the rectangle and copy the highlighted area. Note that you can paste *into*, but you cannot copy *from*, a graphics application or an application running in graphics mode.

Note

You can use a mouse to select from the TaskMAX menu and to mark a block of data during copy and paste operations, as long as you have a mouse loaded on your system. The following description of copying data using TaskMAX indicates when you can use the mouse instead of keys.

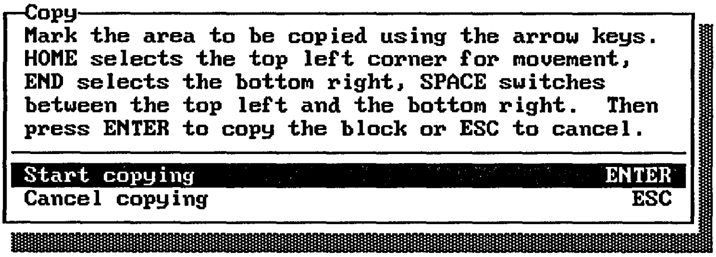
To copy a block of data:

1. Switch to the application you want to copy from; remember it must be loaded as a task within TaskMAX. Display on the screen the data you want to copy. Note that you can only copy what you can see on the screen at one time. If the data goes off the screen, you will have to copy it in two sections.

2. Display the TaskMAX menu by pressing **Ctrl+Esc**.

3. Select `Copy data (export)` by pressing **F5** or clicking your mouse on `Copy data (export)`.

4. The TaskMAX menu is replaced by a box listing the keys you use to mark the rectangle to be copied. Note that you can also mark the rectangle with your mouse. Press **Enter** to continue or **Esc** to cancel.



```
Copy
Mark the area to be copied using the arrow keys.
HOME selects the top left corner for movement,
END selects the bottom right, SPACE switches
between the top left and the bottom right.  Then
press ENTER to copy the block or ESC to cancel.

Start copying                                 ENTER
Cancel copying                                  ESC
```

5. If you press **Enter**, the box disappears and the line of data at the cursor position is highlighted in reverse video (if you have a color monitor) or inverse video (if you have a mono monitor).

   You select the block to copy by shrinking or expanding the highlighted area on the screen, using your mouse or the cursor arrow keys. If you use the arrow keys, move the top left corner of the highlighted area first, using the ↓ and → arrow keys. When you have positioned this corner, press **End** to switch to the bottom right corner. Now use the ↑ and ← arrow keys to move this corner. Note that the top left corner remains *anchored* (or fixed) where you placed it. If, when you have finished, you find you want to adjust the position of the top left corner, press **Home** to move to it and use the arrow keys to alter it. You can switch back to the bottom right corner again by pressing **End**. Repeat this process until you have marked the area you require.

(Note that you can also use the **Spacebar** to switch between the corners.)

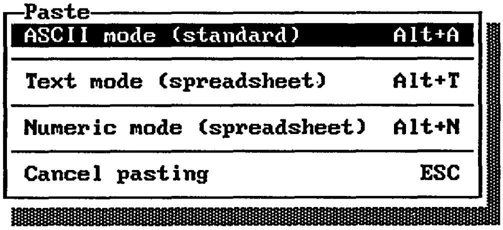
To cancel marking

To cancel marking, press **Esc**. If after cancelling, you still want to mark a block of data, repeat steps 2 to 5 above.

6 When you have finished marking, press **Enter** to copy the block to memory; the highlight disappears. If you want to paste the block to another point in the current task, position the cursor where the copy is to be pasted. Otherwise, switch to the task you want to copy to and, if necessary, load the appropriate file; see section 6.4, "Switching between tasks". When the task is loaded, remember to position the cursor, using the cursor arrow keys, to indicate where you want the copy to be pasted.

7 Display the TaskMAX menu again. Select `Paste data (import)` by pressing **F6** or clicking on `Paste data (import)` with your mouse.

8 The TaskMAX menu is replaced by another menu listing three paste options (before you paste your data, you need to select a format that the destination task will accept).



Paste

ASCII mode (standard) Alt+A

Text mode (spreadsheet) Alt+T

Numeric mode (spreadsheet) Alt+N

Cancel pasting ESC

The paste format determines the character(s) at the end of every typed line. The three paste modes you can choose from are:

❑ `ASCII mode (standard)`

This *standard* mode will work with most applications (for example word processors). A carriage (or hard) return is added at the end of each typed line; this is the normal DOS end-of-line character.

The remaining format modes are specifically for use with spreadsheets.

❑ `Text mode (spreadsheet)`

Use this mode if you are copying a column of text into a spreadsheet. The block of data you are pasting will be prefixed by the ' character, so that the data is treated as left-justified text. A down arrow character (↓) is added at the end of each line so that the text fits neatly into a column of cells.

❑ `Numeric mode (spreadsheet)`

Use this mode if you are copying to a spreadsheet and you only want numeric values pasted. For example, you may have a table in a word processor from which you want to copy the numbers, but not the text. This mode ensures that all non-numeric text (apart from, for example, decimal separators and minus signs) are stripped away from each line before it is pasted into the spreadsheet. A down arrow character (↓) is added to the end of every line to force the next line down to the next cell. Bracketed numbers, for example ($100), are treated as negative values (that is -100).

To select the option you require, press the hotkey displayed to the right of the mode. For example, press **Alt+T** to select `Text mode (spreadsheet)`. You can also select an option by moving the highlight bar to your selection and pressing **Enter**, or clicking on your selection with a mouse. Press **Esc** to cancel.

9 You can copy the data to:

❑ As many different places in the file as you like, by re-positioning the cursor and selecting `Paste data (import)` again.

❑ Another file in a different application, by switching to the application, positioning the cursor and selecting `Paste data (import)`.

❑ An editor, change it and copy it again to another file, or store the data as a file and import the file.

❑ An ASCII text file, by switching to the DR DOS command prompt and typing:

```
C>COPY con filename.ext
```

Press **Enter** when you have finished typing the command and the cursor moves down to a blank line. Now paste the data as described above from step 7; at step 8, choose `ASCII mode (standard)`. When the data is pasted, press **Ctrl+Z** then **Enter** to store the file. You can now import this ASCII text file into an application. This can be useful if you are copying large blocks of data, because you can paste the "file" into an application using its own data "import" facilities.

The data copied is stored until you use the `Copy data (export)` option again, and it is then overwritten.

6.6 Deleting tasks

To delete a task:

1. Switch to the task you want to remove.
2. Exit the application running as that task.
3. Display the TaskMAX menu by pressing **Ctrl+Esc**. Note that the name on the menu for your application is replaced by `COMMAND` (representing the DR DOS command prompt).
4. As you have just used the application you want to delete, it is displayed as the current task; press **Del** to delete it. If you have not exited the application running as the task you want to remove and you have open files, you will be asked if you really want to delete the task. Type **Y** for yes, or **N** to cancel. Proceed with care, because if you delete a task with open files, data may be lost.

Note

When you delete a task, the task at the top of the task list becomes current and is displayed.

If there is only one task in the menu, the `Delete selected task` option is replaced by `Remove TaskMAX`; see section 6.7, "Removing TaskMAX".

Note

If you press **Ctrl+Alt+Del** while TaskMAX is installed and you have more than one task added, your computer will not reboot, and instead the current task is deleted. This is a safety measure to prevent the computer being rebooted while several applications are suspended in the background with open files. Try to avoid pressing **Ctrl+Alt+Del**, especially if you are working on a file at the time, because the application may be reading or writing to the file. If you do have files open and you try to remove TaskMAX, a message will appear asking you if you really want to do this.

6.7 Removing TaskMAX

To unload TaskMAX (that is, to remove it from the computer's memory):

1. Delete all the tasks you have added (see section 6.6, "Deleting tasks", in this chapter) until `Remove TaskMAX` appears on the TaskMAX menu.
2. Select `Remove TaskMAX` by pressing **Del**; you are returned to the DR DOS command prompt. TaskMAX is now no longer installed. (Note that removing TaskMAX this way does not delete the TaskMAX program files from your hard disk.)

6.8 Configuring TaskMAX

You can configure TaskMAX using the SETUP program or directly from the command line.

Configuring TaskMAX via SETUP

In the SETUP program (see Chapter 11, "Customizing your system") you can skip directly to the TaskMAX configuration screens.

SETUP allows you to:

- □ Change the activation keys you use to display the TaskMAX menu. The defaults are **Ctrl+Esc**. You are given a choice of key combinations to select from. You can change **Ctrl** separately from **Esc** (**Esc** is described as the menu "hotkey").
- □ Change the color of the TaskMAX menu. You are given a choice of color sets to select from.
- □ Configure TaskMAX so that background tasks are swapped to expanded or extended memory if you have an Intel 80286, 386, 386SX or i486-based computer with more than 1 Mbyte of RAM. This will make swapping tasks faster. Note that if you have this type of memory, use it to speed up the swapping of tasks rather than setting up a memory disk.
- □ Set a limit on the amount of expanded (EMS) memory used per task. If one application attempts to allocate all expanded memory to itself, this setting will override it and allocate the value you specify per task.
- □ Change the location on disk to which your tasks are swapped when suspended. The default is **c:\drdos\tmp**. You may want to swap files to a memory disk (see VDISK.SYS in Chapter 11, "Customizing your system") to increase the speed with which you can switch between applications; that is, the speed with which tasks are swapped in and out of memory. (You can create a memory disk using SETUP.)

Note

Ensure that the drive you set the swap path to has enough free space to store the tasks you want to load; see section 6.3, "Adding tasks", in this chapter.

Before making changes to the TaskMAX configuration remove TaskMAX first (see section 6.7, "Removing TaskMAX") then make your changes using SETUP and reload TaskMAX.

Note

For definitions of, and more information on using different types of memory with DR DOS, see Appendix C, "Memory" and Chapter 12, "Using MemoryMAX".

TaskMAX.ini configuration file

The SETUP program adds any necessary statements to the following:

- ❑ Your **autoexec.bat** file (see Chapter 11, "Customizing your system").
- ❑ A special configuration file called **taskmax.ini**, used only by TaskMAX. This file is stored along with the **taskmax.exe** program in the default directory **c:\drdos** or in the directory specified by an environment variable DRDOSCFG. DRDOSCFG is automatically set when you use SETUP and specify a directory for configuration files. (You can also set the DRDOSCFG variable yourself using a SET command.) The **Taskmax.ini** file is automatically created (or changed) by INSTALL (or SETUP).

Configuring TaskMAX from the command line

If you prefer, you can configure TaskMAX directly from the command line; note that you cannot alter the default keys or menu colors this way. The full syntax of the command is:

```
TASKMAX [/?|/H] [/D=dirpath] [/X[=nnnn]] [/E[=nnnn]]
[/L=nnnn] [/C command] [/N[:nn][name]] [/K:nn]
```

If you originally configured TaskMAX via SETUP, any changes made to the configuration at the command line are only effective during the copy of TaskMAX currently loaded.

TaskMAX command switches

The TaskMAX parameters are explained below. If you type an invalid switch, TaskMAX displays help text.

Command switches

/? or /H	Displays help text.
/D=*dirpath*	The programs you suspend in the background are swapped to the file in the specified location on your hard disk. The default *dirpath* is **c:\drdos\tmp**. Ensure that the drive you set your swap path to has enough free space to store the tasks you want to load.

Command switches

You may want to swap files to a memory disk (see VDISK.SYS in Chapter 11, "Customizing your system") to increase the speed with which you can switch between applications; that is, the speed with which tasks are swapped. If your C: drive is nearly full, and you have a D: drive, you may want to swap tasks to a directory there.

To calculate the size of memory disk, load all the applications you would typically use and check the size of the swap file created on the disk. Set up the memory disk to be slightly larger than the size of the swap file and then configure TaskMAX to use the memory disk.

Note that /D is only applicable when you are loading TaskMAX.

/X[=*nnnn*] Swaps your background tasks to expanded memory, where *nnnn* is the size of the expanded memory in Kbytes. If you type /X on its own, all expanded memory is allocated.

Entering /X=0 disables use of this type of memory (for example, you may want to reserve all expanded memory for your applications).

Note that the /X switch is only applicable when you are loading TaskMAX.

/E[=*nnnn*] Swaps your background tasks to extended memory, where *nnnn* is the size of the extended memory in Kbytes. If you type /E on its own, all extended memory is allocated. Entering /E=0 disables use of this type of memory (for example, you may want to reserve all extended memory for your applications).

For both the /X and /E switches, if you enter a value that exceeds the amount of memory available, the maximum is used. For example, if you type /E=3000 and only 2 Mbytes of memory are available, 2 Mbytes are allocated.

Note that the /E switch is only applicable when you are loading TaskMAX.

Command switches

/L=*nnnn*

Allows you to set a limit on the amount of expanded (EMS) memory used per task. This means that if one application attempts to allocate all expanded memory to itself, this switch will override it and allocate the value you specify per task. The value you specify will apply to all tasks you add. For definitions of, and more information on using, different types of memory with DR DOS, see Appendix C, "Memory" and Chapter 12, "Using MemoryMAX". See also section 6.10, "Important notes on using TaskMAX", in this chapter.

Note that TaskMAX must already be loaded before you use the /L switch.

/C *command*

Allows you to add a task to the TaskMAX menu directly from the command line. *Command* is the application you want loaded as the task; you can load the application in the usual way. For example:

```
C>TASKMAX  /C  c:\wp\wordpro
```

loads the application Wordpro from the directory **c:\wp** and adds it as a task to the TaskMAX menu. If you normally load an application for example, Wordpro, using a batch file you can enter:

```
C>TASKMAX  /C  c:\bats\wp.bat
```

The batch file, **wp.bat**, is processed as normal and Wordpro is added as a task. Note that you can also add the /N switch to the batch file to preset the taskname of the application; see below.

Note that TaskMAX must already be loaded before you use the /C switch.

/K:*nn*

Removes the specified task without prompting you first, where *nn* is the number of the task in the TaskMAX menu.

Command switches

Note that TaskMAX must already be loaded before you use the /K switch.

/N[:*nn*] [*name*] Allows you to specify the name of a task, overriding the name allocated by TaskMAX. You can type up to 8 characters. The name you specify remains in place for the current TaskMAX copy. You can rename either the current task or a specific task. For example:

```
C>TASKMAX /N spread
```

sets the name of the current task to *spread.* The following example would rename a specific task, *task2* on the TaskMAX menu, to *spread*:

```
C>TASKMAX /N:2 spread
```

To restore the original TaskMAX taskname, type /N if it is the current task, or /N:*nn* if it is a specific task, for example:

```
C>TASKMAX /N:2
```

restores task 2 to its original taskname which is the name of the application's executable program in memory. Note that this only takes effect when you next switch tasks.

Note that TaskMAX must already be loaded before you use the /N switch.

Examples

The following command changes the default swap directory to drive E:, which in this case is a memory disk:

```
C>TASKMAX  /D=e:\
```

The following command limits the allocation of expanded memory for each task to 1 Mbyte:

```
C>TASKMAX  /L=1024
```

The following command allocates all extended memory for background tasks to be swapped to:

```
C>TASKMAX  /E
```

The following command disables use of expanded memory:

```
C>TASKMAX  /X=0
```

The next command loads the application Graphics as a task:

```
C>TASKMAX  /C  graphics
```

The next command renames task 5 in the task list to **wp**:

```
C>TASKMAX  /N:5  wp
```

and

```
C>TASKMAX  /N:5
```

restores the original TaskMAX taskname. Note that this only takes effect when you next switch tasks.

6.9 TaskMAX and the DR DOS environment

Global environment

DR DOS provides an environment for applications with facilities that they can make use of. For this reason it is often called the *global* environment. For example, it is where information from the PATH and PROMPT commands are recorded. This is the environment where TaskMAX is loaded.

Each task you add to TaskMAX inherits the global environment, including any drivers and TSR programs loaded there and available memory.

Anything you want to be shared by *all* tasks must be loaded *before* you load taskMAX. For example:

- ❑ TSRs including print spoolers (a print spooler allows you to print files without stopping the computer from performing other tasks at the same time; see the PRINT command in Chapter 10, "DR DOS commands"). This is very important because if you load a print spooler from within a task you will only be able to print using the spooler while the task is active.

- ❑ Network drivers. Note that logging, and mapping of printers and remote drives should be done *before* TaskMAX is loaded.
- ❑ Mouse drivers.

Note

It is advisable to keep items loaded globally to a minimum, because anything loaded before TaskMAX will reduce the memory available to all the applications loaded as tasks.

Local environment

When you load an application as a task with TaskMAX, it creates its own *local* environment from within a task; for example, if you load a TSR, only the local environment is affected (that is, you can only use the TSR from that task). This can be advantageous because the TSR will only use memory from the task it was loaded in.

If TaskMAX is loaded automatically when you switch on your computer, make sure any entries for other programs you want to load, or changes you wish to make to the global environment via autoexec.bat, are inserted before the TaskMAX entry, because DR DOS loads entries in the order they appear. You can use SETUP to do this or edit your **autoexec.bat** file directly using the DR DOS EDITOR. If you use a different editor, make sure you store **autoexec.bat** as a plain ASCII text file.

The DR DOS configuration file, **config.sys**, is processed before the **autoexec.bat** file when you start your computer. Any device drivers required by your system will be loaded when the **config.sys** file is processed. Note that you cannot load TaskMAX from the **config.sys** file.

6.10 Important notes on using TaskMAX

Note the following when using applications as tasks within TaskMAX.

Disk optimization and disk editor utilities

Always make sure TaskMAX is not loaded (see section 6.7, "Removing TaskMAX") *before* using disk optimization programs or disk editors.

TaskMAX and command.com

Although TaskMAX is primarily designed for switching applications, you can also set up different DR DOS command interpreter programs (**command.com**) as tasks by selecting `Create new task` and loading another command interpreter. You can customize the local environment for each **command.com** task, giving several different command line interfaces; see COMMAND in Chapter 10, "DR DOS commands".

Note that if you load a second copy of command.com, before loading TaskMAX you will *not* be able to load TaskMAX from it. You can only load TaskMAX from the first copy of **command.com**.

TaskMAX and TSRs

When you load a TSR as a task, TaskMAX automatically recognizes it as a TSR so that even if you are not actually running it, the TSR stays loaded as a task and appears on the TaskMAX menu; when you finish running the TSR, you will see a message indicating that it is still loaded and that you must remove it yourself if you no longer want to keep it as a task.

Allocating expanded memory

When some applications are loaded, they try to allocate all available expanded memory solely for their use. They in effect expect to be the only application loaded at the time. So if you load another application in another task, there may not be any expanded memory available for it. To counteract this, use the SETUP program or the /L:*nnnn* switch from the command line to limit the amount of expanded memory each task may allocate.

If you have 512 Kbytes of expanded memory allocated for use by applications and when loading TaskMAX you set a limit of 256 Kbytes using /L=256, the first task you add that uses expanded memory will take 256 Kbytes. The second task you load which uses expanded memory will also take up to 256 Kbytes. This means that if you load a third task that uses expanded memory, there will be no more available for it. Therefore, make sure the tasks you load *first* are those that need expanded memory.

Setting the number of open files

DR DOS normally expects only one application to be loaded at a time. If more than one application is being used via TaskMAX, you must increase the number of open files allowed per application as allocated by the FILES statement in the **config.sys** file (see Chapter 11, "Customizing your system"). If the value is not large enough, you will see an error message similar to this:

```
Can't start [application]: file limit in config.sys too low
```

The value you enter in **config.sys** depends upon the number of files that need to be opened simultaneously by each application. For example, if you add a word processor and a spreadsheet as tasks within TaskMAX, and each requires a value of 20 files, the value in **config.sys** must be at least 40. If a graphics program requires 20 files, and an editor 10, the value in **config.sys** must be at least 30. Read your applications' documentation to find the optimum number of open files required. Note that you can change the FILES statement using SETUP. However, do not *over* specify the number of open files because each file entry has a portion of memory assigned to it.

Problems when running multiple copies of an application

If you open more than one copy of an application in TaskMAX and you experience problems, for example you see an error message like this:

```
overlay files cannot be opened
```

or if other system errors occur, it may be because:

❑ The application has been designed to believe it is the only application running. Use the DR DOS ATTRIB +R command to set the application's **overlay** and **.exe** files to read-only. Your application's documentation should tell you which files these are. (Overlay files are used by an application when it will not fit into memory in one go. The application's main file is loaded and the remainder is divided between overlay files which are loaded into memory as and when required.)

or

❑ Some applications create temporary files with fixed names. For example, some word processors create a temporary backup file of a file you have created, so if you cancel changes it can restore the original. If this is the case, and you try to load a second copy of the application, it may interfere with the temporary files belonging to the first copy.

Read your application's documentation to find if there is an environment variable you can use to override where these files are created. If one exists, you can use it to specify a different location for each of the temporary files.
If the application creates the temporary file in the local directory where it has been loaded, try loading each new copy of the application from a different directory.

Problems switching tasks

If you cannot display the TaskMAX menu in, or switch from, an application you have added as a task, the application may have taken control of the keyboard from TaskMAX. Instead of TaskMAX interpreting the keys that you press, the application is intervening and reading your keystrokes directly.

Using SETUP, change the TaskMAX activation key, **Ctrl**, to a different key or key combination; see section 6.8, "Configuring TaskMAX". The preferred alternative is **Ctrl+Left-shift** (the **Left-shift** key is the **Shift** key on the left side of your keyboard). If this does not help, the application you are trying to switch from may have a "system" option that selects the DR DOS command line processor, **command.com**. If so, selecting this option should enable you to switch tasks from the application; read your application's documentation.

When you switch back to this application again, you need to type EXIT at the DR DOS command prompt to return to your application again. If you restart the application at the DR DOS command prompt, you will have two copies running in the background.

Using communications software

If you are using communications software like electronic mail or file transfer, make sure you do not switch tasks while the communications software is sending or receiving data via the serial port on your computer because data might be lost. Use your application to stop the remote card sending data before you switch tasks.

7

The EDITOR

THIS chapter describes how to use EDITOR, a full-screen editor which you can use to create and edit text files. You can use it with any keyboard, and there are shorthand forms for some of the common commands. You can use EDITOR to:

- □ Create a new file.
- □ Browse through a file.
- □ Edit a file.

7.1 Starting EDITOR

There are two ways you can start EDITOR from the command prompt:

1. Type EDITOR followed by the name of the file you want to edit or create and press **Enter**, for example:

   ```
   C>EDITOR myfile.txt
   ```

   When you start EDITOR it looks for the file you specified (in the above example, **myfile.txt**) on the current drive, unless you specify a different drive or pathname before the name of the file.

   If EDITOR finds the file, it displays it on screen. If EDITOR cannot find the file it displays a message such as:

   ```
   c:\myfile.txt . . . file not found
   Create new file (Y/N)?
   ```

   Type Y to create the file, or N to go to the EDITOR title screen where you can type in another filename, or press **Esc** to leave EDITOR.

2. Type EDITOR and press **Enter**. This takes you directly to the EDITOR title screen.

   Type in a filename and press **Enter**, or press **Esc** to exit from the title screen back to the command prompt.

7.2 Quitting EDITOR

You can quit editing a file in three ways:

- ❑ To save your edited file and exit directly, press **Ctrl+KX**.
- ❑ To save your edited file and return to the EDITOR title screen, press **Ctrl+KD**.
- ❑ To abandon any changes made since the last time the file was saved and return to the EDITOR title screen, press **Ctrl+KQ**.

EDITOR asks you if you really want to quit without saving your file, unless you have not made any changes. You are then returned to the EDITOR title screen, where you can press **Esc** to quit EDITOR. **Ctrl+KQ** is useful if you want to abandon a file that you opened by mistake.

Note

You can press **Ctrl+R** at the EDITOR title screen to display the previous filename edited, and press **Enter** to go back to editing that file.

7.3 Entering text

To enter text, type as you would if using a word processor or typewriter except, when you reach the end of a line, press **Enter** to start a new line. This is because in EDITOR there is no "word wrap"; it does not automatically start a new line unless you tell it to.

Try entering the following text, remembering to press **Enter** at the end of each line:

```
REM this batch file changes to the Wundawrd directory
REM and starts my word processor
cd \wundawrd
ww
```

This text creates a simple batch file used to start a word processor called Wundawrd. A batch file is a quick way of carrying out a series of frequently-used commands, you simply type the batch file name at the command line and all the commands in the file are carried out for you; see Chapter 8, "Redirecting input and output" for more information. REM is a batch file subcommand.

Notice as you type, at the top of the screen EDITOR keeps track of how many characters and columns you are entering:

```
    Chr=108 Col=3
```

It tells you exactly which character and column the cursor is on. Also, your directory path and filename are displayed at the top of the screen.

Using the Tab key

You can use the Tab key to move the cursor across the page and insert text at set points. The tab stops are preset to every eighth column across the screen. You can think of the **Tab** key as meaning "add space characters until you reach the next tab stop".

Moving text down a line

If you press **Ctrl+N** when editing text, the rest of the line at the cursor is taken down to the next line, while the cursor remains where it is. For example, if the cursor was positioned after "starts" in this line:

```
REM and starts my batch file
```

and you press **Ctrl+N**, "my batch file" would move to the next line, like this:

```
REM and starts
my batch file
```

while the cursor is still positioned after "starts".

Entering control characters

You may wish to put control characters in a file, for example to control a printer. You cannot do this directly as EDITOR interprets control characters as commands.

However, if you precede the control character by an additional **Ctrl+P**, EDITOR understands that your entry is in fact to be placed in the file you are editing.

For example, to insert **Ctrl+A** into a file, you would press **Ctrl+P** followed by **Ctrl+A**.

Entering character codes

You can enter character codes directly into a file by pressing **Ctrl+QN**. This can be useful, for example, if you want to create graphics effects, and is equivalent to the effect you can obtain on some keyboards by pressing the **Alt** key together with a key on the numeric keypad. When you press **Ctrl+QN**, EDITOR prompts you for the decimal or hexadecimal code of the character you require. If you want to enter a hexadecimal code, precede it with the characters 0X.

Getting online help

You can get help in EDITOR by pressing **F1** or **Ctrl+J.** To exit from the help screen, press **Esc.**

7.4 Moving the cursor

You can move the cursor around your file in steps of one character, one word or one page, or you can jump directly to the start or end of the file.

To move the cursor one character

❑ To the right, → or **Ctrl+D**.

❑ To the left, ← or **Ctrl+S.**

❑ Up one line, ↑ or **Ctrl+E.**

❑ Down one line, ↓ or **Ctrl+X**.

Note

On some keyboards the four keys **D**, **S**, **E** and **X** form an approximate diamond shape on your keyboard, sometimes called the "cursor-control diamond". The position of these keys may help you to remember what they do.

To move the cursor one word

❑ To the left, **Ctrl+A**.

❑ To the right, **Ctrl+F**.

The keys **A** and **F** are also part of the "cursor-control diamond".

To move the cursor one page

❑ To the previous page, **PgUp** or **Ctrl+R**.

❑ To the next page, **PgDn** or **Ctrl+C.**

A "page" is the number of lines displayed on your screen, less eleven. On a 25-line display, pressing **PgDn** moves the cursor fourteen lines down. Some of the previous page will still be displayed on the screen, to make it easier for you to find your place again.

To move the cursor to the start or end of a line

❏ To the beginning of a line, press **Ctrl+QS**.

❏ To the end of a line, press **Ctrl+QD**.

To move the cursor to the start or end of a file

❏ To the beginning of a file, press **Home** or **Ctrl+QR**.

❏ To the end of a file, press **End** or **Ctrl+QC**.

The cursor-control diamond

To retrace our steps, here are all the cursor movement control keys. Again note the arrangement of command letters into the "cursor-control diamond" to help you remember these commands.

		Start of file			
		Home (**Ctrl+QR**)			
	Previous line			Previous page	
	↑ (**Ctrl+E)**			**PgUp (Ctrl+R)**	
Start of line	Word left	Char left	Char right	Word right	End of line
Ctrl+QS	**Ctrl+A**	← **(Ctrl+S)**	→ **(Ctrl+D)**	**Ctrl+F**	**Ctrl+QD**
	Next line			Next page	
	↓ (**Ctrl+X)**			**PgDn** (**Ctrl+C)**	
		End of file			
		End (**Ctrl+QC**)			

7.5 Inserting and overwriting text

EDITOR has two modes for entering text: insert (ins) and overwrite. When you start EDITOR, it is automatically in insert mode. The message `ins` is shown at the top rightmost corner of the screen.

To insert text, position the cursor at the point where the new text is to start and type it in. Existing text moves to the right to make room for what you type.

You can switch to overwrite mode by pressing **Ins** or **Ctrl+V.** Now when you type you will overwrite whatever the cursor is positioned on.

To switch overwrite mode off, press **Ins** or **Ctrl+V** again.

Note

Remember if `ins` is at the top of the screen, you are in insert mode. If `ins` is not displayed, you are in overwrite mode.

Inserting a file at the cursor

To read in a file, place the cursor where you want the file to be inserted, and press **Ctrl+KR**. You are prompted:

```
Name of file to read in?
```

Type in the name of the file you want to insert and press **Enter**. The file is inserted at the cursor. The remaining text is moved down to accommodate the new text.

7.6 Deleting text

Text can be deleted a character at a time, a word at a time or one line at a time. (You can also delete blocks of text; see section 7.7, "Working with blocks of text".)

Deleting a character

□ To delete the character the cursor is on, press the **Del** key or **Ctrl+G**.

The character is deleted from the screen, and the text to the right of the cursor shifts to the left to fill the gap.

If you press **Del** or **Ctrl+G** repeatedly, characters will continue to be deleted until you reach the end of the file.

□ To delete the character to the left of the cursor, press the **Backspace** key or **Ctrl+H**.

The character is deleted from the screen and the text to the right of the cursor shifts to the left to fill the gap.

If you press **Backspace** or **Ctrl+H** repeatedly, characters continue to be deleted until you reach the beginning of the file.

For example:

If you typed Wundaief instead of Wundawrd, place the cursor on i and press the **Del** key until the letters i, e, and f, are deleted. The line containing the mistyped Wundawrd would change like this, with the cursor positioned at the end of the line:

```
cd \Wunda
```

Now you type in the correct characters, that is wrd.

If you press **Del** at the end of a line, you delete the invisible "newline" character and bring up the next line. Similarly, pressing ← (**Backspace**) or **Ctrl+H** at the beginning of a line moves that line up to the end of the previous line.

Note

If you want to delete more than one character, it is usually quicker to use one of the key combinations described in the following sections.

Deleting a word

In EDITOR a "word" is defined as a string of characters that starts at the current cursor position and continues up to and including the first following space.

□ To delete a whole word, position the cursor at the beginning of the word to be deleted and press **Ctrl+T.**

□ To delete part of a word, place the cursor within the word and press **Ctrl+T.** The letters at and beyond the cursor are deleted. The text to the right of the cursor shifts to the left to fill the gap.

For example:

If the cursor is positioned at the start of the word "changes" in the following line:

```
REM this batch file changes to the Wundawrd directory
```

pressing **Ctrl+T** repeatedly changes the appearance of the line like this:

```
REM this batch file to the Wundawrd directory
REM this batch file the Wundawrd directory
REM this batch file Wundawrd directory
REM this batch file directory
REM this batch file
```

Deleting a line

Position the cursor anywhere in the line to be deleted and press **Ctrl+Y.** The line is deleted and lines below it move up to fill the gap.

For example:

If the cursor was positioned in the second line of the following text:

```
REM this batch file changes to the Wundawrd directory
REM and starts my word processor
cd \wundawrd
ww
```

Pressing **Ctrl+Y** changes the appearance of the display like this:

```
REM this batch file changes to the Wundawrd directory
cd \wundawrd
ww
```

Pressing **Ctrl+Y** again produces this result:

```
REM this batch file changes to the Wundawrd directory
ww
```

Pressing **Ctrl+Y** once more, produces this display:

```
REM this batch file changes to the Wundawrd directory
```

Note

On some keyboards, the keys **G**, **H**, **T** and **Y** (for deleting characters, words and lines) are next to each other and form an approximate square. You may find this helps you to remember these commands.

7.7 Working with blocks of text

You can mark a section or block of text and then move, delete, copy it, or store it into a separate file. A block of text can be any size (according to how much disk space you have got).

Marking a block of text

To mark a block of text, place the cursor at the start of the section you want to mark and press **Ctrl+KB**. The text is marked with `<B>`. Move the cursor to the end of the block of text and press **Ctrl+KK**. The end of the text block is marked with `<K>` and the text block is highlighted.

Moving a block of marked text

To move a block of marked text, place the cursor where you want the block to be moved to and press **Ctrl+KV**. The text is erased from its original position and moved to where the cursor is. The text moves up to fill the gap.

Deleting a block of marked text

To delete a block of marked text, press **Ctrl+KY**. The text is deleted and the text beneath it moves up to fill the gap.

Copying a block of marked text

To copy a block of marked text, move the cursor to where you want the block of text copied to, press **Ctrl+KC** . The block of text is copied to the cursor position, still marked. The original text remains where it is and is no longer marked.

Storing a block of marked text in a file

To store a block of marked text in a separate file, press **Ctrl+KW**. You are prompted:

```
Name of file to write to?
```

Type in the name of the file where the text is to be stored and press **Enter**. The file is created and the block of marked text is written to it. If the file already exists, you are asked:

```
File already exists, overwrite (Y/N)?
```

Type Y to overwrite the file, N if you want to keep the file as it is. If you answer N, no action is taken; press **Ctrl+KW** again and enter a different filename.

7.8 Saving your work

You can save files in three ways:

□ To save and exit from the EDITOR, press **Ctrl+KX.**

□ To save and continue editing your file, press **Ctrl+KS.**

□ To save your work and begin editing a new file, press **Ctrl+KD.**

This saves your work and updates your backup file, as described in section 7.9, "Automatic backup files".

7.9 Automatic backup files

When you open an existing file, a "working copy" of the file is stored in memory. The original file remains on your disk, unaltered.

When you save your edited file, the working copy is stored on the disk and is given the name of the original file. The old file becomes an EDITOR backup file with the extension **.bak**.

For example:

If you open the file **mytext.doc** and edit it, when you save the file the following happens:

- ❑ The original file **mytext.doc** is renamed to **mytext.bak**.
- ❑ The new version is stored with the name **mytext.doc**.

You can use backup files to recover files. For example if the current version is:

- ❑ found to be incorrect.
- ❑ accidentally erased.

If you accidentally deleted the file **mytext.doc**, you could revert to the **.bak** file by typing:

```
C>COPY mytext.bak mytext.doc
```

Note

To prevent any filename conflicts, EDITOR does not allow you to edit files with the extension **.bak**.

7.10 Technical details

- ❏ EDITOR produces ASCII files with no embedded specific control characters, so its output can be used for batch and similar files. It allows 8-bit characters to be entered and displayed.
- ❏ There is no size limit on the files you can edit, apart from the size limit imposed by the amount of disk space you have available.
- ❏ The **Enter** key inserts two characters into the file being edited: a carriage return (ASCII code 13) and a line feed (ASCII code 10), in that order. Note that a single **CR**, a single **LF** or a **CR/LF** pair is always treated as a single character when moving through the text.
- ❏ The **Tab** key inserts a character (ASCII code 09) into the file. On screen the **Tab** character inserts spaces from the cursor position to the next tab stop. A tab stop is preset at every eighth column across the screen.
- ❏ EDITOR does not insert any special End-of-File marker, but will recognize the End-of-File marker code 26 (ASCII **Ctrl+Z**) used by some older style editors.
- ❏ EDITOR provides automatic creation of **.bak** backup files. It does not allow you to edit files with a **.bak** extension.
- ❏ During use, EDITOR creates two temporary files in the same directory as the file you are editing. These are automatically deleted when you quit EDITOR.

Writing to the screen

EDITOR writes direct to the screen memory (rather than via BIOS calls) if you have an EGA or VGA display. For all other display types, EDITOR writes to the screen via BIOS calls. However, you can force EDITOR to write via BIOS calls regardless of the display type by invoking EDITOR with the /D switch:

```
C>EDITOR /D
```

7.11 Command summary

Table 7-1 Summary of EDITOR Commands

Keys	Function
Getting help	
F1 or **Ctrl+J**	Displays help text
Moving the cursor	
Home or **Ctrl+QR**	Start of the file
End or **Ctrl+QC**	End of the file
Ctrl+QS	Start of the line
Ctrl+QD	End of the line
← or **Ctrl+S**	Character left
Ctrl+A	Word left
→ or **Ctrl+D**	Character right
Ctrl+F	Word right
↑ or **Ctrl+E**	Previous line
↓ or **Ctrl+X**	Next line
Ctrl+W	Scroll text down one line
Ctrl+Z	Scroll text up one line
PgUp or **Ctrl+R**	Previous page
PgDn or **Ctrl+C**	Next page
Deleting text	
Del or **Ctrl+G**	Delete character
Ctrl+Y	Delete line
Backspace or **Ctrl+H**	Delete character to the left
Ctrl+T	Delete word to the right

Table 7-1 Summary of EDITOR Commands (continued)

Keys	Function
Working with blocks of text	
Ctrl+KB	Mark the start of a block of text
Ctrl+KK	Mark the end of a block of text
Ctrl+KV	Move a marked block of text
Ctrl+KY	Delete a marked block of text
Ctrl+KC	Copy a marked block of text
Ctrl+KW	Store a marked block of text in a file
Saving files	
Ctrl+KX	Save changes, exit
Ctrl+KD	Save changes, edit new file
Ctrl+KS	Save changes, resume
Ctrl+KQ	Abandon changes, edit new file
Other commands	
Ins or **Ctrl+V**	Switch between insert and overwrite modes
Ctrl+KR	Read in a file at the cursor
Enter	Insert a new line at the cursor
Ctrl+N	Insert a new line after the cursor
Ctrl+P	Insert a control character at the cursor
Ctrl+QN	Enter character codes directly into a file

8

Redirecting input and output

NORMALLY you enter information at the keyboard, which is the *standard input device* and the information is sent to your screen, which is the *standard output device.* In DR DOS you can redirect this flow of information and send it to a place other than your screen, for example to a file or printer.

In this chapter we will talk a lot about *input* and *output.* Input is simply what you enter at the keyboard, for example a command, and output is the *result* of that command.

8.1 Using redirection

Redirecting output

Instead of your information being sent to the screen you can send it to a file or print device. To do this you use the redirection symbol >.

For example, to redirect the listing that the DIR command produces directly to a file instead of displaying it on the screen, type:

```
C>DIR >mylist.doc
```

If the file **mylist.doc** did not exist, it would be created for you. If it does exist, the original will be overwritten, so take care.

Instead of the filename, you can enter a file specification:

```
C>DIR >a:\report\mylist.doc
```

if the file you want to redirect output to is not on the current drive.

You can also redirect the CHKDSK command's disk status report to a file:

```
C>CHKDSK a: >status
```

Redirecting output to or from a device

You can redirect output to a device rather than a file or command. Typical devices are printers, plotters and modems.

As discussed in Chapter 3, "Managing your information", devices are often named according to the communications port on the computer to which they are connected. Printers can be called “PRN” or “LPTn”, (where *n* could be 1, 2 or 3). Table 8-1 lists the device names and shows their use with redirection.

Table 8-1 Device Names and Their Use With Redirection

Device	What it is	Use
CON	User console (the keyboard and the screen).	Input and Output
AUX or COM1	First serial communications port.	Input and Output
COMn	Second and subsequent serial ports where n is 2, 3, 4, depending on the particular computer.	Input and Output
PRN or LPT1	First parallel printer.	Output
LPT2	Second parallel printer.	Output
LPT3	Third parallel printer.	Output
NUL	All output from this device is discarded. No input is returned from the device. It is used to discard unwanted output.	Input and Output

For details on how to set up a port for a particular device, refer to the MODE command in Chapter 10, "DR DOS commands". The form of the command for redirecting to a device is:

```
command >device
```

For example, the command:

```
C>DIR a: >prn
```

sends the DIR listing of drive A: to the printer rather than to the screen.

Appending output

You can append the output of a command to the *end* of a file, using the redirection symbol >>. For example, to add a file **index.dat** to the end of an existing file **book.txt**, type:

```
C>TYPE index.dat>>book.txt
```

The file **book.txt** now contains the contents of both files.

Redirecting input

Sometimes it is useful to make the contents of a file the input for a command. You achieve this using the redirection symbol <.

For example, to make the input to the SORT command (which sorts the contents of a file into alphabetical order) a file called **teleph.doc**, type:

```
C>SORT <teleph.doc
```

SORT takes the file, sorts its contents alphabetically and outputs the result to the screen.

Taking this a step further, to store the output of SORT into a file instead of just displaying it on screen, type:

```
C>SORT <teleph.doc>telelist.doc
```

SORT takes the **teleph.doc** file, sorts it into alphabetical order, and stores the result in the file **telelist.doc**.

8.2 Using pipes and filters

Pipes

Sometimes you will find it convenient to use the output from one program as the input to another. For example, the DIR command often produces a listing so long it cannot all be shown on the screen. In this case you can connect the output of DIR to the input of the MORE command, which displays output one screenful at a time. You can make this happen using a pipe; the piping symbol is | (a vertical bar). This is how you would invoke the DIR and MORE commands together:

```
DIR | MORE
```

You can pipe as many commands as you like. Piping is often used with filter commands (MORE is a filter command).

Note

When you pipe programs together, the data piped between them is stored by DR DOS in temporary files it creates on the default disk; there must be sufficient space on the disk for them and the disk should not be write-protected. (The temporary files are removed when the operation is complete.)

Filter commands

DR DOS has several commands that act as filters and can be piped together. A filter command reads your input, acts upon it in some way and then outputs the result, usually to the screen. The main filter commands are:

Command	**Function**
SORT	Arranges data into alphabetical order before displaying it on the screen.
MORE	Displays a file's contents a screenful at a time.
FIND	Extracts specified items of data for display.

For example, SORT is often used with the DIR command to sort the listing produced for example by filename or by size of file. Typing:

```
C>DIR | SORT /R
```

sorts the directory listing produced by DIR in reverse alphabetical order (the /R switch indicates it should be done from last to first instead of the default which is first to last).

The following command sorts the file **namelist.doc** using SORT. SORT arranges the file contents into alphabetical order before it is displayed on the screen:

```
C>SORT <namelist.doc
```

So, if the file **namelist.doc** contained the following names:

Henry
Sally
James
Sue
Fred

they are arranged into alphabetical order as follows:

```
fred
henry
james
sally
sue
```

In the following command line DIR and FIND are piped together to search for (FIND) and list (DIR) a directory called **drdos**:

```
C>DIR | FIND drdos

DRDOS       <DIR>        3-01-90   3:42p
```

Note

For more details on these commands, see Chapter 10, "DR DOS commands".

Combining pipes and redirection

You can also use piping and redirection together. For example:

```
C>DIR | SORT >dirsort
```

produces a directory listing which is sorted into alphabetical order, and then stored in the file **dirsort**.

The following command sorts the file **tele.lst** into alphabetical order, and displays the result a screenful at a time:

```
C>SORT <tele.lst | MORE
```

9

Grouping commands in batch files

USUALLY when you work with DR DOS you type in commands at the command line and DR DOS carries them out. This is *interactive* processing. There is another kind of processing you can use called *batch* processing.

With batch processing you collect a series of commands together into a batch file. After you type the name of the batch file at the command line, each command in the file is executed in turn, just as if you had typed the commands one at a time. Batch files can have any eight character name, but they must always have the extension **.bat**.

9.1 Using batch files

As you work regularly with DR DOS you will probably find yourself typing the same command or sequence of commands over and over. For example, you may have a set procedure each morning to load your spreadsheet program, and another procedure in the evening to backup all the files you created or altered that day, or you might regularly print a group of files.

In batch processing you can group all the commands you need to carry out these tasks into one file to save yourself the bother of typing them all in. For example, if you used the following commands to tidy your disk and backup your files at the end of each day:

```
C>CD \letters

C>ERASE *.old

C>MD  a:\dayswork

C>COPY *.doc a:\dayswork
```

you could create a batch file to contain all these commands called **dayover.bat.** Then at the end of each day, all you would need to type is:

```
C>dayover
```

In this example, DR DOS first carries out the command `CD \letters` (change to the subdirectory **letters**). It then returns to the batch file and carries out the command `ERASE *.old` (delete all files with the extension ***.old**). Next it processes the command `MD a:\dayswork` (make the subdirectory **dayswork** on drive A:). Finally it returns to the batch file to carry out the command `COPY *.doc a:\dayswork` (copy all files with the extension **.doc** to the subdirectory **dayswork** on a diskette in drive A:).

Note that each new command in the batch file must start on a new line.

Batch files are a flexible and powerful feature of DR DOS. You can create very simple to very complex batch files. They are useful for shortening commands, and for ensuring a procedure is carried out the exact same way every time.

9.2 Creating batch files

You can create batch files with EDITOR, a word processor or with the COPY command (see Chapter 10, "DR DOS commands"). Below is an example of how to use EDITOR to create a simple batch file. (If you create the file with your favorite word processor, make sure the finished file is in plain text, that is ASCII format; refer to your word processor manual for details.)

To create the example file **dayover.bat**:

1. From the command prompt type:

   ```
   EDITOR dayover.bat
   ```

   as this is a new file you see a message similar to this:

   ```
      C:\dayover.bat ....file not found
      Create new file (Y/N)?
   ```

   Answer Y for Yes and the file is created; the EDITOR screen is displayed.

2. Type the commands you want to enter in the batch file, pressing **Enter** at the end of each line:

   ```
   CD\letters
   ERASE *.old
   MD  a:\dayswork
   COPY *.doc a:\dayswork
   ```

3. Press **Ctrl+KX** to save the file and quit EDITOR.

A similar procedure can be used to create any batch file.

9.3 Running batch files

Any batch file that you want to run must be located in either the current directory or one of the directories in the search path set up by the PATH command. (The PATH command is described in Chapter 10, "DR DOS commands").

To run the batch file simply type the batch file's filename. You do not need to type the **.bat** extension. So to run the file **dayover.bat**, all you need type is:

```
C>dayover
```

The system prompt returns when the batch file is finished.

Stopping a batch file

To stop a batch file while it is running, press **Ctrl+C** and you will see:

```
    Halt Batch Process (Y/N)?
```

Type Y for Yes and the remaining commands in the batch file are *not* carried out; you are returned to the system prompt. Type N for No and the remaining commands are processed.

9.4 Making general purpose batch files

DR DOS allows you to make general purpose batch files, that is, batch files that can be made to do different things by adding parameters when it is run.

To allow this kind of thing to happen you need to use replacement variables in your **.bat** file. Replacement variables take the form *%n*, where *n* is a number from zero through nine.

When the batch file is run, the variables it contains are replaced by the different strings found on the command line.

The variable %0 represents the name of the batch file and its drive specification, if included. %1 represents the first parameter on the command line, %2 stands for any second parameter, %3 the third and so on. Use the SHIFT subcommand to alter this correlation.

For example, suppose you often move files from the subdirectory **\amanda\letters** to another subdirectory called **\queries\donewith**. You need a general-purpose batch file that will copy *any* file from the first subdirectory to the second, and then delete the original file. If the batch file is called **moveover.bat,** all you will want to type is:

```
C>MOVEOVER filename
```

where *filename* is the name of *any* file that you want to move.

So, you could create a batch file called **moveover.bat**, in the **c:\amanda\letters** directory, containing the following lines:

```
COPY %1 c:\queries\donewith
ERASE %1
```

Then, if you typed the batch file command:

```
C>MOVEOVER jones.let
```

the commands actually carried out by DR DOS would be:

```
C>COPY jones.let c:\queries\donewith
```

followed by:

```
C>ERASE jones.let
```

If you type the batch file command:

```
C>MOVEOVER smith.let
```

the commands actually carried out by DR DOS would be:

```
C>COPY smith.let c:\queries\donewith
```

followed by:

```
C>ERASE smith.let
```

Remember these points when including variables in your general purpose batch files:

- ❑ You can specify up to 10 variables (%0 through %9) at any one time. If you need to use more than 10 variables, see the SHIFT command.
- ❑ Always separate variable values by a space, and make sure that values are entered in the proper sequence.
- ❑ If you want to use % as a filename within a batch file, you must include it twice. For example, if the filename is **test%.dat,** the batch file must specify **test%%.dat.** It is generally not good practice to use the percentage sign in filenames.

Environment variables

Batch files can also retrieve and use environment variables such as “OS” (defining the operating system), “VER” (the current version number), and “path” (the current path). See also the SET command in Chapter 10, "DR DOS commands".

The environment variable names must be surrounded by % signs when used in batch files.

For example, including the following command in the **autoexec.bat** file:

```
PROMPT %OS% $P$G
```

changes your default prompt to:

```
DRDOS C:\>
```

9.5 Batch file subcommands

Batch files can be made to be much more than a simple list of DR DOS commands and variables. Using batch file subcommands you can create batch files that will:

❑ Call a second batch file from within the first one.

❑ Display (or not display) the batch file commands on the screen.

❑ Cause the batch file to pause while you change diskettes.

❑ Display a message on the screen.

Other batch file subcommands enable you to write batch files that are like simple programs. Such batch files can take decisions and process files in different ways depending on the type of file.

Note

Batch processing subcommands are all internal commands.

Table 9-1 contains an alphabetical list of all the batch processing subcommands.

Table 9-1 Batch Processing Subcommands

Command	Explanation
@ command	Prevents a command from being displayed.
:label	Defines a label that can be referenced by a GOSUB, GOTO or SWITCH batch subcommand.
CALL	Calls a batch file from within a batch file.
ECHO	Allows or inhibits the display of command lines or displays messages.
FOR	Allows repetitive execution of a command for each file specified.
GOSUB	Transfers control to the line following a label and then returns (at a RETURN command).
GOTO	Transfers control to the line following a label.
IF	Provides conditional execution of commands.
PAUSE	Stops batch file processing until the user presses any key.
REM	Allows comments to be inserted into a batch file.
RETURN	Returns control to the line following a GOSUB or SWITCH command.
SHIFT	Makes it possible to change the position of command line parameters to access more than 10 (%0 through %9) replacement variables.
SWITCH	Allows switching between different routines in the same batch file.

Note

These commands are laid out on the page in a similar way to the commands in Chapter 10, "DR DOS commands". Please refer there for an explanation of page layout if one is required.

Command conventions

The following conventions are used in the command descriptions:

`[]` Square brackets indicate an optional part of the command line. Do not type the square brackets themselves, for example for [/C] you type /C.

`|` A vertical bar indicates an either/or choice, for example ON | OFF. Choose just one of the options, and do not type the vertical bar.

`command` Indicates DR DOS command.

`filespec` Represents a file specification which can include some or all of the following:

```
[d:][dirpath]filename[.ext][;password]
```

where

d: is the drive (optional), for example C:

dirpath is the location of the file in the directory structure, for example **\work\** or **\\.**

filename is the 1-8 character name of the file.

[.ext] is the optional 1-3 character file extension, for example **.bat.**

[;password] is the optional 1-8 character password.

`n` Represents a number you type in. The values you can substitute are given in the command explanation.

@command

Format

```
@command
```

Explanation

If you place the @ symbol in front of a command, it will prevent it from being displayed when a batch file is executed. For example if you place @ in front of the command ECHO OFF, ECHO OFF will not be displayed.

:label

Format

```
:label
```

Explanation

:label defines a label that can be referenced by a GOSUB, GOTO or SWITCH batch subcommand. Only the first eight characters after the colon are considered part of the label, additional characters are ignored. Labels are not displayed during batch file execution.

CALL

Format

```
CALL filespec
```

Explanation

CALL allows a batch file to load and run a second batch file, and then returns to execute the remaining commands in the original batch file; the PATH is searched when the second batch file is looked for.

CALL subcommands can be placed at any line in the batch file.

Example

The following example calls the batch programs **second.bat** and **third.bat** and processes them in turn:

```
CALL second

CALL third
```

ECHO

Format

```
ECHO [ON|OFF|message]
```

Explanation

ECHO allows or inhibits the screen display of DOS commands executed from a batch file. It does not affect the messages produced during command execution.

ECHO is ON by default, and shows all commands as they run. ECHO OFF stops all commands from appearing on the screen.

The ECHO `message` option displays messages on the screen whether or not ECHO is ON or OFF. You can display several lines of text in this way:

```
ECHO This is my very

ECHO own batch file
```

The current status of ECHO is displayed if ECHO is issued from the command line with no parameters.

Note

Preceding a command with the @ character prevents the command from being displayed. For example, if a batch file included the command @ECHO OFF, the words `echo off` would not be displayed on the screen when the batch file is run.

@ECHO OFF is often used at the start of a batch file to prevent all commands from being displayed.

FOR

Format

```
FOR %%variable IN (fileset) DO command
```

Explanation

FOR allows for repetitive execution of a command. Each filename in the set is sequentially substituted in the command following DO. Wildcards are valid for members of the set and all matching filenames on the disk are substituted in the command. Only one FOR command can be used on a command line.

Example

If you enter this command in a batch file:

```
FOR %%f IN (myprog.asm acct.bas acct2.bas)DO type %%f
```

the three specified files are each typed to the screen in turn. If you enter a command like this at the command line, you only need enter one set of % signs, for example:

```
FOR %f IN (myprog.asm acct.bas acct2.bas)DO type %f
```

GOSUB

Format

```
GOSUB label
```

Explanation

The GOSUB subcommand means that processing of the batch file will continue at the line following the specified label and then return to the line following the GOSUB command when a RETURN command is encountered.

For more information about specifying labels, read the description of the GOTO subcommand.

You must specify a RETURN command with GOSUB, so that processing can return after the subroutine identified by the label is executed.

Example

The following lines in a batch file would display the file **intro.txt** and then execute a subroutine to display the files **lista.txt**, **listb.txt**, and **listc.txt** before calling the batch file **summary.bat**:

```
TYPE intro.txt

GOSUB lists

CALL summary

EXIT

:lists

TYPE lista.txt

TYPE listb.txt

TYPE listc.txt

RETURN
```

GOTO

Format

```
GOTO label
```

Explanation

The GOTO subcommand means processing of the batch file will continue from the line following on from the label specified. If the label is not found, the current batch file ends with this message:

```
     Label ... not found
```

To set up a label in a batch file, use a colon (:) followed by a character string. The first eight characters of the string are considered the significant characters, and they must be different from those of all other labels.

Labels within a batch file are not shown while the file is executing. Labels not referenced by a GOTO subcommand may be used to place comments in your batch file.

Example

The following batch file displays the file **info.txt** continuously, until you press **Ctrl+C** or **Ctrl+Break**:

```
:start
TYPE info.txt
GOTO start
```

IF

Format

```
IF [NOT] condition command
```

Explanation

IF allows conditional processing of DOS commands. The DR DOS statement is executed when the IF condition is true. When the condition is false, the DR DOS command is ignored and the next statement is executed.

condition can be one of the following:

- ❑ **ERRORLEVEL number**

  This condition is true if the previous program had an exit code of greater than or equal to *number*. Your programs can set an error code that can then be checked by the IF ERRORLEVEL command.

- ❑ **string1==string2**

  When *string1* and *string2* are identical, this condition is true.

- ❑ **EXIST filespec**

  If *filespec* is on the indicated drive, EXIST filespec is true. Pathnames are allowed with the filespec. Note that an explanation of filespec can be found in Chapter 10, "DR DOS commands", section 10.1.

- ❑ **DIREXIST dirpath**

  If *dirpath* is on the indicated drive, DIREXIST dirpath is true.

If you include the NOT parameter, DR DOS will carry out the command when the condition is *false*.

Note

Many DR DOS commands set the ERROR LEVEL which can be tested with an IF ERRORLEVEL command; see Chapter 10, "DR DOS commands" for details.

Examples

IF EXIST filespec command:

```
IF EXIST myfile GOTO xyz
DIR B:
.
.
:xyz
command
```

This condition is true if executed with **myfile** on the default drive. The *command* following the label **xyz** is executed. If **myfile** is not found, the `GOTO xyz` is not executed and processing continues with the `DIR B:` command.

IF string1==string2 command:

```
IF "%1"=="John" ECHO What are you doing here?
```

The batch file with this command, assuming **John** is named as the %1 parameter, performs the ECHO batch command. `What are you doing here?` is displayed. If **John** is not given as the %1 parameter, the condition is false, and the ECHO batch command is not executed.

Note

You should always place quotation marks around the text string, as this prevents syntax errors if no parameters are specified.

IF ERRORLEVEL number command:

```
XCOPY %1 c:\queries\donewith
IF ERRORLEVEL 1 GOTO nodelete
DEL %1
:nodelete
```

The above example further enhances the batch file **moveover.bat** (described in section 9.4, "Making general purpose batch files") to only delete a file if the file is copied first. The XCOPY command is used to copy the file. When it completes, XCOPY sets the errorlevel to **0** if it finishes successfully, and to **1** if it fails. If it fails, the condition is true and the line `IF ERRORLEVEL 1 GOTO nodelete` is processed. This ensures the file is not deleted unless it has been copied first.

IF NOT EXIST filespec command:

```
IF NOT EXIST a:%1 COPY b:%1 a:
myprog1
```

This batch file is looking for a certain file to be on drive A:. The IF condition is executed before **myprog1** to ensure that the required file is on drive A:. If the file is not on drive A:, the condition is true and the copy command is processed, copying the file from drive B: to drive A:. If the file is on drive A:, the copy is not processed.

PAUSE

Format

```
PAUSE [remark]
```

Explanation

You can insert a PAUSE subcommand into a batch file to delay the processing of the file. The pause in processing allows you to let an action occur, such as changing disks.

PAUSE always shows the following message:

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

The optional remark can contain a maximum of 122 characters.

Example

The following example shows a PAUSE subcommand in a batch file:

```
COPY a:*.* b:
PAUSE Insert new disk in drive A
COPY a:*.* b:
```

When the batch file is run, the first diskette in drive A: is copied. Then the processing is suspended and this message appears:

```
    PAUSE Insert new disk in drive A
    Strike a key when ready . . .
```

Two responses are possible:

- ❑ Insert the new diskette and press any key to continue.
- ❑ Press **Ctrl+C** to stop the processing.

REM

Format

```
REM | ; [remark]
```

Explanation

Use REM to insert remarks in a batch file.

The maximum length of a remark is 123 characters. You can use spaces, tabs and commas as separators. You can also use REM to add space to your batch file for readability.

Note

If ECHO is ON, REM comments appear on the screen as the file is processed.

If ECHO is OFF, REM comments are not displayed.

Example

The following is an example of a REM command:

```
REM Archive backup files for security
```

RETURN

Format

```
RETURN
```

Explanation

RETURN causes processing of the batch file to return to the line following a GOSUB or SWITCH command, after executing the commands following the label specified by the GOSUB or SWITCH command.

Refer to the descriptions of GOSUB and SWITCH for more information about how RETURN is used.

SHIFT

Format

```
SHIFT
```

Explanation

SHIFT allows you to change the position of command line parameters to access more than 10 (%0 through %9) replacement variables. Execute the SHIFT command to get past the tenth parameter on the parameter list. The parameters on the command line are shifted one position to the left; the parameter in the %1 position moves to the 0% position. Each shift command moves all the parameters to the left by one more position.

Example

Consider the following batch file called **dirlist.bat**:

```
ECHO off
REM This batch file calls DIR on a number of
REM specified directories or drives
:loop
IF "%1%" == "" GOTO done
DIR /P %1
SHIFT
PAUSE
GOTO loop
:done
ECHO Task complete
```

This example displays directory listings of various specified directories or drives, operating on the %1 position as many times as there are parameters included on the command line. At the command line you could type for example.

```
C>DIRLIST a:\sales b:\sales\foreign\*.doc b:\clients
```

This would produce a listing for each of the specified directories.

SWITCH

Format

```
SWITCH label1, label2[, labeln]
```

Explanation

The SWITCH command allows you to switch between different routines within the same batch file. SWITCH causes the operating system to prompt you, and then jump to the label you specify. Processing returns to the line following the SWITCH command when a RETURN statement is encountered.

The maximum number of labels you can use with a SWITCH command is 9.

9.6 The autoexec.bat file

The **autoexec.bat** file is a special batch file that DR DOS runs each time your computer is started. **Autoexec.bat** sets up your computer the way you want it each time the computer is switched on. For more information see Chapter 11, "Customizing your system", section 11.5.

10

DR DOS commands

THIS chapter contains a complete list and description of the DR DOS commands.

It also provides an explanation of command syntax and how to edit commands at the command line using the DR DOS command line editing, and extended command line editing keys.

Note that batch processing subcommands are explained in Chapter 9, "Grouping commands in batch files", and the **config.sys** file commands in Chapter 11, "Customizing your system".

10.1 Command syntax explained

The syntax of each DR DOS command is a combination of the following items:

`filename` The command's filename. Some command filenames can be abbreviated, as explained in the text of this chapter. You do not need to include the file extension. To use an external command you must have it's corresponding "program file" in your current directory, or in one of the directories pointed to by the path statement.

`dirname` Represents a directory name.

`filespec` Represents a file specification which can include some or all of the following:

```
[d:][dirpath]filename[.ext][;password]
```

where

d: is the drive (optional), for example C:

dirpath is the location of the file in the directory structure, for example **\work** or \.

filename is the 1-8 character name of the file.

.ext is the optional 1-3 character file extension, for example **.bat**.

;password is the optional 1-8 character password.

`wildspec` This is the same as a filespec, but it indicates that you can use the wildcard characters ? and * in filenames and extensions to specify groups of files.

`@` If you see the symbol @ in front of a filespec, it means you can insert a filename containing a list of files which you want the command to have an effect on. You can create a text file (using EDITOR) containing several different filenames, you can even use wildcard characters in the names.

Note that each filename in your text file must be placed on a new line, or be separated by a space, tab or comma. Note also that you cannot use wildcard characters in the filelist name.

`device` Represents a peripheral device, for example prn.

`[ ]` Square brackets mean an optional clause in a command. Type what is shown in the brackets, do not enter the brackets into the command line. For example for [/C], you type /C.

`|` A vertical bar means either/or, for example ON | OFF. Choose one or the other of the separated items. Do not type the vertical bar.

... Three periods means you can repeat an item. If a filespec or wildspec is followed by ..., it means you can have multiple file or wildspecs on one command line. For example, the syntax for the FIND command is:

```
FIND [options] "search string" [@]wildspec...
```

So instead of typing two commands like this:

```
FIND error *.C
```

and

```
FIND error *.H
```

you can put the wildspecs *.C and *.H on the same command line, as follows:

```
FIND error *.C *.H
```

`n` This means enter a number within a specified range. The command explanation gives the range of valid numbers.

`command`	Any parameter or option entered on a command line after the command name.

Note

All punctuation (excluding the special cases listed above) should be entered as shown.

10.2 Editing commands at the command line

DR DOS has powerful command line editing features to save you effort when typing commands. In particular, it can store in memory the commands you have issued recently. This saves you for example having to retype a command line if you make a typing error. By using certain keys you can redisplay the command, edit it and reissue it.

Say you mistype the ERASE command:

```
C>efase myfile.doc
```

The error message:

```
    Command or filename not recognized
```

is displayed. Although you have entered the command by pressing **Enter**, it is not lost, but stored in a special area of memory called a *buffer*. You can use the command line editing keys to correct the command and then re-enter it:

Press ↑	To display the previous command line.
Press ←	To move the cursor using the left arrow key until it is on top of the `f`.
Press **Del**	To delete `f` from the stored line.
Press **Ins**	To turn on “insert mode”.
Type **r**	To enter `r` into the stored line.
Press ↵	To re-enter the correct command.

Similarly the command:

```
C>erase myfle.doc
```

can be corrected by the following keystrokes:

Press ↑	To display the previous command lines.
Press ←	To move the cursor until it is on the `l`.
Press **Ins**	To switch on “insert mode”.
Type **i**	To add `i` to the display.
Press ↵	To re-enter the correct command.

Command line editing keys

The command line editing keys comprise the function keys **F1** to **F6, Del, Ins, Backspace** and **Esc.** Their functions are listed in Table 10-1.

Table 10-1 DR DOS Command Line Editing Keys

Key	Meaning
Esc	Cancels the command line and leaves the original contents of the stored line unchanged.
Backspace	Deletes the previous character in the stored line.
Del	Deletes the current highlighted character in the stored line.
Ins	Allows you to insert characters into the stored line. Any characters you type are displayed on screen. You can switch Insert mode off by pressing **Ins** again.
F1	Copies and displays one character from the stored line.
F2	Followed by a character, copies all characters up to a specified character.
F3	Copies all remaining characters from the stored line to the display.
F4	Followed by a character, deletes all characters up to a specified character in the stored line.
F5	Copies the current line (as edited) to the stored line. The line is not edited as a command and can be edited further if desired.

F6	Specifies the End-Of-File marker (**Ctrl+Z**), for example if you create a file using the COPY command pressing **F6** when you have finished entering the text, gives you the marker which indicates *this is where the file ends*, for example:
	`COPY CON test.bat`
	`DIR a:`
	`DIR b:`
	`F6`

Extended command line editing

DR DOS offers additional editing keys to those described in section 10.2, "Editing commands at the command line". Table 10-2 lists the extended editing keystrokes and their meanings.

Table 10-2 DR DOS Extended Command Line Editing Keys

Keystrokes	Meaning
← or **Ctrl+S**	Move one character to the left.
→ or **Ctrl+D**	Move one character to the right.
↑ or **Ctrl+E**	Move to the previous command in memory.
↓ or **Ctrl+X**	Move to the next command in memory.
Enter or **Ctrl+M**	Enter the command line.
Backspace or **Ctrl+H**	Delete the character to the left of the cursor.
Del or **Ctrl+G**	Delete the character at the cursor.
Ins or **Ctrl+V**	Switch insert/overstrike mode on or off.
Ctrl+A	Move one word left.
Ctrl+F	Move one word right.
Home or **Ctrl+Q**	Move to the beginning of the line.
End or **Ctrl+W**	Move to the end of the line.
Ctrl+T	Delete a word.
Ctrl+Y	Delete a line.
Ctrl+B	Delete to the beginning of the line.
Ctrl+K	Delete to the end of the line.
Ctrl+R	Start Command line search, see *.
Ctrl+_	Turns Command line search mode on or off. The default is off, see *.

*Searching for commands

Ctrl+R displays the last matching command. You can then enter the displayed command again, or edit it before pressing **Enter**.

Ctrl+_ switches on a more sophisticated version of command line search. When it is active, each time a character is typed, DR DOS displays the last command which matches the first character typed on the command line. For example, if you type d then the last command you entered that started with the letter d is displayed. If you type c, then the last command you entered that began with the letter c is displayed.

Command line search remains on until you type **Ctrl+_**again.

Note

Some of these commands are only available if you have included the command line HISTORY= in your **config.sys** file. This may have been set up when DR DOS was installed, but if you find it was not, you can use SETUP to reconfigure it; see Chapter 11, "Customizing your system".

DR DOS control key combinations

There are certain control key combinations within DR DOS that can be used to carry out particular tasks. They are listed below:

Ctrl+C or **Ctrl+Break**	Stops commands when they are running. The processing of the command is stopped, and the command prompt returned.
Ctrl+P	Switches between echoing text to the screen, or to a printer.
Ctrl+S or **Pause**	Suspends scrolling. To start scrolling again, press any key.

10.3 Command summary

This section contains a table listing all the DR DOS commands with a brief summary of what they do. The table shows whether the commands are internal or external (see Chapter 2, "Basic command concepts" for more on internal and external commands). You can display help text on any of the DR DOS commands listed in Table 10.3 by typing the command followed by /? or /H (for example APPEND /?). For more detailed online help, you can also use the DOSBook command, explained in this chapter.

Table 10.3 DR DOS Command Set

Command	Explanation	Status
APPEND	Specifies a search path for data files and overlays.	External
ASSIGN	Assigns a different drive letter to a disk drive or subdirectory.	External
ATTRIB	Displays and modifies file attributes.	External
BACKUP	Makes backup copies of files, directories, and disks and diskettes, for security.	External
BREAK	Allows you to stop a program running under DR DOS by pressing **Ctrl+Break**.	Internal
CHCP	Changes (selects) a specified code page.	Internal
CHDIR(CD)	Changes the current directory on a drive and displays the current directory path.	Internal
CHKDSK	Checks the integrity of data on a disk and (optionally) makes corrections.	External
CLS	Clears the display screen.	Internal
COMMAND	Loads a second copy of the DR DOS default command processor (**command.com**).	External
COMP	Compares two files character by character and reports mismatches.	External
COPY	Copies files to other locations or combines multiple files into one file.	Internal
CTTY	Redirects input and output to a different device.	Internal
CURSOR	Modifies the cursor and its flash frequency.	External

Command	Explanation	Status
DATE	Displays and sets the current date.	Internal
DEL	Removes files from directories.	Internal
DELPURGE	Frees space occupied by *pending delete* files.	External
DELQ	Removes files but queries you first.	Internal
DELWATCH	Saves deleted files on disk as pending delete files.	External
DIR	Lists the contents of a directory.	Internal
DISKCOMP	Compares entire diskettes (of the same format) track by track.	External
DISKCOPY	Copies entire diskettes (of the same format).	External
DISKMAP	Saves a copy of the File Allocation Table on a disk, to assist file recovery.	External
DISKOPT	Optimizes disk performance by rearranging files on the disk. DISKOPT is described in more detail in Chapter 13, "Optimizing your disk".	External
DOSBOOK	Starts the DR DOS on-line documentation.	External
EDITOR	Allows you to edit (display, write, and change) text files. The EDITOR is described in Chapter 7, "The EDITOR".	External
ERAQ	Erases files but queries you first.	Internal
ERASE(ERA)	Removes files from directories.	Internal
EXE2BIN	Converts executable files (**.exe**) to binary image (**.bin**) or **.com** files.	External
EXIT	Returns you to another running application.	Internal
FASTOPEN	Reduces the amount of time taken to access frequently-accessed files on a hard disk.	External
FC	Compares ASCII text or binary files and reports the differences found.	External
FDISK	Formats partitions on a hard disk and provides information about partitions.	External
FILELINK	Transfers files between two computers connected by their serial communications ports.	External
FIND	Finds a string of characters in a group of text files.	External
FORMAT	Formats a disk or diskette so that it can be used.	External

Command	Explanation	Status
GRAFTABL	Allows you to display extra and international characters with a Color/Graphics Adaptor (CGA).	External
GRAPHICS	Allows you to print a graphics display using the **PrtScr** key.	External
HILOAD	Loads specified programs into upper memory when you start DR DOS.	Internal
JOIN	Attaches the directory structure on a disk drive to an empty subdirectory on another disk drive.	External
KEYB	Loads the appropriate keyboard program for non-US keyboards.	External
LABEL	Creates, modifies, or deletes the volume label on a disk or diskette.	External
LOCK	Temporarily locks your operating system for security.	External
MEM	Displays information about the way memory is currently being used on your computer.	External
MEMMAX	Selectively disables DR DOS enhanced memory features that conflict with certain applications.	External
MKDIR(MD)	Creates a subdirectory in the root directory or another subdirectory.	Internal
MODE	Changes the configuration of devices attached to your system (such as monitors, printers, and code pages).	External
MORE	Causes DR DOS applications' output to the console to be displayed a screenful at a time.	Internal
MOVE	Moves files and subdirectories to other locations.	External
NLSFUNC	Provides support for extended country information and allows the CHCP command for selecting code pages.	External
PASSWORD	Assigns passwords to files and paths, for security.	External
PATH	Sets a search path for commands or batch files.	Internal
PRINT	Prints text files.	External
PROMPT	Modifies the command prompt.	Internal
RECOVER	Retrieves corrupted file information, for individual files or an entire disk.	External

Command	Explanation	Status
RENAME (REN)	Renames files or moves files between subdirectories on the same disk.	Internal
RENDIR	Renames directories.	External
REPLACE	Copies selected files.	External
RESTORE	Restores files and directories that were backed up using the BACKUP command.	External
RMDIR(RD)	Removes specified subdirectories.	Internal
SCRIPT	Provides PostScript support.	External
SET	Inserts variable strings into the command processor's environment.	Internal
SETUP	Changes the system configuration values. The SETUP program is described in Chapter 11, "Customizing your system".	External
SHARE	Supports file sharing.	External
SID	Allows interactive testing and debugging of programs. SID is described in Appendix D, "The SID debugging tool".	External
SORT	Reads standard input data, sorts it alphabetically, and then writes it to the standard output device.	External
SSTOR	Starts the SuperStor data compression program. SuperStor is described in more detail in Chapter 13, Optimizing your disk".	External
SUBST	Replaces a path specification with a drive letter.	External
SUPERPCK	Starts the Super PC-Kwik disk cache program. Super PC-Kwik is described in more detail in Chapter 13, "Optimizing your disk".	External
SYS	Transfers the DR DOS system files and **command.com** from the source drive to a specified drive.	External
TASKMAX	Allows you to switch easily between different applications. The TASKMAX utility is described in Chapter 6, "Switching tasks".	External
TIME	Displays or sets the time of day.	Internal
TOUCH	Resets the time and date stamps for files.	External

Command	Explanation	Status
TREE	Displays the hierarchical path of directories and subdirectories on a disk.	External
TYPE	Displays the contents of a text file on the screen.	Internal
UNDELETE	Attempts to recover deleted files.	External
UNFORMAT	Unformats a diskette that has been "safe formatted" using the DR DOS FORMAT command.	External
UNINSTAL	Restores your old operating system.	External
VER	Displays the version of DR DOS you are currently using.	Internal
VERIFY	Checks that data is correctly written to disk after every write operation.	Internal
VOL	Displays the volume label of a specified drive.	Internal
XCOPY	Selectively copies groups of files.	External
XDEL	Selectively deletes groups of files and subdirectories.	External
XDIR	Lists the contents of a directory, with extra options such as including files in subdirectories.	External

10.4 The DR DOS command set

The commands in the DR DOS command set are described like this:

Command name

Format

```
The command syntax in full
```

Explanation

What the command is and does.

Command switches or options
A description of what the command switches (if any) do.

Remarks

Any relevant notes or tips on using the command.

Examples

Where appropriate, examples of use are given.

APPEND

Format

```
APPEND [/?|/H] d:dirpath [;[d:]dirpath...]
[/X:ON|OFF] [/E:ON|OFF] [/PATH:ON|OFF]
```

Explanation

APPEND enables programs to look in other specified directories for files that it cannot find in your current directory. It is similar to the PATH command. PATH is used to search for **.com**, **.exe** and **.bat** files. APPEND is used when programs want to search for any other type of file.

The APPEND command is generally used with older-style DOS applications which do not support pathnames. Normally you should not need this command, unless your applications documentation advises you to use it.

The two main forms of the command are:

- ❑ APPEND followed by the /X switch.
- ❑ APPEND followed by pathnames.

Before you enter an APPEND command to specify a search path, you can issue APPEND with the /X:ON switch. This causes APPEND to work with programs that use the Search First, Find First, and Execute functions. To switch it off again, type APPEND /X:OFF.

Caution Use this option with care; it can lead to problems when used with certain application programs. Some applications cannot use the APPEND command; you will find that they read files successfully from directories other than the current directory, but they can only create files in the current directory. This results in two copies of files: an edited version on the current directory and the unedited version left unchanged on the original directory.

Command switches

/? or /H	Displays help text.
/X:ON	Append ON for search and exec functions.
/X:OFF	Append OFF for search and exec functions (the default).
/E:ON	The append path to be kept in the DOS environment.
/E:OFF	The append path not to be kept in the DOS environment (the default).
/PATH:ON	Append will apply to files having pathnames (the default).
/PATH:OFF	Append will not apply to files having pathnames.

APPEND with pathnames

APPEND followed by pathnames is used to specify a search path. Once entered, APPEND specifies where DR DOS looks for files if it cannot find them in the current directory. You can specify several search directories provided each is separated by a semi-colon (;). The total length of the APPEND command cannot exceed 128 characters.

To return to the default value of no search path, type:

```
A>APPEND ;
```

APPEND and BACKUP

Before you use the BACKUP command you should cancel the APPEND search string by entering:

```
A>APPEND ;
```

Examples

The following command sets up a search path such that if a program cannot find the file you want in your current directory, it looks in the directory **c:\spreads\cashflow**.

```
A>APPEND c:\spreads\cashflow
```

The following example extends the search path so that the program also looks in the directory **\oldones** on the A: drive.

```
A>APPEND c:\spreads\cashflow;a:\oldones
```

ASSIGN

Format

```
ASSIGN [/?|/H] [old_drive_letter=new_drive_letter]...
[/A]
```

Explanation

ASSIGN causes commands that would access a certain drive to access a different drive. It is mainly used when you have a program set up to access a certain drive like drive A: but the information to access is stored on a different drive (for example drive C:). You can use ASSIGN rather than re-installing or modifying your program.

To cancel an ASSIGN statement, simply type ASSIGN on its own.

Command switches	
/? or /H	Displays help text.
/A	Displays the current status and assignment.

Remarks

Do not use the following DR DOS commands when an ASSIGN command is in effect:

BACKUP
DISKCOMP
DISKCOPY
FORMAT
LABEL
PRINT
RESTORE
SUBST

Examples

The following ASSIGN command reassigns drive A: as drive C:

```
C>ASSIGN a=c
```

So for example, if you type:

```
C>DIR a:
```

you will now see a DIR listing of the C: drive. Typing ASSIGN /A will give you the current assignment made, in this case:

```
    A:=>C:
```

If you type :

```
C>DIR c:
```

you will also see a DIR listing of the C: drive.

Caution The above example means that any command issued that includes the drive letter A is read as if it was the letter C. This means drive A: is inaccessible to you. In general, do not use ASSIGN unless you are running a specific program that requires it.

This command cancels any ASSIGN statements made:

```
C>ASSIGN
```

ATTRIB

Format

```
ATTRIB [/?|H][+|-A][+|-H][+|-R][+|-S][@]wildspec...
[/P][/S]
```

Explanation

ATTRIB displays a file's attributes, and can be used to switch these attributes on or off. The attributes included are the archive, hidden, read-only and system attributes.

If you do not specify any filenames on the command line, ATTRIB command assumes *.* and acts on all files in the current directory.

A attribute

The archive attribute (A) shows that a file has been modified. The ATTRIB command enables you manually to switch the archive bit on or off for any given file. This is useful when you use the BACKUP switch /M which copies all files with the archive bit set.

S attribute

The system attribute (S), when switched on, prevents DIR from displaying the file, COPY from copying it, and MOVE from moving it unless a switch is specified. It is used for the DR DOS system files that you do not need to see in everyday work.

Note

COPY, XCOPY and MOVE have switches that allow you to copy system files.

R attribute

A file that is set to be read-only (R) cannot be erased, renamed, or altered in any of the normal ways. Setting a file with the read-only attribute should protect it from being inadvertently altered. However, the MOVE, REPLACE, TOUCH and XCOPY commands all have /R switches that allow you to alter read-only files.

H attribute

The hidden attribute (H), when switched on, hides files from view in standard directory listings.

Command switches

/? or /H	Displays help text.
+A	Sets the archive attribute (-A switches it off).
+R	Sets the read-only attribute (-R switches it off).
+H	Sets the hidden system file attribute (-H switches it off).
+S	Sets the system attribute (-S switches it off).
/P	Pauses after each screenful in a long display.
/S	Sets the file attributes in the subdirectories associated with the chosen directory.

Examples

This example switches on the archive attribute for all the files with the extension **.new** in the current directory. You can then use a command like XCOPY /M to copy all the files marked for archiving.

```
A>ATTRIB +A *.new
```

This command switches off the read-only attribute for all the files with the **.bas** extension in the current directory, and any associated subdirectories.

```
A>ATTRIB -R *.bas /S
```

BACKUP

Format

```
BACKUP [/?|/H] source_drive [filespec]
destination_drive [/S] [/F[:nnn]] [/M] [/D:date]
[/T:time] [/A] [/L:filespec]
```

Explanation

BACKUP makes backup copies of your files.

You can backup whole hard disks and diskettes, or directories, groups of files, or even single files. The backed up files must be restored using the RESTORE command before they can be used as DR DOS files.

DR DOS BACKUP and RESTORE programs are compatible with DOS BACKUP and RESTORE programs, version 3.30 or later. Files backed up with a DOS BACKUP can be restored with DR DOS RESTORE and vice versa.

Caution Password-protected files and subdirectories are NOT backed up, unless the global password is set before running BACKUP, and the files have been protected with the same password.

Command switches

/? or /H — Displays help text.

/S — Backs up the current directory and all the subdirectories associated with it.

You must be in the root directory and specify /S if you want to backup an entire disk. The default is to backup only the *current* directory.

/F[:*nnn*] — Formats the destination diskette before copying the files. You can use *nnn* to specify the format required, where *nnn* is one of the following values:

Command switches (continued)

For 5¼" diskettes	For 3½" diskettes
160 (160 Kbytes)	720 (20 Kbytes)
180 (180 Kbytes)	1.44 (1.44 Mbytes)
320 (320 Kbytes)	2.88 (2.88 Mbytes)
360 (360 Kbytes)	
1.20 (1.2 Mbytes)	

/M — Backs up all the files on the disk modified since the last backup. (This switch actually copies all the files for which the archive bit is set. You can set the archive bit manually by using the ATTRIB command.)

/D:*date* — Backs up only the files that have been modified since the specified date. The date format you use is governed by the COUNTRY command in your **config.sys** file, but is usually the one commonly used in your country: month/day/year in the US, day/month/year in most European countries, and year/month/day in Japan.

/T:*time* — Backs up only the files that have been modified since the specified time. The format to use is hh:mm:ss.

/A — Adds new files to the backup disk without erasing existing backup files (the default is to overwrite all the existing information on the backup disk).

/L:*filespec* — Creates a file recording the date and time of backup and number of the backup diskette on which each file is stored. The default filename for this backup log is **backup.log**, though you can specify another filename. DR DOS places the log in the root directory of the source disk. If you specify a name for the log file that already exists on the root directory, DR DOS appends the new log information at the end of the existing log file.

BACKUP reports any errors using the codes shown in Table 10-4.

Table 10-4 BACKUP Error Codes

Code	Explanation
0	Normal completion.
1	No files found to backup.
2	Some files not backed up due to file sharing conflict.
3	Backup ended by **Ctrl+Break** command.
4	Backup ended by an error.

This feature can be useful in creating batch files with an IF ERRORLEVEL statement in them to improve the process, see Chapter 9, "Grouping commands in batch files".

Remarks

Do not use the BACKUP program with disks affected by an ASSIGN, JOIN or SUBST command.

When you make a backup, label and number all your diskettes consecutively to make it easier when you come to restore them using the RESTORE command.

If you do not want files overwritten on your backup diskettes, specify the /A switch in the command line.

For information on taking backups, see Chapter 4, "Protecting your information".

Examples

The following command backs up the entire hard disk onto diskettes. When each diskette is full, you are prompted to remove it and insert the next one

```
C>BACKUP c:\ a: /S
```

Note

To backup a full 20 Mbyte hard disk you may need up to 60 diskettes, though this number varies with the capacity of your diskette drive.

The following command backs up all the **.doc** files that have been modified since the last BACKUP on directory **\ww** and on the subdirectories associated with it.

```
C>BACKUP c:\ww\*.doc a: /S /M
```

BREAK

Format

```
BREAK [/?|/H][ON|OFF]
```

Explanation

BREAK, when switched ON, allows you to stop a running program by pressing **Ctrl+Break.** The program will then stop the next time it tries to call any DOS function, such as accessing the disk.

If BREAK is OFF (the default value), you can only issue a **Ctrl+Break** command when the program you want to stop is performing standard input or output operations such as printing or outputting to the console.

If you type BREAK on its own, you see the current BREAK value.

Command switches	
/? or /H	Displays help text.

Examples

The following command switches BREAK ON. BREAK remains ON until you issue another BREAK command or restart DR DOS:

A>**BREAK ON**

The following command shows the current status of the BREAK command:

A>**BREAK**

CHCP

Format

```
CHCP [/?|/H] [cp]
```

Explanation

Use CHCP to change the active code page for DR DOS and all devices that support code page switching by selecting another code page.

The cp parameter specifies the code page number you want to select. The only other parameter you can use with CHCP is /? or /H, to display help information about CHCP.

Entering CHCP with no parameters displays a message telling you which code page is currently active.

Command switches	
/? or /H	Displays help text.

Remarks

Make sure you do the following before using CHCP:

- ❑ Load NLSFUNC. Refer to the NLSFUNC explanation in this chapter.
- ❑ Prepare the code page you want to select using the MODE PREPARE command. Refer to the MODE command explanation in this chapter.
- ❑ Include the COUNTRY statement in your **config.sys** file, or use the INSTALL or SETUP program to include it automatically. Refer to the COUNTRY statement explanation in Chapter 11, "Customizing your system" for more information.
- ❑ Include DEVICE statements in your **config.sys** file to load the DISPLAY.SYS and PRINTER.SYS device drivers, as appropriate, or use the SETUP program to include them automatically. Refer to the DISPLAY.SYS and PRINTER.SYS explanations in Chapter 11, "Customizing your system" for more information.

- ❑ Switch on all printers that support code page switching and make sure they are online.

Refer to Chapter 14, "Setting up code page switching" for more information about code page switching.

Example

The following command changes the code page for all devices that support code page switching to Code Page 850:

```
C>CHCP 850
```

CHDIR (CD)

Format

```
CHDIR [/?|/H] [d:]|[[d:]dirpath]
```

Explanation

The CHDIR (change directory) command changes the current directory on a drive and displays the current directory path.

When no parameter is specified, CHDIR displays the directory path leading to the current directory. When only a drive is specified, the current directory of that drive is displayed. Note that a path consisting of just a drive and a backslash (\) indicates that the root directory is the current directory.

If a directory is specified, CHDIR makes the last directory in the path the current directory, for the current drive or the drive specified.

If the directory path specified is on the current drive, you are immediately moved to the new directory. If, on the other hand, the directory path relates to a drive other than the one you are in, you remain where you are. However, if you now move to the drive you have specified, you will find yourself in the new current directory.

If the directory you are currently in is not part of the path to the directory you want, you must specify the complete path from the root, preceding the first directory with a backslash.

Command Switches	
/? or /H	Displays help text.

Special characters

To help you move between directories, CHDIR responds to two special characters \ (backslash) and .. (two periods).

Enter a backslash alone with CHDIR and you are returned to the current or designated drive's root directory. For example, the command:

```
A>CHDIR C:\
```

makes the root directory on drive C: the current directory.

Enter two periods (..) with CHDIR and you are returned to the next directory up in the direction of the root.

For example, if the directory **bonus** in the path **accounts\wages\bonus** is the current directory, the command:

```
A>CHDIR ..
```

makes **wages** the current directory.

Examples

The following command makes **personal** the current directory. The backslash that precedes **clients** indicates that the path to **personal** begins from the root of drive C:, not from the current directory.

```
C>CHDIR  \clients\personal
```

The following command makes the drive C: root directory the current directory.

```
C>CHDIR  \
```

The following command makes **memos** the current directory on drive A:. All references to A: now access this directory.

```
C>CHDIR  a:\wordproc\memos
```

CHKDSK

Format

```
CHKDSK [/?|/H] [d:][wildspec] [/F] [/V]
```

Explanation

To check your disks regularly for errors, and to keep track of the amount of space you have used, you can use the CHKDSK command. You can use CHKDSK to check a hard disk or a diskette, as long as the diskette is in the drive before you enter the command.

Note

CHKDSK also works on SuperStor compressed drives. Refer to Chapter 13, "Optimizing your disk" for information about creating SuperStor disks.

If you do not specify a disk drive with CHKDSK on the command line, your current drive is assumed.

CHKDSK does the following:

- Scans the disk or diskette for space allocation and spacing errors
- Produces a report indicating the amount of space left (in bytes), the amount of space used (in bytes) and the number of existing files (including hidden and *pending delete* files), the amount of memory available, and errors found.

A typical CHKDSK status report would appear as follows:

```
Volume MYDISK created 28 June 1992 13:44

 33,462,272 bytes total disk space
    135,168 bytes in 9 hidden files
     81,920 bytes in 31 directories
 21,278,720 bytes in 1122 user files
    446,464 bytes in 22 pending delete user files
     10,240 bytes in bad sectors
 11,509,760 bytes available on disk
```

```
655,360 bytes total memory
581,904 bytes available
561,704 bytes in largest free memory block
```

Note

If you specify CHKDSK with a file specification, CHKDSK checks that the specified files are allocated in contiguous clusters on the disk, and displays any files that consist of discontiguous clusters.

Correcting errors

CHKDSK will correct errors it finds if you specify the /F switch. If you do not specify the /F switch, CHKDSK will ask you whether you want corrections written to the disk. Note that correcting errors can involve altering the disk's File Allocation Table (FAT), a kind of index that DR DOS uses to locate files and allocate space for new files. Altering the FAT can mean loss of data.

When CHKDSK analyzes a disk, it looks for file extracts that are "lost" or "mixed up" such that DR DOS can no longer determine which file they come from. If CHKDSK finds any of these lost clusters, you see a message like the following one:

```
Five lost clusters found in 2 chains.
Convert lost chains to filennnn.chk files (Y/N)?
```

Answer Y for Yes or N for No.

- If you answer Y, CHKDSK collects the lost clusters into files named **filennnn.chk** (where **nnnn** is a number between 0000 and 9999) in the root directory. As long as these files are not in binary format, you may be able to look at them using the EDITOR or the TYPE command, recover any valuable data they contain, and then delete them.
- If you answer N, CHKDSK frees the lost clusters so that they can be used when new files are created.

Command switches

/? or /H	Displays help text.
/F	Allows CHKDSK to write corrections back to the disk.
/V	Displays filenames as they are processed.

Remarks

CHKDSK does not work on the following:

❑ Disk drives used in ASSIGN, SUBST, or JOIN commands

❑ Networked disk drives.

Also note that CHKDSK cannot be used if TaskMAX is running with other sessions open.

Pending delete files saved by DELWATCH will be displayed with a ♣ character in the first position.

Examples

The following command checks the diskette in drive A:, and writes any corrections necessary back to the diskette:

```
C>CHKDSK a:  /F
```

The next command sends the CHKDSK status report directly to a file:

```
C>CHKDSK d:filename
```

CLS

Format

```
CLS [/?|/H]
```

Explanation

CLS clears the display screen. If foreground and background colors were selected previously, the colors will be unchanged. Otherwise, the screen is set to white characters on a black background. It is useful if you want to start with a fresh screen.

Command switches	
/? or /H	Displays help text.

Example

To start with a fresh screen, from the command prompt type:

```
C>CLS
```

COMMAND

Format

```
COMMAND [/?|/H][d:][dirpath] [CTTY-devicename]
[/E:nnnnn] [/P[:filename]] [/C string] [/ML|/MH|/MU]
```

Explanation

COMMAND starts a second copy of the DR DOS default command processor command.com. **Command.com** is a program which is loaded into memory every time you start DR DOS. It processes the commands you issue. In the case of internal commands, **command.com** can execute the commands directly. In the case of external commands and other applications, it arranges for the appropriate programs to be loaded from their files on disk. The COMMAND options are most commonly used by applications when they execute DOS internal commands.

The command processor executes in an *environment* which is the collection of all system variables such as PATH and PROMPT. When a command processor is started, it inherits the environment from its *parent*. Any changes made to the new environment (for example using SET or PROMPT), do not affect the primary environment.

The command processor resides in memory in two parts: the resident part which remains in memory all the time, as long as DR DOS is running, and the transient part which may be overwritten by some external commands or applications. If this happens, then the resident part can reload the transient part from the disk file **command.com** in the directory specified by the d: and dirpath options, when the application completes, and before the next command is processed.

The second copy of the command processor remains in memory until you issue an EXIT command.

You can specify a device name in the command syntax if you want to use a different device for input and output, e.g. AUX. See also the CTTY command description.

Note

COMMAND entered on its own causes a new copy of the command processor to be loaded.

Command switches

/? or /H — Displays help text.

/C*string* — Makes the command processor carry out the command(s) or programs specified in the string and then exit and return to the primary command processor. If this switch is used with other switches it must be the last one specified on the command line.

/E:*nnnnn* — The size of the environment in bytes where *nnnnn* is in the range 512 (minimum) to 32,751 (maximum). If you specify a value outside this range, the command defaults to the minimum value.

/P[:*filename*] — Fixes this copy of the command processor permanently in memory. The EXIT command is disabled. /P also causes **autoexec.bat** to be executed when **command.com** starts. Specify a *filename* with /P to execute a file other than **autoexec.bat**.

/ML | /MH | /MU — Forces this copy of the command processor to be loaded in a specific area of memory; /ML for conventional (lower) memory, /MH for high memory, or /MU for upper memory. Note that if there is insufficient high or upper memory available, the command processor is loaded into conventional memory. Before you can use /MH or /MU, a device driver that supports high and upper memory must be loaded.

Example

To set the environment size when you load the command processor, enter:

```
C>COMMAND /E:1024
```

COMP

Format

```
COMP [/?|/H] [wildspec wildspec] [/A] [/M:n]
```

Explanation

This command compares two files character by character and reports any character mismatches. COMP also reports if the End-Of-File (EOF) character is missing from any file that it examines.

Wildcards are allowed in the file specification. If you leave out the filenames altogether, COMP assumes a *wildspec* of *.* and prompts you for the source and destination.

Command switches

/? or /H	Displays help text.
/A	Displays the reported mismatches in ASCII format. (The default is to display them in hexadecimal format.)
/M:*n*	Specifies the maximum number of mismatches before COMP terminates automatically; 10 is the default, 0 allows an unlimited number of mismatches.

Examples

The following command compares all the **.doc** files in directory **c:\address** with files of the same name on the root directory of a diskette in drive A:

```
C>COMP  \address\ *.doc  a:\
```

The following command compares the file **today.doc** with the file **lastweek.doc** to see if the two contain identical information. The two files are both in the same directory, which is the current directory. Only five mismatches are allowed before COMP terminates automatically.

```
C>COMP today.doc lastweek.doc /M:5
```

COPY

Format

To copy files

```
COPY [switch] [/?|/H] wildspec|device [+wildspec]...
[switch] filespec|device [/A] [/B] [/V] [/S] [/C] [/Z]
```

Explanation

The main function of the COPY command is to copy one or more files to another location either on the same disk or another disk. It can also be used to combine files together into one file, to create files and to send files to a printer.

Password-protected files

COPY can be used with password-protected files but note:

- ❏ COPY does not prompt for a password.
- ❏ The destination copy of the file will not have password-protection unless you assign a password with the PASSWORD command.

Copying files

Wildspec is the *source* and must always be given. *Filespec* is the *destination.* The source and destination file specifications consist of the complete path, filename and filename extension.

If you omit the destination *filespec*, the "copy" is created in the current directory of the disk in the default drive, with the same name as the source file. You cannot copy a file to itself.

If you use wildcards in the source *wildspec*, the names of the files are displayed as they are copied.

If you type:

```
C>COPY *.* reports
```

all the files in the current directory would be copied to the directory **reports**. However, if the directory did not exist, ALL the files would be copied to a *single* file called **reports** in the current directory.

Specifying the destination filespec

If *filespec* is a drive name only, for example:

```
C>COPY file.txt a:
```

the file is copied to the specified drive with its original name.

WARNING If you enter a destination file specification that matches one already on the destination drive, the copy *overwrites* the existing file.

If *filespec* is a filename only, for example:

```
C>COPY file.txt new.txt
```

the original file is copied to a file on the current drive with the name given. Note that this is how you can use COPY to rename a file or files.

If *filespec* includes a drive name, for example:

```
C>COPY file.txt a:new.txt
```

the original file is copied to the file on the designated drive.

Note

To copy entire directories and subdirectories, use the command XCOPY.

COPY has several switches you can add to either the source or the destination specifications.

Command switches

/? or /H — Displays help text.

/A — Causes the file to be treated as an ASCII (text) file.

When you specify the /A switch with the source specification, the file contents are copied up to, but not including, the first **Ctrl+Z** (End-Of-File) character. Remaining file characters are not copied.

Using /A with a destination specification adds a **Ctrl+Z** as the last character in the file, which is required by some word processors.

/B — When applied to the source *wildspec*, /B causes the file to be treated as a binary file; the entire file is copied regardless of any **Ctrl+Z** character found in the file.

When you use /B with your destination *filespec*, no end-of-file character (**Ctrl+Z**) is added.

Note

The /A and /B switches only need to be entered when you are combining ASCII and binary files. These parameters are not necessary to copy data, program or batch files.

/V — Causes COPY to compare the destination file with the source file contents to verify the accuracy of the copy. Because of the comparison, it takes longer to complete the copy with the /V switch.

/S — Includes system and hidden files in the copy.

/C — Asks for confirmation before copying each file. If you are combining multiple groups of files, it confirms each group by first source name.

/Z — Zeros the top bit of every byte in the destination file. This switch should only be used with ASCII files. (It is designed for use with older-style word processors which use the eighth bit in every byte to mark beginnings

Command switches (continued)

and ends of words. This means text written in such a word processor would be unintelligible in a newer-style word processor. The /Z switch combats this problem.)

Examples

To copy the file **myfile** from the current drive to drive B:

```
A>COPY myfile b:
```

To copy the file **myfile.typ** on drive B: to the current drive A:

```
A>COPY b:myfile.typ
```

This command duplicates the file **myfile.typ** in the same directory and names the copy **yourfile.typ**:

```
A>COPY myfile.typ yourfile.typ
```

You can copy a file from one drive to a different drive by specifying drive designations *before* the filenames. This example copies **phone.lst** from drive B: to drive C:, and verifies the destination file against the source file for an exact match:

```
A>COPY b:phone.lst c:phone.lst /V
```

Combining files

To combine a number of source files into a single destination file, type COPY followed by the filenames to be combined, with a plus sign between them. Enter a space between the two combined files and the destination file. For example:

```
C>COPY file1.txt+file2.txt  result.txt
```

The files are combined in the order in which they are listed, or if there is a wildcard, in the order they are encountered in the directory. The destination file is always the last file named.

Caution If the files exist already, they are overwritten.

You should not combine files if one of the source files has the same name and extension as the destination file.

Example

To combine the file **myfile.typ** on drive A: with **yourfile.typ** on drive B:, copy the file to drive C:, and name it **result.typ**:

```
A>COPY myfile.typ+b:yourfile.typ
  c:result.typ
```

Copying files to a console or printer

You can use COPY to display or print a file. To do this, use the form:

```
COPY wildspec device
```

COPY can be used instead of TYPE to display or print a file. To perform this type of copy, enter CON or PRN as the destination. This command is valid only for text (ASCII) files, not application or program files.

The printer and console device names are shown in Table 10-5.

Table 10-5 COPY Device Names and Descriptions

Name	Description
CON	Designates your screen when it is the destination and your keyboard when it is the source device.
LPT*n*	Designates your destination printers where *n* is 1, 2, or 3.
PRN	Designates your current printer.
NUL	Designates a dummy output device and is usually used for program testing.
COM*n*	Designates the serial ports where *n* is 1, 2, 3, or 4.
AUX	Designates serial port 1 (COM 1).

Examples

To read the file **myfile** from drive D: and display it on the console:

```
A>COPY d:myfile con
```

To print all files with the file extension **.txt** on the current drive:

```
A>COPY *.txt prn
```

Creating a file with COPY

You can create a text file using COPY by specifying the keyboard as the source device:

```
C>COPY CON filespec
```

When you specify CON as the source device, every character you type in is recorded in the destination file you specified, except line-editing characters such as **Ctrl+H**.

Press **Ctrl+Z** or the **F6** key when you have finished typing your file, and close it by pressing **Enter**.

Examples

To create a file called **showme** with the text "this is my file", on the current disk, type:

```
A>COPY CON showme
  this is my file
```

After you have pressed **Ctrl+Z** to add the End-Of-File marker (which says this is where the file finishes) and then pressed **Enter**, the command prompt is returned to the screen.

You can now display the file **showme** by typing:

```
A>COPY showme con
```

CTTY

Format

```
CTTY [/?|/H] device
```

Explanation

Normally DR DOS takes input from your keyboard and sends it to your screen. Using the CTTY command you can redirect input and output to a different device, like a serial terminal; see Table 10-6 for devices you can specify.

Table 10-6 Devices You Can Specify With CTTY

Devices	Function
CON	Keyboard input and screen output. This is the default.
AUX or COM1	First serial (communication) port.
COM2	Second serial (communication) port.

Note

Before connecting a device to a port, you may need to set the correct communication parameters for the port using the MODE command.

Command switches	
/? or /H	Displays help text.

Examples

The following command hands over control to the device attached to the COM2 port, say a terminal:

```
C>CTTY COM2
```

To give control back to the keyboard type (from the terminal):

```
C>CTTY CON
```

CURSOR

Format

```
CURSOR [/?|/H] [/Snn] [/C] [OFF]
```

Explanation

On some types of displays (like LCD displays), the cursor blinks faster than the display can handle and so becomes a dim blur which is difficult to find.

The CURSOR command provides an easily visible cursor for use on laptops and portable computers with long persistence Liquid Crystal Displays. CURSOR works on CGA, EGA and VGA displays. The usual hardware-generated cursor appears as a single line under the character on which it is placed, whereas the software-generated cursor produced by the CURSOR command, appears as a large block. It has a variable flash rate to suit your own preference and the quality of the display you are using.

Command switches

/? or /H	Displays help text.
/S*nn*	Sets the flash interval, where *nn* is a 2 digit decimal value giving the cursor flash interval in multiples of 1/20 of a second. The permitted values are 1-20. The default is 4/20 of a second.
/C	Enables CGA compatibility. This prevents the snow-like interference that could be caused by using the software cursor on a CGA screen.
OFF	Disables the software cursor, and restores the hardware cursor.

Example

This command sets the software cursor with a flash interval of 6/20 of a second:

```
C>CURSOR /S6
```

DATE

Format

```
DATE [/?|/H] [nn-nn-nn]
```

Explanation

The DATE command displays or sets the month, day, and year.

DR DOS has its own internal clock, which needs to be set using the DATE and TIME commands each time you start the computer or reset DR DOS.

DR DOS maintains the internal clock for two reasons:

- ❑ To provide you and your applications continuous access to the date and time.
- ❑ To have the time available for file time and date stamping. (Time and date stamping records the current time and date whenever you create or update a file.)

To be prompted automatically whenever you start or reset your computer, include DATE and TIME commands in your **autoexec.bat** file. (There is no need for DATE or TIME commands in your **autoexec.bat** file if your computer has a battery-backed real-time clock.)

Command switches	
/? or /H	Displays help text.

If you do not enter the date, DR DOS assumes one. The DATE command always lets you change the current date. When you enter:

A>**DATE**

the current date is shown and you are prompted:

```
    Date:  day mm/dd/yy
    Enter  date(mm-dd-yy):
```

where *day* is the day of the week and *mm/dd/yy* is the month, day, and year (if you are using US format).

You can enter a new date or press **Enter** to keep the current values.

To set the date from the command line prompt type:

```
A>DATE nn-nn-nn
```

where the date is expressed in the format expressed in your country. The *nn-nn-nn* parameter must be made up of values *mm, dd,* and *yy,* in one of the following orders:

```
mm-dd-yy
dd-mm-yy
yy-mm-dd
```

where:

- ❑ *mm* is a one- or two-digit number from 1 to 12 indicating the month
- ❑ *dd* is a one- or two-digit number from 1 to 31 indicating the day
- ❑ *yy* is a two- or four-digit number indicating the year

You can use either a hyphen (-) or a slash (/) to separate your entries.

If you enter a date that is incorrect or out of range, you see:

```
    Invalid date specified
    Enter  date:
```

Enter the date as described above or press the **Enter** key to keep the previous value.

Remarks

The date display sequence has been preset according to the method used in the country where your system has been configured. The US form is *mm-dd-yy,* the European is *dd-mm-yy* and the Japanese, *yy-mm-dd.* You can change your configuration using the SETUP program or by making a COUNTRY entry in the **config.sys** file, see Chapter 11, "Customizing your system".

If your system has a real-time clock, the DATE and TIME commands will automatically update it.

DEL

Format

```
DEL [/?|/H] wildspec [/C] [/S]
```

Explanation

DEL removes the specified file or files from the directory. The file specification can identify a particular file, or it can indicate a group of files through the wildcard characters * and ?.

After you erase a file, you may not be able to retrieve it, so be very careful when you use wildcards. Generally, DEL does *not* query you before erasing files. However, it does query you when you use the asterisk (*) wildcard for both the filename and the extension (*.*). For *.* it displays the following prompt:

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

Type Y to erase all files in the directory; type N to cancel the command.

Command switches

/? or /H	Displays help text.
/C	Prompts you to confirm deletion.
/S	Includes system files.

Refer to the descriptions of DELWATCH, DISKMAP and UNDELETE for information about recovering deleted files.

Example

This command erases all the **.bak** files on the current directory without querying you.

```
A>DEL *.bak
```

DELPURGE

Format

```
DELPURGE [/?|/H] wildspec [/A] [/L] [/S] [/P]
[/D:date|/D:-nn] [/T:time]
```

Explanation

Use DELPURGE to free the disk space occupied by *pending de*lete files.

Pending delete files are saved by the DELWATCH command when you delete files. Refer to the DELWATCH command explanation for more information about pending delete files.

Command switches

/? or /H	Displays help text.
/A	Removes the specified files without prompting you first; the default is no prompt.
/L	Lists the specified files but does not remove them.
/S	Removes files in subdirectories below the specified subdirectory.
/P	Pauses after each screenful is displayed.
/D:*date* or /D:-*nn*	Removes files deleted before the specified date, if you specify a date, or files deleted before the last *nn* days, where *nn* is a number. If you specify /D:-10, for example, DELPURGE removes all files you deleted more than 10 days ago.
/T:*time*	Removes files deleted before the specified time only.

Example

The following command removes all the pending delete files with the filename extension **.bak**, in the root directory of drive C: and all subdirectories below it, prompting you before removing each file and pausing after each screenful:

```
C>DELPURGE c:\*.bak /S /P
```

DELQ

Format

```
DELQ [/?|/H] wildspec [/S]
```

Explanation

The DELQ and ERAQ commands have the same format and both erase files. They both query you to make sure you want to erase each file.

DELQ and ERAQ are most useful when used with the wildcard characters * and ? to erase an entire class of files. Instead of erasing every file that matches the file specification, DELQ and ERAQ list the specified files individually, followed by a ? prompt, to which you respond either with a Y for Yes (erase the file) or with an N for No (do not erase the file).

Command switches	
/? or /H	Displays help text.
/S	Includes system files.

Refer to the descriptions of DELWATCH, DISKMAP and UNDELETE for information about recovering deleted files.

Example

This command gives you the choice to delete each of the **.txt** files in the current directory.

```
A>DELQ *.txt
```

DELWATCH

Format

```
DELWATCH [/?|/H] d: [d:...] [/S] [/D] [/B:nnn]
[/O:ext[+ext...]] [/E:ext[+ext...]] [/F:nnn] [/MU]
[/ML]
```

Explanation

Use DELWATCH to save files when you delete them from directories.

Pending delete files

DELWATCH is a TSR you can optionally load. When DELWATCH is loaded, any file you delete *appears* to be deleted, but it is only marked instead as a *pen*ding delet*e* file. The file does not appear in the directory listings, but the file remains saved on the disk. You can therefore recover it at any time using the UNDELETE command. Refer to the UNDELETE command explanation in this chapter for information about how UNDELETE uses DELWATCH.

DELWATCH limits the number of pending delete files it saves to 200 by default, but you can set a different limit. A limit on the number of pending delete files to save can prevent the disk filling up with deleted files and affecting operating system performance. Once you set a limit, DELWATCH automatically begins removing the oldest pending delete files from the disk when it reaches the limit. Even if you do not set a limit (by specifying 65535), DELWATCH automatically removes the oldest pending delete files first if the disk becomes full.

Installing DELWATCH

You can load DELWATCH at the command line or automatically by including it in your **autoexec.bat** file. Use the SETUP program to edit **autoexec.bat**, or edit manually with the EDITOR.

WARNING Before you decide to install DELWATCH, be aware that since it reduces the amount of application memory available, it is automatically loaded into upper memory (if available), unless you force it to load into conventional memory using the /ML switch. Refer to Chapter 12, "Using MemoryMAX" for more information about using upper memory.

If you start DR DOS without loading DELWATCH, after loading it previously on the same system, the space occupied by existing pending delete files is not available. In this case, you can use the DELPURGE comand to free the space. Refer to the DELPURGE command explanation in this chapter.

Command switches

/? or /H — Displays help text.

/S — Displays the current status of DELWATCH on your system.

/D — Disables DELWATCH on the specified drives. Note that you cannot use any other switch with /D.

/B:*nnn* — Specifies how many files with the same name and dirpath to save. DELWATCH removes the oldest deleted files first when it reaches the limit. Specify a number between 1 (the default) and 65535.

/O:*ext*[+*ext*...] — Saves files with the specified extensions only. You can specify up to 10 extensions, and you can also use wilcard characters.

/E:*ext*[+*ext*...] — Saves all files *except* those with the specified extensions. You can specify up to 10 extensions, and you can also use wildcard characters.

/F:*nnn* | ALL — Specifies the maximum number of files to save. The default limit is 200 (20 for diskette). Specify 65535 for no limit until the disk becomes full. Specify ALL if you do not want any files removed, even if the disk becomes full; this guarantees that all deleted files can be recovered. Note that if you specify ALL, /B (if also specified) is ignored.

/MU — Do not install the DELWATCH TSR unless there is enough upper memory available for it. By default, DELWATCH is automatically loaded into upper memory, if available. Before you use /MU, a device driver that supports upper memory must be loaded.

Command switches (continued)	
/ML	Loads the DELWATCH TSR in conventional memory, even if upper memory is available.

Remark

When DELWATCH is loaded, the time it takes to delete files is increased. In particular, the *first* time a file is deleted after you install DELWATCH may take a considerable time, especially if you specified a large /F limit. However, subsequent file deletes are *not* noticeably slower with DELWATCH installed.

Pending delete files in the root directory

Unless you specify /F:ALL, note that pending delete files in your root directory may be removed if it becomes full, even if the limit of pending delete files to save has not yet been reached.

Example

The following command would load DELWATCH on drive C: and save all deleted files except those with extension **.bak**, up to a limit of 500. Once DELWATCH saves 500 pending delete files on drive C:, it will remove the oldest file to make space for each new file.

```
C>DELWATCH c: /E:bak /F:500
```

DIR

Format

```
DIR
[/?|/H][wildspec][/W|/L|/2][/D|/S|/A][/N][/P][/R|/C]
```

Explanation

DIR displays the contents of a disk directory. DIR with no parameters displays all files, other than system or hidden files (see the ATTRIB command), and all subdirectories in the current drive's current directory.

Subdirectories are separated from each other in the command line by a backslash (\).

You can search for a particular file. For example, if you type:

```
A>DIR myfile.gtm
```

DIR searches the current directory for **myfile.gtm**. If it cannot find the file, the following message is displayed:

```
    File not found
```

DIR shows the contents of the current or specified drive's current directory unless you specify another drive or path.

You can select a file or related group of files to display by entering a specific or a general file specification using the wildcard characters * and ?.

The DIR display shows the following information about the current directory:

- ❑ The drive you selected.
- ❑ The volume label of the current drive.
- ❑ The full path used for the file search.
- ❑ The list of file names.
- ❑ The number of files found.
- ❑ The amount of storage space left, in bytes.
- ❑ Whether or not any system files were omitted from the display.

The DIR switches can be entered individually, in combination and in any order.

Command switches

/? or /H	Displays help text.
/A	Displays all files.
/C	Makes other switches the default for the next DIR command issued; DIR /C /W, for example.
/D	Displays files where the system attribute is reset. This is a default switch.
/L	Includes size, date, and time. This is the default.
/2	Same as /L, except that files are displayed in two column format.
/N	Allows you to return to the default paging switch.
/P	Pauses at the end of each full page. The default is no paging.
/R	Tells DIR to remember new switches you specified as the default switch.
/S	Displays all files with the system attribute set.
/W	Displays only the names of files with backslashes (\) before directories.

Examples

The following command displays all files and subdirectories in drive C:’s current directory with the system attribute:

```
A>DIR c: /S
```

The next command displays all files and subdirectories with the **.bat** file extension in drive A: ’s current directory. It uses the “Long” format, giving information on size, and date and time stamping. The display pauses with each full screen. Files with the system attribute set are included and the switches remembered for the next DIR command without switches. (The filespec is not remembered.)

```
A>DIR *.bat /L /P /A /R
```

DISKCOMP

Format

```
DISKCOMP [/?|/H] [drivel] [drive2] [/1] [/8] [/A]
[/M] [/V]
```

Explanation

DISKCOMP compares two entire diskettes track by track. This is useful, for example, to check that a DISKCOPY operation has worked correctly.

If you specify DISKCOMP with no drives, DISKCOMP assumes the current drive.

If you omit *drive2*, DISKCOMP compares *drive1* with the current drive. If *drive1* and *drive2* are identical, DISKCOMP prompts you when you need to swap disks. If you omit both drives, DISKCOMP uses the current drive.

If a mismatch is found, DISKCOMP displays a "Compare" error message showing the track where the mismatch was discovered.

Making better use of memory

DISKCOMP automatically makes use of expanded memory, extended memory, and temporary files (in that order) to hold more diskette contents in memory so that it does not have to access the diskette so many times. This is particularly useful if you have a single disk-drive, because you will not have to change diskettes so often. Note that you must have an appropriate device driver loaded (EMM386.SYS or HIDOS.SYS, for example) before DISKCOMP can make use of expanded or extended memory. DISKCOMP will automatically use a temporary file specified with the environment variable TEMP. Refer to the SET command description in this chapter for more information about the TEMP variable.

Comparing image files

You can use DISKCOMP to compare an image file (created using DISKCOPY) with a diskette. Note that you cannot compare an image file with another image file.

You can also use DISKCOMP to verify that an entire diskette or image file can be read, by specifying the /V switch.

Command switches

/? or /H Displays help text.

/1 Compares the first side only of a double-sided diskette.

/8 Compares only the first eight sectors per track of a diskette.

/A Causes the computer to sound a "beep" when the comparison is complete or when you need to change diskettes.

/M Allows you to make multiple comparisons of a single image file or diskette. You are prompted to insert the first diskette to be compared. When the comparison is complete you are asked if you want to compare another diskette. If you type Y for yes, you are prompted to insert the next diskette, and so on, until you have completed all the comparisons you want to make.

/V Verifies that the whole diskette or image file can be read.

Remarks

DISKCOMP can only be used to compare diskettes copied using DISKCOPY.

DISKCOMP does not work with networked drives or with drives affected by an ASSIGN, JOIN or SUBST command.

The first diskette to be compared must be the same type as the second diskette, that is, they must *both* be 360 Kbytes single-sided single-density, for example, or 1.2 Mbytes double-sided double-density. If they are not of the same type, you will get an error message.

Examples

The following command compares the diskette in drive A: with the diskette in drive B:

```
A>DISKCOMP a: b:
```

The next command verifies the diskette in drive B:

```
A>DISKCOMP b: /V
```

The following command compares a diskette against multiple copies using drive A: only:

```
C>DISKCOMP a: a: /M
```

The following command compares the image file **diska.img** in directory **c:\image** with a diskette in drive B:, sounding a beep when the comparison is complete:

```
C>DISKCOMP c:\image\diska.img  b:  /A
```

DISKCOPY

Format

```
DISKCOPY [/?|/H] [source_drive] [destination_drive]
[/1] [/A] [/M]
```

Explanation

The DISKCOPY command copies entire diskettes of the same format. If the destination diskette has a different format, or is unformatted, DISKCOPY will format it to match the source diskette (where this is possible). If you want to copy files between diskettes with incompatible formats, use the XCOPY command.

WARNING DISKCOPY works with any removable media, but not with a hard disk. DISKCOPY erases anything that exists already on the destination.

If you omit *destination_drive*, DISKCOPY copies to the default drive, not the *source_drive*. If the source and destination drives are identical, DISKCOPY prompts you when you need to swap diskettes. If you omit both drives, DISKCOPY copies to the default drive.

Making better use of memory

DISKCOPY automatically makes use of expanded memory, extended memory and temporary files (in that order) to hold more diskette contents in memory so that it does not have to access the diskette so many times. This is particularly useful if you have a single disk-drive, because you will not have to change diskettes so often. Note that you must have an appropriate device driver loaded (EMM386.SYS or HIDOS.SYS, for example) before DISKCOPY can make use of expanded or extended memory. DISKCOPY will automatically use a temporary file specified with the environment variable TEMP. Refer to the SET command description in this chapter for more information about the TEMP variable.

Copying to image files

You can use DISKCOPY to store an image of a diskette into a file; that is, an exact copy of the diskette's contents. This image file can then be copied to, or compared against (using DISKCOMP), another diskette. For example, if you type

```
C>DISKCOPY a: c:\image\diska.img
```

the contents of drive A: are copied to the file **diska.img**, in directory **c:\image**. Using the /M switch (see below), you could use the image file **diska.img** as a source for multiple copies to diskettes.

Command switches

/? or /H Displays help text.

/1 Copies only side 0 of a diskette, enabling to copy single-sided diskettes.

/A Causes the computer to sound a "beep" when the copy is complete or when you need to change diskettes.

/M Allows you to make multiple copies of a single source. The source must be an image file or drive and the destination must be a drive. The source is copied into memory and you are prompted to insert the destination diskette. When the first copy is complete, you are asked if you want to copy another diskette. If you type Y for yes, you are prompted to insert the next destination diskette, and so on, until you have completed the number of copies you want to make.

Remarks

If you want to copy a complete structure of directories and the volume label from one type of disk to another, you can use the XCOPY command.

Note the following:

- A 180 Kbyte drive can copy 160 and 180 Kbyte diskettes.
- A 360 Kbyte drive can copy 160, 180, 320 and 360 Kbyte diskettes.
- A 1.2 Mbyte drive can copy 1.2 Mbyte diskettes. It can also copy 160, 180, 320 and 360 Kbyte diskettes, but you may not be able to read these diskettes in a 180 or 360 Kbyte drive.
- A 720 Kbyte drive can only copy 720 Kbyte diskettes.

❑ A 1.44 Mbyte drive can copy 720 Kbytes and 1.44 Mbyte diskettes.

Examples

If you enter the command:

```
A>DISKCOPY a: a:
```

DISKCOPY prompts you to insert the source diskette. When the information on the source diskette is copied into memory, you are prompted to insert the destination diskette. The information in memory is then copied onto the destination diskette. This may be repeated several times, depending on the size of the disk.

The following command generates the image file **diskb.img** in directory **c:\image** from a diskette in drive B:

```
C>DISKCOPY  b:  c:\image\diskb.img
```

The next command copies the image file **diskb.img** to a diskette in drive A:, sounding a beep when the copying is complete:

```
C>DISKCOPY  c:\image\diskb.img  a:  /A
```

The next command makes multiple copies of the image file **diskb.img** to drive B:

```
C>DISKCOPY  c:\image\diskb.img  b:  /M
```

DISKMAP

Format

```
DISKMAP [/?|/H] [/D] drive: [drive: [drive:]...]
```

Explanation

DISKMAP provides a way of saving information about deleted files to help UNDELETE with file recovery operations.

DISKMAP makes a copy of the current File Allocation Table (FAT) on a disk, and saves the copy until you run DISKMAP again to make a new copy. The FAT is an index, or "map", that exists at the start of every disk, and the operating system uses it to locate file entries. DISKMAP provides a pointer to the original location of a deleted file, and as long as the space occupied by the file has not been reallocated since the delete, UNDELETE should be able to recover the file. DISKMAP also records the number of files contained in a directory, information that UNDELETE could not determine with *unaided* file recovery.

The advantage of using DISKMAP rather than DELWATCH, to recover deleted files, is that DISKMAP has no effect on your application memory. Remember, however, that using DISKMAP does not guarantee successful file recovery like DELWATCH. Refer to the UNDELETE command explanation in this chapter for information about how UNDELETE uses DISKMAP, and DELWATCH.

Command switches	
/? or /H	Displays help text.
/D	Erases the existing DISKMAP file, if one exists, and writes a completely new file.

To use DISKMAP on the command line, type DISKMAP on its own or followed by a switch or disk drives. If you specify a drive, **diskmap.dat** (the DISKMAP file) is created on the disk and overwrites the existing DISKMAP file, if present. If you type DISKMAP on its own, the existing **diskmap.dat** on the current drive is updated.

Rather than running DISKMAP yourself, you can also run it automatically by doing either of the following:

- ❑ Including it in your **autoexec.bat** file. Use the SETUP program to automatically alter **autoexec.bat**, or edit manually with the EDITOR.
- ❑ Changing the DR DOS command prompt to run a DISKMAP command every time you return to the prompt. Use PROMPT and SET commands, either on the command line or in **autoexec.bat**. Refer to the PROMPT command description for information about how to do this.

Example

The following command would copy the FATs for drive C: and drive A: and save them on each disk respectively as the current DISKMAP files.

```
C>DISKMAP c: a:
```

DISKOPT

Format

```
DISKOPT [/?|/H] d:
```

Explanation

Use DISKOPT to optimize the performance of a hard disk.

DISKOPT is a menu-driven utility that changes the position of data on a disk to make files contiguous, move all free space to the end of the disk, and (optionally) sort directories.

The only parameter you specify with DISKOPT is the drive of the disk you want to optimize.

Command switches	
/? or /H	Displays help text.

Read Chapter 13, "Optimizing your disk" for a complete explanation about how to use DISKOPT.

WARNING Before you use DISKOPT, you should know how it works; refer to the DISKOPT description in Chapter 13, "Optimizing your disk".

DOSBOOK

Format

```
DOSBOOK [/?|/H][command name][/B]
```

Explanation

DOSBook contains information on the three areas listed below:

❑ DR DOS Basics.

❑ Commands and Utilities.

❑ Troubleshooting.

Command switches

/? or /H	Displays help text.
/B	Forces menus to be displayed in black and white. (Some mono displays attempt to simulate color displays with shades of grey, and it can be difficult to distinguish them.)

If you type:

C>**DOSBOOK**

a window appears on your screen, displaying a welcome message. You can press **F1** to learn more about using the DOSBook or you can select one of the three areas given above. Press the **Tab** key to highlight a word or phrase and then press the **Enter** key to see the information. If you select one of the three areas listed above, from the welcome screen, and press **Enter**, a contents list will be displayed. You can use the **Tab** key to highlight the different sections. When you have highlighted the section of information you are interested in, press **Enter** to view it.

You can scroll through the text using the ↑ and ↓ cursor keys. The **Page Up**, **Page Down**, **Home** and **End** keys can be used to move through the text. They are described in Table 10.7.

Table 10.7 DOSBook Keyboard Commands

Keystrokes	Meaning
↑ Page Up **↓ Page Down**	Scrolls through the help text.
Home End	Jumps to the start or end of the current screen display.
Ctrl+Home	Jumps to the start of the current section.
Ctrl+End	Jumps to the end of the current section.
Tab Shift+Tab	Selects the next or previous key word.
Enter	Displays the information associated with the selected key word.

The scroll bar at the right-hand side of the DOSBook window shows how much of the current section is displayed. If a section contains more information than can be displayed in the window, the scroll bar slider moves as you page down through the text until you reach the last page.

There may be highlighted words within the text of a section. These are called *key words* and they indicate that there is further information available. If you select one of these words and press **Enter** you see this additional information. There are two different types of highlight for key words which indicate that the additional information is displayed in one of two ways. It appears either as a short note that is displayed while you hold the **Enter** key pressed down, or a different section of information appears. The highlight differs according to the type of monitor (colour or monochrome) you are using.

Note

If you type:

```
C>DOSBOOK
```

followed by the name of a command, DOSBook will display the information on that command instead of displaying the welcome screen.

At the top left of the DOSBook window are two pull-down menus. You can select either of the menus by typing **Alt+W** to open the Window menu or **Alt+H** to open the Help menu. You close the menu by pressing the **Esc** key.

Once a menu is selected you can move between the two menus by pressing the left ← and right → arrow keys. Within the menu you select an option by typing the highlighted letter. Some of the options may be selected outside the menus by pressing the **Alt** key and a letter. Options that only have a highlighted letter can only be selected from within the open menu.

Table 10.8 Window Menu

Option	Explanation
Resize window (**Alt+F5**)	Enlarges/decreases the size of the window.
Print section	Prints the currently displayed section of text.
Exit, retain window	Exits DOSBook but keeps the window on screen.
Exit (**Alt+X**)	Exits DOSBook and closes the window.

Table 10.9 Help Menu

Options	Explanation
Help for help (**F1**)	Displays this help section.
Contents (**Alt+C**)	Displays the contents section.
Glossary (**Alt+G**)	Displays the glossary section.
Index (**Alt+I**)	Displays the index section.
Backtrack (**Alt+B**)	Returns to the last section you viewed.
Next (**Alt+N**)	Displays the next section.
Previous (**Alt+P**)	Takes you to the previous section.
About	Copyright information.

ERAQ

Format

```
ERAQ [/?|/H] wildspec [/S]
```

Explanation

The ERAQ and DELQ commands have the same form and each erases files and queries you before erasing a file. The ERASE command is similar to these commands, but in most cases erases files without queries.

ERAQ and DELQ are most useful when used with the wildcard characters * and ? to erase an entire class of files. Instead of erasing every file that matches the file specification, ERAQ and DELQ list the specified files individually, followed by a ? prompt, to which you respond either with a Y for Yes (erase the file) or with an N for No (do not erase the file).

Command switches

/? or /H	Displays help text.
/S	Includes system files.

Refer to the descriptions of DELWATCH, DISKMAP and UNDELETE for information about recovering erased files.

Example

This example gives you the choice to delete each of the **.bak** files in the current directory.

```
A>ERAQ *.bak
```

ERASE (ERA)

Format

```
ERASE|ERA [/?|/H] wildspec [/C] [/S]
```

Explanation

ERASE and ERA have the same format and each removes the specified file or files from the directory. The file specification can identify a particular file, or it can indicate a group of files through the wildcard characters * and ?.

After you erase a file, you may not be able to retrieve it, so you should be very careful when you use wildcards. Generally, ERASE does NOT query you before erasing files. However, it does query you when you use the asterisk (*) wildcard for both the filename and the extension (*.*). For *.* it displays the following query:

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

Type Y to erase all files on the (directory); type N to cancel the command.

Command switches	
/? or /H	Displays help text.
/C	Prompts you to confirm deletion.
/S	Include system files.

Refer to the descriptions of DELWATCH, DISKMAP and UNDELETE for information about recovering erased files.

Example

This command erases all the **.bak** files on the current directory, without querying you.

```
A>ERA *.bak
```

EXE2BIN

Format

```
EXE2BIN [/?|/H] file1[.ex1] [file2[.ex2]] [/Sxxxx]
```

Explanation

EXE2BIN is a utility which converts **.exe** type executable program files produced by a DOS Linker into a binary image (**.bin**) or **.com** executable file.

EXE2BIN is mainly for use by program developers.

Files with the extension **.exe** produced by a DOS Linker have a special header which includes a relocation table. The header provides DOS with information on the size, memory requirements and entry point of a program as well as the locations in the program of absolute segment address references. Files with the extension **.com** do not contain this special header as they are absolute memory images of the program to be executed. Whether the file produced is a **.bin** or .com type file depends on what is set in the initial IP (Instruction Pointer) field in the source **.exe** file header. If the IP field contains a value of 100H, a **.com** file is produced. If the IP field contains a value of 0, a **.bin** file is the result. This is not applicable if the destination filename extension (**ex2**) is specified.

A **.exe** program must have the following characteristics before it can be converted to a **.com** file:

- ❑ The program must have a maximum size once loaded of 65277 bytes.
- ❑ The program must have its entry point at offset 100H within its segment.
- ❑ The program must not have a stack segment declared.
- ❑ The program must have no absolute segment address references which require segment fixups stored in the **.exe** header. (Segment fixups are needed if the program contains instructions requiring segment relocation, which is usually done by the DOS loader using the **.exe** file's relocation table.)

For a **.exe** program to be converted to a **.bin** file it must:

- ❑ have its entry point at absolute offset 0 in its memory usage.
- ❑ not have a stack segment declared.

Command switches

/? or /H — Displays help text.

file1 — The path to, and name of, the source **.exe** file. If no extension is specified a default of **.exe** is used.

file2 — The path to, and filename of, the destination file to be produced. If *file2* is not specified, *file1* is used with an extension of **.com** or **.bin**. If *file2* is specified but without an extension, the source file is used with a **.com** or **.bin** extension, depending on the contents of the source **.exe** header.

/S*xxxx* — /S followed by four hexadecimal digits gives the base segment value to be used if any segment fixups are required for the **.bin** file. If fixups are needed, but this parameter is omitted, you are prompted to enter a value.

Examples

This command converts the **.exe** file **data.exe** to the file **data.bin**, type:

```
C>EXE2BIN data
```

The following command converts the file **utility.tst** (which includes a valid **.exe** header and fulfils the requirements for conversion to a **.com** file) to the file **result.fin** (which will be a valid executable file):

```
C>EXE2BIN utility.tst result.fin
```

The following command converts the file **utility.tst** (which includes a valid **.exe** header and fulfils the requirements for conversion to a **.bin** file) to the file **abinfil.fin** (which will be a valid **.bin** file), using a base segment fixup value of 1234 hexadecimal:

```
C>EXE2BIN utility.tst abinfil.fin /S1234
```

EXIT

Format

```
EXIT [/?|/H]
```

Explanation

This command returns you to another program. Some programs allow you to leave them, work with DR DOS commands, and then return to what you were doing before. EXIT is the command you use to return from the DR DOS prompt to the program.

EXIT also stops a secondary **command.com** if one is loaded, (see the entry for COMMAND).

Command switches	
/? or /H	Displays help text.

FASTOPEN

Format

```
FASTOPEN [/?|/H]
```

Explanation

The FASTOPEN statement used in **config.sys** reduces the time taken to access files on a hard disk which are being accessed repeatedly. It does this by remembering the locations of the files on disk, so that DR DOS does not have to navigate along the path every time the file is opened.

DR DOS sets up a special table of these filenames in memory and you can set the size of this table by inserting a FASTOPEN statement in your **config.sys** file using SETUP. See Chapter 11, "Customizing your system" for more information.

FASTOPEN.EXE is supplied with DR DOS, but does not perform any function. It is supplied explicitly for applications which require it to exist.

Command switches	
/? or /H	Displays help text.

FC

Format

```
FC [/?|/H] [@]oldwildspec newwildspec [/A] [/B] [/C]
[/Gn] [/L] [/Mn] [/W]
```

Explanation

FC compares the contents of ASCII text files or binary files and displays the differences found.

You can use FC to compare, for example, two versions of a document written with the DR DOS EDITOR to determine what changes are required to the old version to make it the same as the new version. You could also use FC to compare versions of a program you are developing to determine what changes were made to the first version to produce the second version.

You can compare two files or, as long as you are comparing files with the same name, use wilcard characters to compare groups of files. You can also compare the contents of a filelist with files (of the same name) in another directory.

Note

You can send the output of the FC command to the screen or a file by using the redirection symbol. See Chapter 8, "Redirecting input and output".

FC determines the type of files you are comparing from the filename extension and performs a comparison in either ASCII mode or binary mode, depending on the file type.

ASCII text files

FC compares the contents of ASCII text files line by line and then indicates what lines need to be inserted, replaced, or deleted in the old file to make it match the new file. Note that you see a message indicating that the files match, if this is the case.

For example, suppose you wanted to compare the following files:

```
autoexec.old                     autoexec.bat

:DRDOSBEG                        @ECHO OFF
PATH D:\DRDOS;C:\DOS;C:BATS      :DRDOSBEG
HILOAD KEYB UK+                  PATH D:\DRDOS;C:\DOS;C:BATS
VERIFY OFF                       HILOAD KEYB UK+
HILOAD SHARE /L:20               VERIFY OFF
C:\GLMOUSE\GLMOUSE               HILOAD SHARE /L:20
PROMPT $P$G                      PROMPT $P$G
MEMMAX -U -L >NUL                MEMMAX -U -L >NUL
:DRDOSEND                        :DRDOSEND
```

If you type the following command:

```
C>FC autoexec.old autoexec.bat
```

FC makes the comparison and since the two files differ, you see the following information:

```
After line 0 in c:\AUTOEXEC.OLD insert line 1 from c:\AUTOEXEC.BAT
> @ECHO OFF

Delete line 6 in c:\AUTOEXEC.OLD
< C:\GLMOUSE\GLMOUSE
```

Resynchronizing files

When it compares ASCII text files, FC *resynchronizes* the files each time it encounters a mismatching line. This means that if an entire line has been added or deleted in the new file, FC adjusts the line numbers it is comparing so that the lines match.

For example, assume the following lines in **autoexec.old** and **autoexec.bat**:

autoexec.old	**autoexec.bat**
Line 1 :drdosbeg	**Line 1** @echo off
Line 2 path d:\drdos;c:\dos;c:bats	**Line 2** :drdosbeg
Line 3 hiload keyb uk+	**Line 3** path d:\drdos;c:\dos;c:bats

FC will find that **Line 1** of **autoexec.old** does not match **Line 1** of **autoexec.bat**. If FC did not resynchronize the files, it would list the remaining lines of **autoexec.old** as mismatches; but by resynchronizing the files, FC adjusts the line numbers so that **Line 1** in **autoexec.old** is compared with **Line 2** in **autoexec.bat**, and **Line 2** in **autoexec.old** with **Line 3** in **autoexec.bat**, and so on until the next difference is found.

Binary files

FC compares binary files (**.bin**, **.cmd**, **.com**, **.exe**, **.lib**, **.obj** and **.sys** file types) byte by byte and then displays the differences found in hexadecimal form. You see a display similar to the following example:

```
Binary comparison between c:\PROGRAM1.EXE and c:\PROGRAM2.EXE

                    Offset       Hex  ASCII     Hex   ASCII
                    00000A       63    c        32     2
                    00000C       63    c        32     2
                    000019       66    f        67     g
                    00001B       66    f        67     g
                    00001E       67    g        68     h
                    000020       67    g        68     h
                    000023       68    h        69     i
                    000025       68    h        69     i
                    000028       69    i        6A     j
                    00002A       69    i        6A     j
                    00002D       6A    j        6B     k
                    00002F       6A    j        6B     k
                    000032       6B    k        33     3
                    000034       6B    k        33     3
                    c:\PROGRAM1.EXE is longer than c:\PROGRAM2.EXE
```

Note that FC does **not** resynchronize files when it finds a difference during a binary comparison.

As shown in the example above, you see the following message:

```
file1 longer than file2
```

when one file has fewer lines of data than the other. You only receive this message with a binary comparison.

When FC reaches 20 mismatches between binary files, you see the following message:

```
comparison stopped after 20 mismatches
```

You can change the default from 20 using the /M*n* switch; see command switches below.

Command switches

/? or /H Displays help text.

/A Abbreviates the ouput during an ASCII text file comparison. Use /A when the files you are comparing are likely to result in a large amount of output. An abbreviated output means that only the first and last lines of each block of differences are displayed; the lines in between are represented by an ellipsis (...). For example:

```
Replace lines 6-9 in c:\AUTOEXEC.OLD
< C:\GLMOUSE\GLMOUSE
< (...)
< :DRDOSEND

with lines 7-9 from c:\AUTOEXEC.BAT
> PROMPT $P$G
> (...)
> :DRDOSEND
```

/B Forces FC to perform a binary comparison. Use /B when you want to compare files that do not have standard filename extensions but that you want to compare byte by byte.

/C Forces FC to ignore case (that is, whether letters are uppercase or lowercase) during an ASCII text file comparison. For example, if you use /C, the following lines will match:

```
THIS PROGRAM LINE WILL MATCH

This program line will match
```

/G*n* Changes the number of lines that must match before resynchronization occurs. The default is 5.

/L Forces FC to perform an ASCII text file comparison. ASCII (line by line) comparison is the default when you are comparing files that do not have standard binary filename extensions.

Command Switches (continued)

/M*n* — Changes the maximum number of mismatches that can be found during a binary comparison; when FC reaches the maximum, the file comparison terminates. The default is 20. Use /M0 to specify an unlimited number of mismatches.

/W — Forces FC to ignore spaces and tabs during ASCII text file comparison. FC ignores spaces and tabs *within* a line but CRs (carriage returns) and LFs (line feeds) are not ignored. You can use /W to limit mismatches to words only.

If you use /W, the following two lines will match, for example:

```
This line will match
 Thislinewillmatch
```

Examples

The following command compares **plan.doc** on the current drive with **plan.doc** on drive B:, in ASCII mode:

```
A>FC plan.doc b:\
```

Note that you do not need to specify **plan.doc** after **b:\**.

The next command compares **program1.com** with **program2.com**, in binary mode:

```
A>FC program1.com program2.com /B
```

The following command compares all files in the directory **c:\dr** with matching files (files with the same name) in directory **d:\dr**, defaulting to ASCII or binary mode depending on the filename extension of each file:

```
A>FC c:\dr\*.* d:\dr\*.*
```

The next command redirects the output of FC to a file called **result.doc** on drive D:

```
A>FC autoexec.old autoexec.bat > d:\result.doc
```

The following command uses a filelist to compare a subset of files with the same subset in directory **\drdos**:

```
C>FC @filelist.fl \drdos
```

filelist.fl contains the following:

```
program1.exe
program2.exe
program3.exe
```

Therefore, FC compares:

program1.exe with **c:\drdos\program1.exe**
program2.exe with **c:\drdos\program2.exe**
program3.exe with **c:\drdos\program3.exe**

FDISK

Format

```
FDISK [/?|/H] [/D]
```

Explanation

FDISK is a menu-driven utility used to divide a hard disk into partitions. It also formats the hard disk.

Before you can use a hard disk you need to partition it. This means to define *sections* on the disk. The maximum size section or partition you can have is 512 Mbyte (you can choose to have smaller partitions if you wish). If you are preparing a hard disk for the first time you will need to prepare a special DOS partition in which to install DR DOS. This is called the *primary* partition; see "Create DOS primary partition" later in this section.

Using FDISK you can create one primary DOS partition and one extended DOS partition, which may contain as many logical disk drives as you require. You can then pass information between DOS partitions on a hard disk as if between different drives.

For example, if the disk is divided into a primary partition and an extended partition containing two logical drives, you can address the partitions as follows:

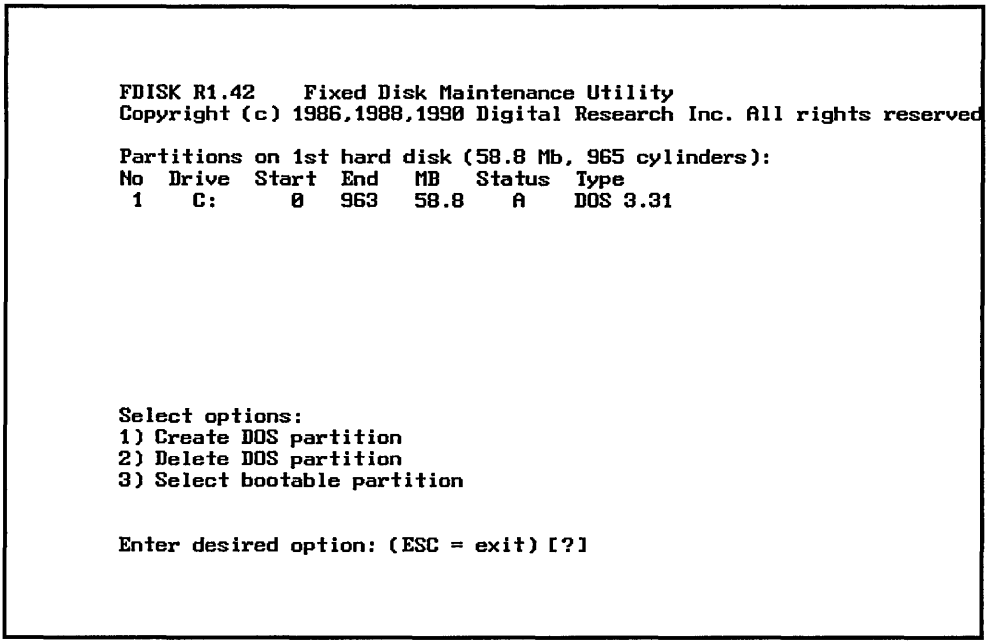
- ❑ C: is the primary partition.
- ❑ D: is the first logical drive on the extended partition.
- ❑ E: is the second logical drive on the extended partition.

When FDISK partitions a disk, the size of the clusters it creates is a function of the disk size; the larger the disk, the larger the clusters.

Command switches	
/? or /H	Displays help text.
/D	Allows the deletion of an existing non-DOS partition. Use with care!

Starting FDISK

When you type FDISK, the current partition status of the hard disk is displayed followed by a Main Menu of options, something like the following example:



```
FDISK R1.42    Fixed Disk Maintenance Utility
Copyright (c) 1986,1988,1990 Digital Research Inc. All rights reserved

Partitions on 1st hard disk (58.8 Mb, 965 cylinders):
No  Drive  Start  End   MB    Status  Type
 1    C:       0  963   58.8    A     DOS 3.31






Select options:
1) Create DOS partition
2) Delete DOS partition
3) Select bootable partition


Enter desired option: (ESC = exit) [?]
```

Note

If your hard disk contains an extended partition, you also see option 4 `Display logical drives in extended partition`. If your computer has more than one hard disk, you also see option 5 `Select alternative fixed disk`.

Press **Esc** to return to DR DOS from the Main Menu, or to return from a chosen option to the Main Menu.

The different items of status information shown are:

Field	Meaning
No	Each partition in the disk is numbered in the order in which partitions appear on the disk, starting at 1.
Drive	The drive letter assigned to the hard disk drive, starting with C.
Start	The starting cylinder number for the partition.
End	The ending cylinder number for the partition.
MB	The size of the partition in Mbytes.
Status	Either **A** or **N.** **A** indicates that the partition is *active*, that is bootable (your computer attempts to boot from this partition if started from the hard disk). **N** means *not active.*
Type	The format of the partition. This can be DOS (for example DR DOS, Concurrent DOS or another DOS operating system) or non-DOS.

The options you can select from the Main Menu are described below.

Option 1, Create DOS partition

Select this option if you are:

❑ Preparing a hard disk for the first time.

❑ Adding a new partition to a hard disk.

FDISK warns you if there is not enough space to create a new partition or if all partitions are already in use.

When you select this option, you are asked which type of partition you want to create:

```
Select options:
1) Create DOS primary partition
2) Create DOS extended partition
3) Create logical drive in extended partition


Enter desired option: (ESC = exit) [?]
```

Remarks

- □ You can create a primary partition only when you format a hard disk for the first time and no primary partition already exists on the disk.
- □ You can create an extended partition only if a primary partition already exists on the disk.
- □ You can create a logical drive only if you have an extended partition.
- □ A warning message is displayed if you make an invalid selection.

Create DOS primary partition

When you select option 1 `Create DOS primary partition` you are asked if you want to use the largest available space on the hard disk (this example is for a 40.2 Mbyte drive):

```
Use cylinders 1 - 975 for DOS (40.2 MB) (Y/N)?
```

If you want to use all the available space, type Y to accept the default selection and press **Enter**.

If you do not want to use all the available space (for example if you want to have an additional operating system), type N and press **Enter**.

Note

If you plan to add another operating system to the hard disk, check the size requirements for that system *before* you allocate space to DOS.

You are then prompted for the starting and ending cylinders:

```
[Enter] starting cylinder:
```

```
[Enter] ending cylinder:
```

You need at *least* 20 cylinders to run DR DOS.

If FDISK encounters a defective track on the system area of the disk, it does not create a partition. FDISK prompts you to try again with another starting cylinder number. Similarly, if there is insufficient space at a cylinder, FDISK prompts you to try another cylinder number.

You are prompted to:

```
[Enter] disk label:
```

You can give the partition a label of up to eleven characters.

Once you have chosen your partition size you are returned to the Main Menu. The screen displays details of the new partition you have created. Press **Esc** to leave the FDISK program.

The partitions you create contain no program or data files so you must load the files you want into the appropriate partition. For example, you may want to load DR DOS into a partition so that you can boot the system from the hard disk. If you are using FDISK as part of the DR DOS installation procedure, the INSTALL program automatically does this for you. So when you have chosen your partition size and you are asked to reboot, simply leave your DR DOS diskette in drive A: to restart the INSTALL program.

You can use INSTALL to load DR DOS into your partition or you can switch to drive A: and use the SYS command. Then use option 3 `Select bootable partition` (described in this section) to make the partition active.

Create DOS extended partition

Select option 2 `Create DOS extended partition` if you already have a DOS primary partition and want to add another partition to your disk. You are asked to specify the starting and ending cylinders for the partition, in the same way as creating a DOS primary partition. You are then returned to the Main Menu and the screen displays details of the partition you have just created.

Note

You cannot use an extended partition until you have created one or more logical drives within it. The extended partition is not assigned a drive letter of its own.

Create logical drive in extended partition

Choose option 3 `Create logical drive in extended partition` to create a logical drive in an existing extended partition. When you select it the screen displays the current status of the extended partition, for example:

```
Logical drives in extended partition (8.2 Mbyte, 195 cylinders):

No      Drive     Start    End      MB       Type
1        D:       771      785      0.6      DOS 2.0
```

You are then asked to specify the starting and ending cylinders for the logical drive. All logical drives must be contained completely within the extended partition. When the drive has been created you are offered the choice of creating another logical drive or returning to the Main Menu. Press **Esc** to return to the Main Menu.

Option 2, Delete DOS partition

Select this option to delete a current partition.

WARNING When a partition is deleted, you can no longer access data in that partition, so use this option with care. Back up first all files held in the partition (either on diskettes or in another partition on the hard disk).

When you select this option, you are asked which type of partition you wish to delete:

```
Select option:
1) Delete DOS primary partition
2) Delete DOS extended partition
3) Delete logical drive in extended partition


Enter desired option: (ESC = exit) [?]
```

Note

You can only choose option 2 to delete an extended partition if the partition does not contain any logical drives. If the partition you want to delete contains logical drives you must delete them first.

If you select option 1 `Delete DOS primary partition` or option 2 `Delete DOS extended partition` you are prompted to:

```
    Enter number of partition to delete (ESC=exit)
```

Type in the number of the partition you want to delete and press **Enter**, you will see:

```
    Do you really want to erase all data (Y/N)?...
```

If you are ready to delete the partition, type Y and press **Enter**.

If you select option 3 `Delete logical drive in extended partition` the screen displays the logical drives in the extended partition before prompting you to select a drive to delete.

Deleting a non-DOS partition

To delete an existing non-DOS partition, type FDISK /D at the command line to start the FDISK utility in the normal way. The procedure is then the same as for deleting a DOS partition.

If you do not specify /D at the command line, you are not allowed to delete a non-DOS partition.

Option 3, Select bootable partition

Use this option to select which partition on a hard disk is to be *active*. This is the partition containing the operating system and files you want to access when you switch on your computer. If you designate a particular partition as active, your computer searches this partition if DR DOS is booted from the hard disk. Only one partition can be active at a time.

When you select this option, you are prompted to:

```
    Select partition to make bootable.........
```

Type in the number of the partition you want to make active and press **Enter**.

Notes

If your hard disk contains only DR DOS partitions, then only the primary partition can be made active. If you try to select an extended DR DOS partition you will get an error message.

This partition now has an active status (A). The partition that was previously active is now non-active (status N).

Your computer always tries to load an operating system from drive A:. If there is no diskette in drive A: or if drive A: is not working, the computer tries to load an operating system from the active partition on the hard disk.

Option 4, Display logical drives in extended partition

You only see this option in the Main Menu if you have an extended partition on your disk. If you select it, the screen displays the current status of the logical drives in your extended partition, for example:

```
Logical drives in extended partition (8.5Mb, 205 cylinders):

No  Drive Start  End  MB   Type (i.e no status)

1    D     771   975  8.4  DOS 2.0
```

Option 5, Select alternate fixed disk

You only see this option in the Main Menu if you have multiple hard disks on your computer. Select it to switch from one disk to another. The screen displays the current partition status of the disk as follows:

```
Partitions on 2nd hard disk (20.3 MB, 614 cylinders):

No  Drive Start  End  MB    Status  Type
1    F:   1      304  10.1    A     DOS 3.0
2    -    305    613  10.2    N     (other)
```

The options displayed are the same for both disks.

FILELINK

Format

```
FILELINK [/?|/H] command [@][wildspec1] [wildspec2]
[comparm] [/switches]
```

Explanation

FILELINK is a utility used to transfer files between two computers connected by their serial communication ports. This can be useful when you are not able to transfer files via diskettes, perhaps because one computer has a diskette drive of a different type from the other, or does not have a diskette drive at all.

You can also use it to look at a directory of files on another computer. FILELINK must be running on both computers in order to transfer files or produce a directory listing; a useful feature of the utility is that you can install it from one computer to the other using the serial communications port.

FILELINK operates in two different modes: *master* and *slave*. When FILELINK is running in slave mode, it only acts in response to requests received from the other computer running in master mode across the serial communication line, usually to send or receive files.

Note

To avoid confusion when we discuss FILELINK, the computer you are typing a command **on** is referred to as the *master* computer, and the computer connected **to** it via the serial cable is the *slave* computer. FILELINK commands are entered at the command line on the *master* computer.

Command options

/? or /H — Displays help text.

command — FILELINK command (the valid abbreviation is shown in brackets):

DIRECTORY (DIR)
DUPLICATE (DUP)
QUIT (QUI)
RECEIVE (REC)
SETUP (SET)
SLAVE (SLA)
TRANSMIT (TRA)

[@]wildspec1 — The drive, path and name of the file to be copied. Files are renamed if a destination filename is specified. If *wildspec1* is preceded by @, it specifies the drive, path, and name of a file that contains a list of the files to be copied.

wildspec2 — The destination drive, path, and name of the file.

comparm — The communications port and baud rate to be used, in the form COM#:*baud rate*, where # is the COM port number (for example 1 or 2), and *baud rate* can be for example, 9600 or 19200 depending on your own computer.

/switches — See the separate switch description under each FILELINK command.

Before using FILELINK a suitable cable must connect your two computers, see the section "Cabling information" at the end of the FILELINK description.

Baud rates

In a serial communication link, information is transferred one bit after another. The number of bits transferred per second is referred to as the *baud* rate. Due to various standard serial transfer protocols, only 8 bits of data are transferred for every 10 bits of information. So at 9600 baud, up to 960 bytes can be transferred every second. The higher the baud rate, the quicker your files are transferred. FILELINK supports up to 115,200 baud, although not every computer can work at this speed. You may want to start working with FILELINK with a baud rate of 9600 which most computers should accept. Then when you have successfully transferred files using FILELINK, and you know that your serial ports and cable are correct, you can experiment with higher baud rates.

Whichever baud rate you use, it *must* be the same on each computer.

There are various standard baud rates available; they are listed in Table 10-10 with their valid abbreviations.

Table 10-10 Standard Baud Rate

Abbr.	baud rate
115	115, 200
57	57, 600
38	38, 400
19	19, 200
96	9, 600
48	4, 800
24	2, 400
12	1, 200
60	600
30	300
15	150
11	110

Setting the default communications port (SETUP)

You can set a default communications port to be used when FILELINK is invoked without a communications port specified in the command line. To do this use the FILELINK command SETUP in the form:

```
FILELINK SETUP comparm
```

where:

comparm is the communications port and maximum baud rate to be used.

SETUP may be abbreviated to SET.

If you do not set up the default communications port, you will have to remember to specify it each time you issue a FILELINK command.

Typing FILELINK SETUP without any parameters displays the current setup.

Note

You may find it convenient to set up the *default* communications port as the same on both computers. The baud rate *must* be the same.

Examples

The following command sets a default communications port on the *master* computer to COM1 and the maximum baud rate to 9600 baud:

```
C>FILELINK SET com1: 9600
```

The following command displays the current setup:

```
C>FILELINK setup

Current FileLink setup is for: com1:9600
```

Installing FILELINK on a slave computer (DUPLICATE)

To install FILELINK on a slave computer via a serial link, use the FILELINK command DUPLICATE in the form:

```
FILELINK DUPLICATE [comparm]
```

where:

comparm is the communications port to be used on the master computer. If it is not specified, the default communications port is used.

DUPLICATE may be abbreviated to DUP.

You are prompted to carry out various operations on the slave computer to install FILELINK.

Example

The following command copies the FILELINK program via the serial communications port to a slave computer. As no comparm is supplied, the default set by the FILELINK SETUP command is used.

```
C>FILELINK DUP
```

Putting a computer into slave mode (SLAVE)

To put a computer into slave mode, use the FILELINK command SLAVE in the form:

```
FILELINK SLAVE [comparm] [/X]
```

SLAVE may be abbreviated to SLA.

Files cannot be transferred to or from a computer until the FILELINK SLAVE command is issued on the slave computer. If you do not specify a communications port, the default communications port is used. The current directory and drive when slave mode is entered becomes the default for files sent to, or received from, the slave computer.

The valid option you can use with this command is:

Command switch	
/X	If you specify /X on the command line no files are overwritten on the slave computer.

Example

The following command puts a computer into slave mode. As no comparm is given, the default is used as set by the FILELINK SETUP command:

```
C:\JESS>FILELINK SLA
```

Note that as SLAVE mode was entered when the current drive was C: and the current directory was **jess**, these will be the default drive and directory for the slave computer.

To quit FILELINK running in slave mode, press **Ctrl+C**, see also "Quitting FILELINK on a slave computer".

Sending files to a slave computer (TRANSMIT)

To send a file or files from the master computer to the slave computer, use the FILELINK TRANSMIT command in the form:

```
FILELINK TRANSMIT [@]wildspec1 [wildspec2] [comparm]
[/A] [/D:date] [/H] [/M] [/P] [/R] [/S] [/U]
```

where:

wildspec1 is the drive, path and name of the file to be sent from the master computer.

wildspec2 is the destination filespec on the slave computer.

TRANSMIT may be abbreviated to TRA.

Before you can send files, the slave computer must be running in slave mode, see "Putting a computer into slave mode".

The valid options you can use with this command are:

Command switches

/A	Copy only files with the archive attribute.
/D:*date*	Only copy files modified since a specified date.
/H	Copy files with hidden or system attributes.
/M	Copy only those files that have their archive attribute set and either do not exist on the destination or exist but with their archive attribute reset.

Command switches (continued)

	When you specify the /M switch, you ask FILELINK to copy only those files that you have modified on the source directory, but have not modified on the destination directory. This makes use of the fact that DR DOS automatically sets the archive attribute on any file that gets created or changed. You can use this switch to regularly back up files from one computer to another, see "Examples" below.
/P	Prompt before copying each file.
/R	Overwrite read-only files on the destination.
/S	Copy files in subdirectories.
/U	Copy only files that do not exist on the destination, or that exist but are older versions, that is, have an earlier date stamp.

Examples

All the following examples assume that the FILELINK SLAVE command was issued first on the slave computer, as in the example under "Putting a computer into slave mode".

If you have several documents with the file extension **.doc**, and you occasionally alter one or two of them, when you back them up you can copy them to the slave computer using the command:

```
C>FILELINK TRA *.doc
```

However because you know that most of the files are already on the slave computer from the last time you backed up, it takes less time if you use the /M switch, then only those files you have changed since the last backup are copied; the command you use is:

```
C>FILELINK TRA *.doc /M
```

The following command copies all files in the current directory of the master computer (**c:\mary**) to the current directory of the slave computer (**c:\jess**).

```
C:\MARY>FILELINK TRANSMIT *.*
```

The following command copies all the files in the **c:\tools** subdirectory and all files in the subdirectories below **c:\tools** to the current directory on the slave computer's A: drive:

```
C>FILELINK TRA \tools a: /S
```

Getting files from a slave computer (RECEIVE)

To receive files from a slave computer (which must be running in slave mode), use the FILELINK command RECEIVE in the form:

```
FILELINK RECEIVE [@]wildspec1 [wildspec2] [comparm]
[/A] [/D:date] [/H] [/M] [/P] [/R] [/S] [/U]
```

where:

wildspec1 is the drive, path and name of the file to be received from the slave computer. If *wildspec1* is preceded by @, it specifies the drive, path, and name of a file that contains a list of the files to be received.

wildspec2 is the destination wildspec on the master computer.

RECEIVE may be abbreviated to REC.

If you do not specify a communications port, the default communications port is used as set up by the FILELINK SETUP command. The valid options you can specify are the same as those described under "Sending files to a slave computer".

Examples

Note that the first command example assumes that the FILELINK SLAVE command was issued on the slave computer as in the example under "Putting a computer into slave mode".

This command copies all files on the current drive and directory of the slave computer (**c:\jess**), onto the current drive and directory of the master computer (**c:\jane**):

```
C:\JANE>FILELINK RECEIVE *.*
```

The next command copies all files with the extension **.doc** in the current directory of the A: drive on the slave computer, to the directory **c:\letters** on the master computer:

```
C>FILELINK RECEIVE a:*.doc c:\letters
```

The following command copies all documents modified since May 1st 1990 in the directory **c:\profits** on the slave computer, to the directory **c:\sales** on the master computer:

```
C>FILELINK REC c:\profits c:\sales /D:5-1-90
```

Note in this example the date is in the US format, how *you* enter the date depends on how your computer is set up. Enter the DR DOS DATE command on its own to see what your date format should be.

Displaying a list of files on a slave computer (DIRECTORY)

To display a directory of files on a slave computer (which must be running in slave mode), use the FILELINK command DIRECTORY in the form:

```
FILELINK DIRECTORY [filespec] [comparm] [/A]
[/D:date] [/H] [/P] [/S]
```

DIRECTORY may be abbreviated to DIR.

If you do not give a communications port, the default port is used (as set up by the FILELINK SETUP command).

The valid switches are:

Command switches	
/A	Show only files with the archive attribute.
/D:*date*	Show only files modified since the specified date.
/H	Show files with hidden or system attributes.
/S	Show files in subdirectories.
/P	Pauses after each screenful of display.

Examples

The following command displays a directory of all files of the current directory of the current drive on the slave computer:

```
C>FILELINK DIR
```

The following command displays a directory of all files with the extension **.bat** in the current directory of the current drive on the slave computer:

```
C>FILELINK DIR *.bat
```

The following command displays a directory of all files with the extension **.bat** dated 1-1-90 or later, in the root directory of drive A: on the slave computer.

```
C>FILELINK DIR a:\*.bat /D:1-1-90
```

The following command displays a directory of all files with in the current directory of drive C: on the slave computer.

```
C>FILELINK DIR c:
```

Quitting FILELINK on a slave computer (QUIT)

To stop running FILELINK (in slave mode) on a slave computer, either press **Ctrl+C** at the slave computer's keyboard, or use the FILELINK command QUIT from the master computer in the form:

```
FILELINK QUIT [comparm]
```

where:

comparm is the communications port on the master computer, if one is not given the default is used as set up by the FILELINK SETUP command.

QUIT may be abbreviated to QUI.

Example

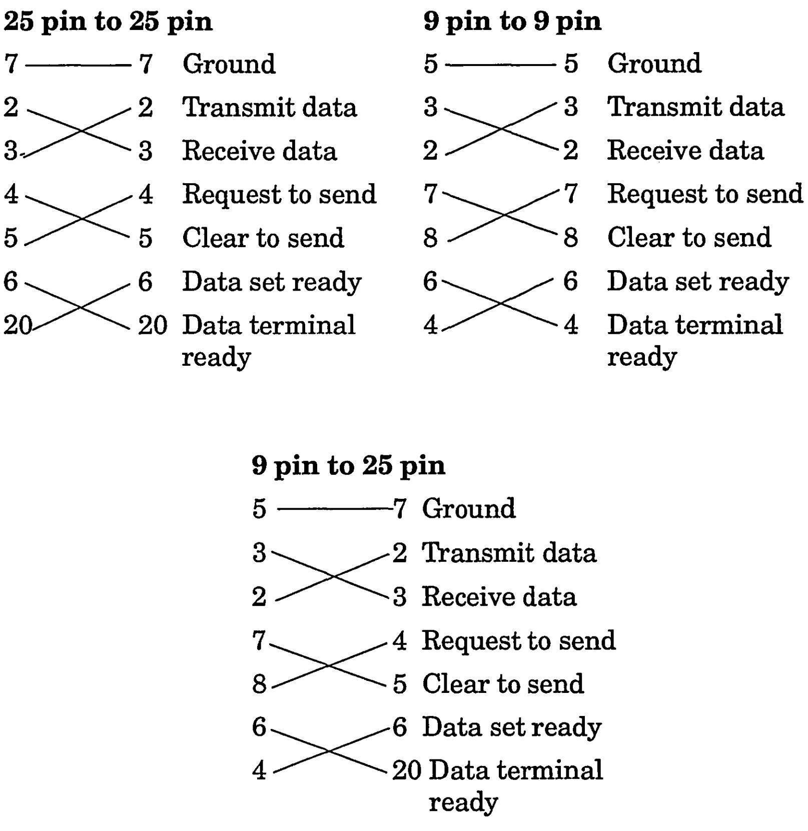
The following command stops FILELINK running on a slave computer via the communications port COM2:

```
C>FILELINK QUI COM2
```

Cabling information

The type of cable you need for FILELINK is an RS232 Null Modem cable. The cable must have connectors that are appropriate to the computers to be linked. Normally your computers will have either 9-pin, or 25-pin D-type male connectors for their serial ports. Therefore the cable to connect them must have 9 or 25-pin (as appropriate) D-type female connectors, see Table 10-11.

Table 10-11 Serial Cable Pin Connections



25 pin to 25 pin

7	7	Ground
2	2	Transmit data
3	3	Receive data
4	4	Request to send
5	5	Clear to send
6	6	Data set ready
20	20	Data terminal ready

9 pin to 9 pin

5	5	Ground
3	3	Transmit data
2	2	Receive data
7	7	Request to send
8	8	Clear to send
6	6	Data set ready
4	4	Data terminal ready

9 pin to 25 pin

5	7	Ground
3	2	Transmit data
2	3	Receive data
7	4	Request to send
8	5	Clear to send
6	6	Data set ready
4	20	Data terminal ready

FIND

Format

```
FIND [/?|/H][/B][/C][/F][/N][/S][/U][/V] "search
string" [@]wildspec...
```

Explanation

FIND searches for a string of characters in a group of text files and displays the lines containing the *search* string. The *search string* must be enclosed in double quotes (""). If the string includes double quotes, you must include an extra pair (that is ""find this"").

Multiple files may be specified on the command line using the wildcard characters ? and *.

If no files are specified FIND reads from standard input (usually the keyboard or a redirected file).

Command switches	
/? or /H	Displays help text.
/B	Changes display format of the list that is produced, so it shows a series of headings with the matches listed beneath.
/C	Shows only the total number of lines that contained the string.
/F	Shows only the names of those files that contain the string.
/N	Displays line numbers.
/S	Searches files in subdirectories that are derived from the current directory.
/U	Case-sensitive search (for example "A" doesn't match "a").
/V	Displays the lines *not* containing the string.

Examples

The following command calculates the number of lines containing the word *sorry* in each of the three files **apology1.let**, **apology2.let** and **apology3.let**:

```
A> FIND /C "sorry" apology1.let apology2.let apology3.let
```

The following command displays all the subdirectories listed in the current directory. FIND acts as a filter, receiving input from the DIR command and displaying as output only those files that contain the string:

```
A>DIR | FIND "<DIR>"
```

FORMAT

Format

```
FORMAT [/?|/H] d: [/T:tracks /N:sectors] [/F:size]
[/1] [/4] [/8] [/A] [/S] [/U] [/V:[label]] [/X]
```

Explanation

The FORMAT command formats disks and diskettes so that you can store information on them.

FORMAT allows you to format a disk or diskette in a default or customized format. The default format for a diskette is the maximum capacity of the drive in which you format it; Table 10.12 lists the default formats for various types of diskette.

Standard formats

Table 10.12 Standard Formats For Diskettes

Diskette size	Tracks	Sectors	Sides	Capacity
5¼"	40	8	1	160 Kbytes
5¼"	40	9	1	180 Kbytes
5¼"	40	8	2	320 Kbytes
5¼"	40	9	2	360 Kbytes
5¼"	80	15	2	1.20 Mbytes
3½"	80	9	2	720 Kbytes
3½"	80	18	2	1.44 Mbytes
3½"	80	36	2	2.88 Mbytes

If you use the /X switch with FORMAT to specify that you are formatting a hard disk, you will be prompted:

```
WARNING ALL existing data on non-removable disk will
be destroyed! -- Continue (Y/N)?
```

Safe format

FORMAT automatically detects the existing format of a disk or diskette. If the existing format is the same as the format you request in the FORMAT command, a safe format is performed. This means that if by accident you format a disk or diskette with data on it, you can use the UNFORMAT command to recover it. During a safe format, a copy of the system information is stored on an unused area of the disk. The UNFORMAT command can locate the system information and use it to restore files from the disk. If there is not enough space on the disk for the safe format information to be recorded without overwriting data, you are prompted:

```
Insufficient space to save UNFORMAT information
without destroying some data -- Continue (Y/N)?
```

If you type Y for yes, you may not be able to recover all the data from the disk using UNFORMAT. Type N, and you are returned to the DR DOS command prompt.

WARNING If you are not formatting a disk of the same type into the same format, all existing data on the disk will be wiped clean. You will not be able to recover any data using UNFORMAT.

Customized formats

You can use FORMAT on any device with a driver that supports the FORMAT operation. For example, you can use it with a drive on a SCSI adaptor but not with a memory disk. You specify customized formats by using the FORMAT switches /T and /N to indicate the number of tracks and sectors per track that you require (see the command switches table below). Note that the device driver for the device must be able to support the values specified.

Command switches

/? or /H Displays help text.

/T:*tracks* Specifies the number of tracks to be formatted. If you are specifying a non-standard format, you must also use /N.

/N:*sectors* Specifies the number of sectors to be formatted. If you are specifying a non-standard format, you must also use /T.

/F:*size* Specifies the capacity of the diskette to be formatted. Use this switch when the format required is other than the default for the drive. Note that it can only be used for formatting diskettes. Values can be:

Command switches (continued)

For 5¼" diskettes	For 3½" diskettes
160 (160 Kbytes)	720 (720 Kbytes)
180 (180 Kbytes)	1.44 (1.44 Mbytes)
320 (320 Kbytes)	2.88 (2.88 Mbytes)
360 (360 Kbytes)	
1.20 (1.2 Mbytes)	

/1 Formats one side only of a 320 Kbytes or 360 Kbytes diskette (only on 5¼" diskettes). If you enter /F:360 followed by /1, the format will be 180 Kbytes. If you type /F:320 followed by /1 the format will be 160 Kbytes. On 1.2 Mbyte drives, use /F:320 or /F:360 before /1.

/4 Forces 360 Kbytes format unless used with the /1 switch where the format will be 180 Kbytes. Use this switch to format 360 Kbytes format diskettes in a 1.2 Mbyte format drive. Note that diskettes formatted in this way cannot be reliably read by 360 Kbytes drives.

/8 Formats 9 sectors but uses only 8. This switch can only be used on its own with 360 Kbytes 5¼" diskettes. If you specify /F:360 followed by /8, the format will be 320 Kbytes. The default is to format 9 sectors and use all 9. On 1.2 Mbyte drives, use /F:320 or /F:360 first.

/A Causes the computer to sound a "beep" when the format operation is complete.

/S Copies DR DOS system files onto the formatted disk. DR DOS first looks for the system files on the drive from which FORMAT was loaded and uses COMSPEC to find **command.com**. If the system files cannot be found, DR DOS prompts you to insert a diskette containing them into drive A:. After reading the system files, DR DOS prompts you to remove the diskette and insert the diskette to be formatted.

Command switches (continued)

/U

Forces FORMAT to perform a full format of the disk. This means that the disk surface will be wiped clean of any data found there.

WARNING If you specify the /U switch, safe format information is not stored on the disk and you will not be able to use the UNFORMAT command to recover data from the disk.

/V:[*label*]

Allows you to specify the volume label for the disk. You can type a volume label of up to 11 characters after /V: /V:admin, for example. If you type /V on its own, you are not prompted for a label during the format operation.

/X

Specifies that you are formatting a fixed disk. Remember *not* to use /X when you are formatting a diskette.

Remarks

FORMAT does not recognize drive assignments made with the ASSIGN, JOIN or SUBST commands.

You cannot format drives over a network.

Volume labels you have added to disks can be amended or deleted using the LABEL command. You can display a disk's current volume label using the VOL command.

Examples

To format a diskette:

1. On a hard disk system with one diskette-drive, insert a blank diskette into drive A: and type:

```
C>FORMAT a:
```

```
Insert target diskette in drive A:
and press ENTER when ready...
```

2 Press **Enter**. After a short time you are prompted:

```
Enter volume label (Max. 11 characters, ENTER for none)
```

if you want to add a label to the disk, type it and press **Enter**. Otherwise, press **Enter** only.

3 You see a message similar to this:

```
Disk formatted successfully

    nnnnnn bytes total disk space.
    nnnnnn bytes available on disk.

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

Type Y for yes if you want to format another disk, or N to return to the DR DOS command prompt.

On a twin-diskette system, with your system diskette in drive A:, type:

```
A>FORMAT b:
```

To format a diskette at 360 Kbytes with the diskette in a 1.2 Mbyte drive, specifying the label on the disk as **archive** and sounding a beep when formatting is complete, type:

```
A>FORMAT a: /F:360 /V:archive /A
```

GRAFTABL

Format

```
GRAFTABL [/?|/H][nnn][/STATUS]
```

Explanation

GRAFTABL enables you to display extra and international characters (ASCII characters 128 through 255) when you have a color/graphics (CGA) adaptor fitted to your computer. GRAFTABL only needs to be loaded once during each DR DOS session, so you may want to include it in your **autoexec.bat** file (you can use the SETUP program to do this).

Command switches

/? or /H — Displays help text.

nnn — Specifies the number of one of the following character sets:

437 USA (the default if no code is supplied)
865 Norway
860 Portugal
863 French Canadian
850 Multilingual

/STATUS — Displays the currently selected country code page.

Remarks

GRAFTABL adds to the amount of space in memory taken up by DR DOS.

Example

When you type:

C>**GRAFTABL**

you see a message like this:

```
    Character table for USA has been loaded.
```

GRAPHICS

Format

```
GRAPHICS [/?|/H][COLOR][/R]
```

Explanation

GRAPHICS allows you to use the **PrtSc** key to print a graphics display onto an IBM-compatible graphics printer. GRAPHICS only needs to be loaded once during a DR DOS session, so you may like to include it in your **autoexec.bat** file.

Command switches

/? or /H	Displays help text.
COLOR	Enables graphics to produce color output on an IBM-compatible color graphics printer, producing eight colors.
/R	Prints black and white exactly as they appear on your display. The default is to print what is white on your screen as black on the printer.

Remarks

GRAPHICS increases the amount of memory space occupied by DR DOS.

HILOAD

Format

```
HILOAD [/?|/H] filespec
```

Explanation

The HILOAD command loads specified programs into upper memory. This leaves more space in conventional memory for your other applications and data. If there is insufficient upper memory for the specified program, it is loaded into conventional memory. Before you can use HILOAD, a device driver that supports upper memory must be loaded. See section 11.4, "Installable DR DOS device drivers" and Chapter 12, "Using MemoryMAX" for descriptions of device drivers.

HILOAD is useful for loading network drivers and TSRs. Generally you should load any TSRs into upper memory. Not all applications are suitable for loading into upper memory though, because they require more than the limited amount of upper memory. You should consider loading the following DR DOS programs into upper memory:

CURSOR .EXE
GRAPHICS.COM
GRAFTABL.COM
JOIN.EXE

HILOAD performs the same function as HIINSTALL, but loads programs after **command.com** has been loaded. You can include HILOAD in your **autoexec.bat** file so that TSRs can be loaded into upper memory at startup.

Notes

There is no need to HILOAD the following DR DOS programs: KEYB.COM, SHARE.EXE, and NLSFUNC.EXE.
Some applications are sensitive to the additional areas of RAM provided by MemoryMAX (see Chapter 12, "Using MemoryMAX"). DR DOS therefore automatically disables upper memory by adding the line MEMMAX -U>NUL to **autoexec.bat**. Before you use HILOAD, you must enable upper memory with the command MEMMAX +U.

You cannot use HILOAD if a third-party memory manager has been loaded, because these memory managers tend to use a different technique to DR DOS for mapping upper memory.

Command switches	
/? or /H	Displays help text.

Examples

The following command loads a network driver into upper memory, assuming you have already enabled upper memory with MEMMAX +U:

```
C>HILOAD C:\LAN\NET3
```

JOIN

Format

```
JOIN [/?|/H] [drive1:[drive2:dirpath]] [/D]
```

Explanation

Using JOIN you can attach the directory structure of one disk drive to an empty subdirectory on another disk drive. Doing this makes the directory structure of the first drive, *drive 1:*, (and all its files) appear to be a subdirectory of the directory you specified on the second drive. Connecting drives using JOIN means you can use commands like TREE and XDIR on joined drives as if they were one large drive.

Command switches and syntax

/? or /H	Displays help text.
drive1	The drive to be joined to the subdirectory of another drive.
drive2:dirpath	The drive and directory used to refer to files on drive 1.
/D	Cancels a JOIN command.

If you give a path on the second drive that does not exist, DR DOS tries to make a directory with that path. If the directory at the end of the path on the second drive already exists, it must be empty. After you have issued a JOIN command, the first drive is no longer available to you and if you try to access it you get an error message.

You can only connect drives using JOIN to a directory leading from the root directory. Note that all subdirectories below the joined directory are not accessible until you disconnect them.

To cancel an existing JOIN command, type JOIN and the first drive followed by the /D switch. Always remember to do this when you have finished working.

To see a list of joined drives, type JOIN on its own.

Caution You must not use the following commands on drives connected with JOIN:

ASSIGN
BACKUP
DISKCOPY
DISKCOMP
FORMAT
RESTORE
SUBST

Also drives connected using JOIN do not work on networked drives.

Examples

This command connects drive A: to subdirectory **\clients**:

```
C>JOIN a: c:\clients
```

Now drive A: can be accessed via **c:\clients**. If you try to access drive A: directly you get this error message:

```
    Invalid drive
```

The following command displays the connected drive:

```
C>JOIN

A: => C:\CLIENTS
```

The following command cancels a JOIN command:

```
C>JOIN a: /D
```

KEYB

Format

```
KEYB [/?|/H] xx[+|-][,codepage] [/ML|/MH|/MU]
```

Explanation

This command loads the keyboard program which makes characters displayed on the screen match those typed at the keyboard. You need to use KEYB each time you start DR DOS if you do not have the default (US) keyboard. If the KEYB command is part of the **autoexec.bat** file, it is loaded automatically each time DR DOS is started. This is set up for you when you install DR DOS.

Command switches and syntax	
/? or /H	Displays help text.
xx	A two-letter country code, as shown in Table 10-13.
+	Forces the keyboard program to assume an enhanced (101/102/104 key) keyboard. KEYB usually recognizes enhanced keyboards automatically. However, if KEYB does not recognize your enhanced keyboard, and you find that certain keys do not match the characters displayed, try re-issuing the KEYB command with the + option.
-	Forces the keyboard program to assume a standard (83/84 key) keyboard.
,codepage	Specifies the code page that you want to use. If this is missing, the country's default code page is selected. It allows the accented characters defined in the selected code page to be produced. Some characters can only be entered using "dead" key combinations. (Dead keys are keys that do not display a character when used on their own, but if followed by a letter, display that letter with an accent.) To produce the character, first type the accent and then press the alphabetical key.

Command switches and syntax (continued)

	To display the accent on its own, type the accent and then the space bar. For more information on which accented characters are allowed in which code page, see Appendix B, "Character sets for code pages supported by DR DOS".
/ML I /MH I /MU	Forces the KEYB program to be loaded into a specific area of memory; /ML for conventional (lower) memory, /MH for high memory, or /MU for upper memory. Note that if there is insufficient high or upper memory, KEYB is loaded into conventional memory. Before you can use /MH or /MU, a device driver that supports high and upper memory must be loaded.

Table 10-13 Keyboard Codes and Code Pages

Country	Abbreviation	Code pages
Belgium	BE	437 850
Canada (French)	CF	863 850
Denmark	DK	865 850
Finland	SU	437 850
France	FR	437 850
Germany	GR	437 850
Hungary	HU	852 850
Italy	IT	437 850
Latin America	LA	437 850
Netherlands	NL	437 850
Norway	NO	865 850
Portugal	PO	860 850
Russia	RU	866 850
Spain	SP	437 850

Table 10-13 Keyboard Codes and Code Pages (continued)

Sweden	SV	437 850
Swiss (French)	SF	437 850
Swiss (German)	SG	437 850
Turkey	TF/TQ	857 850
United Kingdom	UK	437 850
United States*	US	437 850

* default

Remarks

If you wish to change keyboards while the program is running, instead of re-issuing the KEYB command you can switch back to the default (USA) keyboard by holding down the **Ctrl** and **Alt** keys and pressing **F1**. To go back to your chosen national keyboard, type **Ctrl+Alt** and then **F2**. Note: If you are using the Russian character set, the right-hand **Ctrl** key toggles between native and US keyboard layouts.

For more information on code page switching see Chapter 14, "Setting up code page switching".

Examples

The following command loads the United Kingdom keyboard.

```
A>KEYB uk
```

The following command loads the Spanish keyboard and ensures that KEYB recognizes the keyboard as being an enhanced keyboard. It also ensures that codes for use with code page 850 are generated.

```
A>KEYB sp+,850
```

This command specifies a Norwegian enhanced keyboard, and the national code page table 865.

```
A>KEYB no+,865
```

LABEL

Format

```
LABEL [/?|/H] [d:][label]
```

Explanation

The LABEL command allows you to create, modify or delete the volume label on a DR DOS disk or diskette. A volume label is a name given to a disk to identify it and is displayed when you perform a DIR or CHKDSK command. Note you can also use the VOL command to find out if a disk or diskette has a volume label.

Adding a label

To create a volume label, use the form:

```
LABEL label
```

where *label* is a string of up to 11 characters. You can also use spaces, but not tabs. As with naming files and directories, do not include the following characters:

<> = , ; : . * ? [ ] ( ) / \+

If no drive is specified in the command line, the default drive is used. You are prompted to type a label if you do not specify one at the command line.

LABEL does not work with drives affected by a SUBST or JOIN command. You cannot use LABEL on a networked drive.

WARNING When you enter a new label it overwrites an existing label without giving any warning.

Deleting a label

To delete a label, type LABEL and the disk drive where you want the label deleted from.

Command switches	
/? or /H	Displays help text.

Examples

This command assigns the name **workdisk** to the disk in drive B:

```
A>LABEL b:workdisk
```

To delete a volume label on drive B: type:

```
A>LABEL b:
```

you get a message similar to this:

```
    Volume in drive B: is WORKDISK
    Enter Volume label (0 to 11 characters):
```

Press **Enter** and you see this message:

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

Type Y to delete the label.

LOCK

Format

```
LOCK [/?|/H] [password]
```

Explanation

Use LOCK to temporarily lock your operating system so that you can leave the computer unattended without exiting your applications and switching the computer off. LOCK provides additional security to the DR DOS system security you can install during INSTALL or SETUP.

When you enter the LOCK command, you see a screen similar to the system security login screen. This screen prompts you for a LOCK password.

Command switches

/? or /H	Displays help text.
password	Specifies the password required to unlock the system, if you want to use a password different from the password you use to login to the system with system security (the User Key and/or Master Key password), you can specify any password up to 12 characters in length.

There are two different ways of using LOCK:

- ❑ You can run LOCK from the command line. As long as you installed system security during INSTALL or SETUP, you can type LOCK without a password parameter to use your existing login password to unlock the system. Whether you have system security or not, you can type LOCK with a new password to use that password. Note that once you have specified a different LOCK password, you will not be able to use your system login password, if you have one.

  For information about installing system security and specifying User Key and Master Key passwords, refer to Chapter 4, "Protecting your information".

❑ You can load LOCK as a task in TaskMAX. LOCK is automatically loaded as a task if you install both system security and TaskMAX during INSTALL. If you did not select both, load LOCK as a task using the TaskMAX /C command switch.

Refer to Chapter 6, "Switching tasks", for more information about TaskMAX and its switches.

Note

Using LOCK in TaskMAX is the most flexible way because you can stop what you are doing at any time (without having to exit the application) and lock the system.

Remarks

You can edit the LOCK screen directly if you want to change its contents and layout. The screen is contained in the file called **lock.txt**, in directory **c:\drdos**.

Examples

The following command displays the LOCK screen and locks the operating system until you type in `mylock` as the password:

```
C>LOCK mylock
```

The next command, issued from within TaskMAX, loads LOCK as a TaskMAX command:

```
C>TASKMAX /C c:\drdos\lock.exe
```

In TaskMAX, you can then lock the system by pressing **Ctrl+Esc** to display the TaskMAX menu and select the LOCK task. When you unlock the system, the TaskMAX menu is displayed again and you simply select the task you were working in before you used LOCK.

MEM

Format

```
MEM [/?|/H][/B][/D][/S][/P][/M][/A]
```

Explanation

The MEM command displays a range of information on the way memory is used. If you type MEM on its own, you get a report listing how much conventional and extended memory (if your computer has memory above 1 Mbyte), your computer has allocated.

The various switches cause additional information to be displayed. Most users will find the /B switch useful as it explains, amongst other things, where programs are located in memory. The remaining options are primarily for advanced users, or system programmers.

Command switches

/? or /H	Displays help text.
/B	Displays areas of memory used by different parts of the operating system, such as device drivers and BIOS, also the DOS Memory Control Block (MCB) chain which shows how memory is currently allocated to programs.
/D	Shows the names of all built-in, and loadable device drivers, and their location in memory.
/S	Displays the DOS disk buffer chain. For conciseness, adjacent buffers are grouped together.
/P	Pauses at every screenful of information.
/M	Displays graphically the location of RAM, ROM, and EMS memory. This can be useful when used to fine tune the installable device driver EMM386.SYS, see also Chapter 11, "Customizing your system".
/A	Shows all information (a combination of all the above switches).

Examples

The following command displays a report on conventional and extended memory only.

C>**MEM**

You see a display similar to this:

```
  655,360  bytes, ( 640K), conventional memory
  630,080  bytes, ( 615K), largest available block


4,194,304  bytes  (4096K), extended memory
4,194,304  bytes  (4096K), extended memory used
        0  bytes  (   0K), extended memory
                           available
```

MEMMAX

Format

```
MEMMAX [/?|/H] [-|+L] [-|+U] [-|+V] [/U] [/L] [/V]
```

Explanation

The MEMMAX command selectively disables those enhanced memory features provided by DR DOS that may cause conflicts with some applications. If you type MEMMAX on its own, it displays the current status of the enhanced memory features.

The HIDOS.SYS and EMM386.SYS device drivers, described in Chapters 11, "Customizing your system" and 14, "Setting up code page switching", can increase the memory available to applications using upper memory. However, some programs fail because they do not expect to find memory in this region. MEMMAX allows you to selectively enable and disable upper memory from the command line. MEMMAX is also used by DR DOS INSTALL, which automatically appends the command MEMMAX -U to **autoexec.bat**, so preventing applications from using upper memory; you must execute MEMMAX +U to re-enable upper memory if you want to use the HILOAD command.

If the operating system kernel has been loaded into high memory, lower memory becomes available for applications. Under certain circumstances, some applications will fail to load in lower memory. Typically, applications that fail will terminate with the message `!Packed file is corrupt`. By using MEMMAX -L, you can force the application to load successfully above this lower 64Kb region. Once the application has finished, you can reinstate use of this lower 64Kb region with the MEMMAX +L command.

Command switches

/? or /H	Displays help text.
+L	Enables access to lower memory.
-L	Disables lower memory.
+U	Enables access to upper memory.
-U	Disables upper memory (the default state set by INSTALL or SETUP).

Command switches (continued)

+V	Enables memory reserved by the /VIDEO option of EMM386.SYS or HIDOS.SYS.
-V	Disables memory reserved by the /VIDEO option of EMM386.SYS or HIDOS.SYS.
/L	Displays the status of lower memory.
/U	Displays the status of upper memory.
/V	Displays the status of VIDEO memory.

Examples

The following example shows how you can use the MEMMAX command in a batch file that loads network drivers into upper memory. The example assumes that you have already loaded a device driver that supports upper memory.

```
REM****Use MEMMAX to open upper memory****
MEMMAX +U
HILOAD C:\LAN\IPX
HILOAD C:\LAN\NET3
REM****Disable lower memory - it confuses LOGIN****
MEMMAX -L
I:
CD LOGIN
REM****Reopen lower memory****
MEMMAX +L
```

MKDIR (MD)

Format

```
MKDIR | MD [/?|/H] [d:]dirpath
```

Explanation

Use MKDIR to create subdirectories in the root directory or other subdirectories. You can abbreviate MKDIR to MD.

If you do not include a backslash (\) before the first subdirectory name in the directory path, MKDIR begins the path to your new directory from the current directory of the current or specified drive. If you do include the backslash, MKDIR begins the path at the root directory of the current or specified drive. For example if you are in the root directory and you want to create the subdirectory **hank** in the subdirectory **sales**, type:

```
A>MKDIR \sales\hank
```

If you are currently in the **sales** subdirectory and want to create the subdirectory **hank** from **sales**, type:

```
A>MKDIR hank
```

The path from the current directory leading to the new subdirectory must be entered and all the subdirectories in the path must exist. If a subdirectory is specified that does not exist, MKDIR displays the following message:

```
    Path not found
```

The maximum number of characters in a path to directory (including backslashes) is 63.

When a directory is created, DR DOS automatically adds two entries . and .. which it uses as markers to locate and identify the directory; these entries cannot be deleted. Files created in a directory must be unique within it, but you can create files with the same name in another directory.

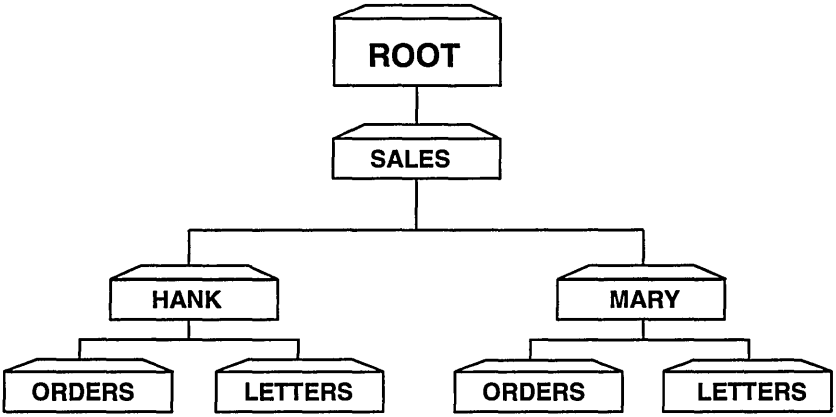
Command switches	
/? or /H	Displays help text.

Example

Use subdirectories to organize your data and command files into logical groups. The sequence of MKDIR commands shown below creates the hierarchical series of subdirectories shown in the chart:

```
C>MKDIR \sales

C>MKDIR\sales\hank

C>MKDIR\sales\mary

C>MKDIR\sales\mary\orders

C>MKDIR\sales\mary\letters

C>MKDIR\sales\hank\orders

C>MKDIR\sales\hank\letters
```

You can have subdirectories with the same name, but they must have different paths.



MODE

The MODE command, in general, changes the configuration of various devices attached to your system. MODE has several versions, each of which is explained separately.

Format

Command switches	
/? or /H	Displays help text.

The different versions of the command are as follows:

- ❑ For setting the display type and keyboard typematic rate:

```
MODE displaymode [,lines]

MODE [displaymode],m[,T]

MODE CON[:][LINES=lines][COL=cols][RATE=r DELAY=d]
```

- ❑ For assigning a printer:

```
MODE LPT#[:]=COM#
```

- ❑ For setting up a printer:

```
MODE LPT#[:][n][,[m][,P]]
```

- ❑ For setting up a communications port:

```
MODE COM#[:]baudrate[,[parity],[databits],
[stopbits][,P]]
```

- ❑ For setting up and using code pages:

```
MODE device CODEPAGE PREPARE = ((cplist) filespec)

MODE device CODEPAGE SELECT = cp

MODE device CODEPAGE [/STATUS]

MODE device CODEPAGE REFRESH
```

Code page switching

A *code page* is a table that defines a particular character set required by a national language. Code page *switching* allows you to use the different character sets required by different national languages. You can use the CODEPAGE versions of the MODE command to prepare, select, display, and refresh code pages. Read Chapter 14, "Setting up code page switching" for more information about when you need to set your system up for code page switching, how you do it, and what you can do with code pages.

Setting the display type and keyboard typematic rate

Using MODE you can select the display type for your monitor and change the current display. This can involve changing between 40 and 80 columns on the screen, or between monochrome and color. You can change the number of lines on the screen or center the display on the screen. You can also alter your keyboard typematic rate (which is how quickly characters are echoed to the screen when a key is held down).

Setting the display type

Format

```
MODE displaymode [,lines]
```

Explanation

This format of MODE allows you to set the display type for a monitor. You might use this command when you have fitted a graphics adaptor and an alternative monitor onto your system, and you want DR DOS to work with the new monitor.

The display types that can be selected are listed below. The mode you have selected remains until you type another MODE command or restart DR DOS.

Option	**Explanation**
CO40	Enables the graphics adaptor and sets the display width to 40 characters per line with a color display.
CO80	Enables the graphics adaptor and sets the display width to 80 characters per line with a color display.

Option	Explanation (continued)
BW40	Enables the graphics adaptor, sets the display width to 40 characters per line, and the display mode to black and white.
BW80	Enables the graphics adaptor, sets the display width to 80 characters per line, and the display mode to black and white.

If the graphics adaptor is already enabled you can use the following commands:

Option	Explanation
40	Sets the display width to 40 characters per line.
80	Sets the display width to 80 characters per line.
MONO	Switches the active display to the original monochrome display (if you have one fitted). This display always has a width of 80 characters per line.
,lines	Sets the number of text rows (25, 43 or 50) on the screen.

Examples

This command enables the graphics adaptor, sets the display width to 80 characters per line, and the display mode to black and white:

```
C>MODE BW80
```

This command selects 80 columns by 43 lines:

```
C>MODE 80,43
```

On mono VGA, mode CO80 is often used to make an application believe that it has a color screen, or conversely is in mono mode. The following command when used on a system with a mono VGA screen, will select mono mode or color mode (with colors shown as gray scales):

```
C>MODE CO80
```

Aligning the display on a CGA monitor

CGA or Color Graphics Monitors do not always display information properly. The MODE command can be used to center data so it is more readable, the format is:

```
MODE [displaymode],m[,T]
```

where:

Option	Explanation
displaymode	Sets the type of display you want to specify, see the list on the previous page.
,m	Shifts the display right or left; enter R for right, L for left.
,T	Causes a test pattern to be displayed so you can see how the screen will be aligned. You will be asked if the alignment is acceptable. If you answer N, the display will be shifted again and you can continue this procedure until the display is correctly positioned.

Example

This command shifts the display one column to the right and then asks you if you can see all of the test pattern.

```
MODE ,R ,T
```

Setting lines, columns and the keyboard typematic rate

To set the number of columns and lines on a display or the keyboard typematic rate (which is how quickly characters are echoed to the screen when a key is held down), use the MODE command in the form:

```
MODE CON[:][LINES=lines][COL=cols][RATE=r DELAY=d]
```

where:

Option	Explanation
lines	Indicates the number of lines on the display.
cols	Represents the number of columns on the display.
r	Sets the typematic interval time (that is the speed at which characters are repeated on the screen), in the range 1-32 where 1 is the *slowest* repeat.
d	Specifies the auto-repeat start-delay time with the possible values of 1-4 (that is if you press and hold down the F key on your keyboard, auto-repeat start-delay is how long it is before F is repeated). Note that 1 is the *shortest* delay.

Note

To set the typematic rate you must specify the rate and delay at the same time.

Examples

This command selects 80 columns by 43 lines on the display.

```
C>MODE CON: COL=80 LINES=43
```

This command sets the keyboard typematic rate to 32 (the shortest rate permitted), and the auto-repeat start-delay to 1 (the shortest delay allowed).

```
C>MODE CON: RATE=32 DELAY=1
```

Assigning a printer

Format

```
MODE LPT#[:]=COM#
```

Explanation

With this form of MODE you can alter the standard assignments for your printer ports.

DR DOS lets you connect up to three printers to your computer. Usually information is sent to these printers through your computer's parallel ports, commonly labeled LPT1, LPT2, and LPT3. The MODE command enables you to send information to a serial printer through one of your computer's serial ports instead. (The serial ports are sometimes called COM ports or communications ports and are labeled COM1 and COM2). For example if you have a program (such as a word processor) that normally sends output to port LPT1, you can use MODE to redirect information to a serial port like COM1 instead.

The hash symbol # represents the number of the port, for example LPT# could be replaced with LPT1. The colon between LPT# and =COM# is optional.

To return to the default settings, type the LTP#: without the COM#.

Examples

To redirect output from LPT1 to COM1, type the following:

```
C>MODE LPT1:=COM1
```

This causes a small part of MODE to be transferred into system memory and you see the message:

```
    MODE: Resident portion installed
```

To return to the default settings type:

```
C>MODE LPT1:
```

Setting up a printer

Format

```
MODE LPT#[:][n][,[m][,P]
```

Explanation

Use this format of MODE to tell DR DOS what kind of printer is connected to a particular port. Your printer documentation should help indicate the values you need (for example for *n*).

Options	Explanation
LPT#:	The printer number, where # can usually be 1, 2, or 3.
n	The printer characters per line: 80 or 132. The default is 80. (80 character printers are often known as "narrow carriage", and 132 character printers as "wide carriage".)
m	The lines per inch for the printer: 6 or 8 (8 is the default).
,P	Tells DR DOS to keep on trying to send information to a printer that is not printing. Without ,P DR DOS will try for only a short length of time before issuing a timeout error message. When you issue MODE with ,P a small part of the MODE program is transferred into system memory and you see the message: `MODE: Resident portion installed` If you select ,P and then find that you want to stop DR DOS continuously trying to send information, press **Ctrl+Break**. To remove the ,P option altogether, type the MODE LPT# command without the ,P.

Examples

This command instructs DR DOS to continue trying to send information to a wide printer at printer port LPT2.

```
C>MODE LPT2:132,,P
```

The following command returns to the default settings on the same port:

```
C>MODE LPT2
```

Setting up a communications port

Format

```
MODE COM#[:]baudrate[,[parity],[databits],
[stopbits][,P]]
```

Explanation

This form of the mode command enables you to set communications parameters for your serial (or communication) ports, for example COM1 and COM2.

The command allows you to set baud rate, parity, data bits, and stop bits, and allows timeout errors to be retried continuously.

Usually you will find that you only need to set the baud rate; the default values for the other options may already be set up for your needs.

Options	**Explanation**
baud rate	Can be 110, 150, 300, 600, 1200, 2400, 4800 and 9600, (also 19200 on some computers such as IBM PS/2™). You only need to type the first two digits of your selection.
parity	Can be N (none), O (odd) or E (even, which is the default).
databits	Can be 7 or 8 (7 is the default).
stopbits	Can be 1 (the normal default) or 2 (the default when the baud rate is set at 110).
,P	Tells DR DOS to keep on trying to send information to a device which is not responding. Without ,P DR DOS tries for only a short length of time before issuing a timeout error message. When you issue MODE with ,P a small part of the MODE program is transferred into system memory and you see the message:

```
Infinite retries on serial port COMn:
MODE: Resident portion installed
```

Option	Explanation (continued)
	If you select ,P and then find that you want to stop DR DOS continuously trying to send information, press **Ctrl+Break**. To remove the ,P option altogether, type the MODE COM*n* command without the ,P.

Note

Always separate the different options in the command line with commas. If options are not required, the correct number of commas must still be inserted.

Examples

The following command sets the communications parameters (sometimes called protocol) on COM1 to 9600 baud. The defaults for the other values will be: even parity, 7 databits, 1 stopbit.

```
C>MODE COM1:96
```

The following command changes the protocol for the above command to odd parity:

```
C>MODE COM1:96,o
```

The following command sets the protocol on COM2 at 4800 baud and 8 databits, and specifies continuous retry:

```
C>MODE COM2:48,,8,,P
```

Preparing code pages

Format

```
MODE device CODEPAGE PREPARE = ((cplist) filespec)
```

Explanation

Use this form of the MODE command to prepare code pages.

All parameters are required.

You see the following message when the command completes:

```
Codepage prepare function completed
```

Ensure that all printers affected by the MODE command are switched on, and online. Otherwise, you get an error message that may not appear immediately.

Options	Explanation
device	Specify CON, PRN, or LPT*n* (where *n* is 1, 2, or 3). Note that PRN is equivalent to LPT1.
CODEPAGE	Specify CODEPAGE or CP.
PREPARE	Specify PREPARE or PREP.
cplist	Specify a list of code pages to prepare. Separate the numbers with commas if you are specifying more than one code page, and enclose the list in brackets as shown in the command format. You can leave a code page unspecified (so that it is unchanged) by using a comma without a preceding number. The following command, for example, leaves the first code page unchanged and prepares 850 as a second code page: `MODE lpt1 CP PREP = ((,850)\4201.cpi)` Supported code pages are 437, 850, 852, 857, 860, 862, 863, 865, and 866. Refer to Chapter 14, "Setting up code page switching" for more information about selecting code pages.
filespec	Specify the location of the file containing the character shapes (the font file) for the code pages. DR DOS supplies the following files:

Option	Explanation (continued)
Font file	**Device**
ega.cpi	EGA or VGA display devices
4201.cpi	IBM Proprinter and XL printers
4208.cpi	IBM Proprinter X24 and XL24 printer
5202.cpi	IBM Quietwriter III printer
1050.cpi	Epson FX 850 and FX 1050 printers.

Examples

The following command prepares Code Page 850 for the CON display device using the fonts file in drive A:

```
C>MODE con CP PREP = ((850) a:\ega.cpi)
```

The next command prepares the first and third code page in a list for the printer on LPT1, using the font file in directory **drdos** on the hard disk:

```
C>MODE lpt1 CP PREP = ((850,,863) c:\drdos\4201.cpi)
```

Selecting code pages

Format

```
MODE device CODEPAGE SELECT = cp
```

Explanation

The MODE PREPARE command only *prepares* code pages. The MODE SELECT version of the command specifies which prepared code page is active on a device. Only one code page per device can be active at any time.

All parameters are required.

You see the following message when the command completes:

```
     Completed codepage select function
```

Ensure that all printers affected by the MODE command are switched on, and online. Otherwise, you get an error message that may not appear immediately.

Option	**Explanation**
device	Specify CON, PRN, or LPT*n* (where *n* is 1, 2, or 3. Note that PRN is equivalent to LPT1.
CODEPAGE	Specify CODEPAGE or CP.
SELECT	Specify SELECT or SEL.
cp	Specify the number of the code page you want to select. The supported code page numbers are: 437, 850, 852, 857, 860, 862, 863, 865, and 866.

Example

The following command selects Code Page 850 for the printer on port LPT2:

```
C>MODE lpt2 CP SEL = 850
```

Displaying the current code page

Format

```
MODE device CODEPAGE [/STATUS]
```

Explanation

Use this version of the MODE command to display the currently selected code page, followed by a list of the prepared code pages.

The list contains hardware code pages and code pages prepared using the MODE PREPARE command.

The active code page status and code page list appear as follows:

```
Active codepage for device XXX is nnn

hardware codepages:
codepage nnn
prepared codepages:
codepage nnn

Completed codepage status function
```

Note that *XXX* is the device and *nnn* is the code page number.

If no code page is active on the drive, you see the following message:

```
No codepage has been selected
```

If you specified additional code pages in the DEVICE statement but did not prepare them, you see:

```
Codepage not prepared
```

Ensure that all printers affected by the MODE command are switched on, and online. Otherwise, you get an error message that may not appear immediately.

Option	**Explanation**
device	Specify CON, PRN, or LPT*n* (where *n* is 1, 2, or 3). Note that PRN is equivalent to LPT1.
CODEPAGE	Specify CODEPAGE or CP.
/STATUS	Optionally, specify /STATUS or /STA.

Example

The following command displays the prepared code pages for the CON display device:

```
C>MODE con CP
```

You would see a message similar to the following:

```
Active codepage for device CON is 850

hardware codepages:
codepage 437
prepared codepages:
codepage 437
codepage 850

Completed codepage status function
```

Refreshing a code page

Format

```
MODE device CODEPAGE REFRESH
```

Explanation

Use this version of the MODE command to refresh the active code page for a specified device.

Refreshing means, for example, that if code page information has been lost when a printer was turned off, the fonts for the active code page are reloaded on to the printer.

All parameters are required.

You see the following message when the command completes:

```
    Completed codepage refresh function
```

Ensure that all printers affected by the MODE command are switched on, and online. Otherwise, you get an error message that may not appear immediately.

Option	Explanation
device	Specify CON, PRN, or LPT*n* (where *n* is 1, 2, or 3). Note that PRN is equivalent to LPT1.
CODEPAGE	Specify CODEPAGE or CP.
REFRESH	Specify REFRESH or REF.

Example

The following command refreshes the code page for the printer on port LPT1:

```
C>MODE lpt1 CP REF
```

If the printer was not switched on or online when you entered the command, you would see an error message when the timeout period for your system elapses. Depending on the length of your timeout period, there could therefore be a delay before you see the message.

MORE

Format

```
MORE [/?|/H]

source | MORE
```

Note

In the above syntax, the vertical line (|) is used in two ways: in `/?|/H` the vertical line signifies either/or, while in `source|MORE` it is the piping symbol.

Explanation

The MORE command is a filter that takes input from a standard input device (for example a redirected file or a piped command) and then displays it a screenful at a time. At the bottom of each screen (except the last one) the phrase:

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

is shown. Move to the next screen by pressing any key.

Command switches

/? or /H	Displays help text.

Example

Use MORE with the DIR command to display a large directory a screenful at a time. The output from DIR is piped to MORE and then displayed.

```
A>DIR c: | MORE
```

For more information on piping, see Chapter 8, "Redirecting input and output".

MOVE

Format

```
MOVE [/?|/H] [@]wildspec dirpath [/A] [/D:date] [/H]
[/M] [/P] [/R] [/S] [/T] [/V] [/W]
```

Explanation

Use the MOVE command to move a file or group of files to another location on the same disk or another disk. You can also use the command to move an entire subdirectory.

MOVE copies each specified file to the specified subdirectory and then deletes the original file. MOVE does not delete subdirectories that are empty after moving, unless you specify the /T switch.

Note that the MOVE command will only copy, and not actually move, read-only files and subdirectories.

Moving files

The *wildspec* is the drive, path, and name of the file or files you want to move. You can also specify a file containing a list of files to move by typing the @ character immediately before the *wildspec*.

The *dirpath* is the drive and path to which you want to move the file or files. You must specify a single *dirpath*; wildcard characters are not allowed.

WARNING If you specify a destination for a file that already contains a file with the same name, MOVE overwrites the file on the destination. The only exception is a read-only file, in which case MOVE does not move, or copy, the file you specified. You can override this exception by using a command switch.

Moving subdirectories

To move the entire contents of a subdirectory and the directory structure, the *wildspec* is the drive and path of the subdirectory you want to move and the *dirpath* is the destination drive and path. You must also include the /T switch to specify that you are moving a subdirectory rather than a file, which is the default for MOVE.

Command Switches

/? or /H	Displays help text. Note that /H must be the only parameter you type, if you are using it to show help.
/A	Moves only those files for which the archive attribute is set, without changing the attribute setting. The archive attribute indicates that the file has been changed. Refer to the ATTRIB command in this chapter and section 4.3 in Chapter 4, "Protecting your information" for information about file attributes.
/D:*date*	Moves only those files modified since the specified date. The format of the date depends on your current country code. Refer to the DATE command in this chapter for more information.
/H	Moves hidden and system files. Without /H these files will be ignored. Refer to the ATTRIB command in this chapter and section 4.3 in Chapter 4, "Protecting your information" for information about the hidden and system attributes.
/M	Moves only those files for which the archive attribute is set and resets the attribute after moving.
/P	Prompts before moving each file.
/R	Overwrites read-only files on the destination that have the same name.
/S	Moves all the files in a subdirectory but not the subdirectory structure.
/T	Moves all files in the specified subdirectory, including all files in all subdirectories below it. Note that when you use /T, most of the other MOVE switches, which apply to files, are not allowed. Use only /P, /V, and /W with /T.
/V	Compares the moved file or subdirectory on the destination with the data before it was moved to verify the accuracy of the move. If there is any inaccuracy, the move aborts.

Command switches (continued)	
/W	Waits for a diskette to be inserted before starting to search for files to move.

Examples

Assume you have a diskette in drive a: that contains a subdirectory called **\text** and that **\text** contains a number of files with the extension **.txt**.

The following command moves all the files with extension **.txt** to a subdirectory called **\txtfiles** on drive c:

```
C>MOVE a:\text\*.txt c:\txtfiles
```

The next command would move the entire subdirectory **\text** so that it would become a subdirectory of **\txtfiles** on drive c: and prompt you before moving each file in **\text**:

```
C>MOVE a:\text c:\txtfiles /T /P
```

NLSFUNC

Format

```
NLSFUNC [/?|/H] filespec [/ML|/MH|/MU]
```

Explanation

NLSFUNC provides support for extended country information and allows you to use the CHCP command to select code pages for all devices set up to use them.

NLSFUNC must be loaded before you can use the CHCP command. To load NLSFUNC, type NLSFUNC on its own at the command line. It uses the country-specific information specified by the **country.sys** file. The **country.sys** file used is the one as specified by the COUNTRY= statement in your **config.sys** file. (You can use the SETUP program to modify your **config.sys** file; see Chapter 11, "Customizing your system".)

NLSFUNC followed by *filespec* specifies the filename and location of the country information file to be used. For example:

```
NLSFUNC c:\drdos\country.sys
```

Command switches	
/? or /H	Displays help text.
/ML|/MH|/MU	Forces NLSFUNC to be loaded into a specific area of memory; /ML for conventional (lower) memory, /MH for high memory, or /MU for upper memory. Note that if there is insufficient high or upper memory, NLSFUNC is loaded into conventional memory. Before you can use /MH or /MU, a device driver that supports high and upper memory must be loaded.

For more information on code page switching, see Chapter 14, "Setting up code page switching".

PASSWORD

Format

```
PASSWORD[/?|/H] [[@]wildspec...]
[/R|/W|/D|/P|/G[:password]][/N][/NP][/NG][/S]
```

Explanation

The PASSWORD command is used to assign passwords to files or paths.

Passwords can be up to eight letters long.

When you issue the PASSWORD command without any parameters, it lists the password attributes assigned to the files in your current directory.

Command switches

/? or /H Displays help text.

/R:*password* The password is required for reading, copying, writing, deleting, renaming, and changing the file's attributes. /R creates the password attributes R,W,D.

/W:*password* The file can be read and copied without giving a password, but the password is required to delete, rename or change the file's attributes. /W creates the password attributes W,D.

/D:*password* The password is required only for deleting, or renaming the file, or changing the file attributes. You do not need the password to read, or modify the file. /D creates the password attribute D.

/P:*password* The password operates on subdirectories instead of files. This means each time you try to use the path to access a directory you are prompted for a password. Subdirectories are automatically given full read, write and delete protection when a password is applied.

Command switches (continued)

/G:*password* Sets the global default password. DR DOS remembers this default password and tries it against each password-protected file you then try to access. This is useful if a group of files have the same password and you want to access the files without typing a password each time. The global default password remains set until it is removed, or until the computer is reset. It is also used before a non-password aware application is used with password-protected files.

/N Removes any password protection. Note that you are prompted for the password before it is removed. /N can also be used in combination with the /P and /G options:

/NP Removes password protection from a directory.

/NG Removes the global default password.

/S The password operates on files and subdirectories associated with the current directory.

Examples

The following example assigns the password SUZE to the file **eyesoff.let**, and ensures that the password must be given before the file can even be read.

```
A>PASSWORD eyesoff.let /R:suze
```

The following command protects all files with the **.bas** extension in the **\handsoff** directory and in all the associated subdirectories. The level of protection is set at W; the files can be read and copied, but not renamed, deleted or altered. Note that you do not need to use the password to access the **\handsoff** directory or the subdirectories; you only need the password with ***.bas** files.

```
A>PASSWORD c:\handsoff\*.bas /W:simon /S
```

The following example assigns the password WINSTON to the subdirectory **top** in the path **c:\secret\top**:

```
A>PASSWORD c:\secret\top /P:winston
```

If you then type:

```
A>CHDIR c:\secret\top;winston
```

you can access the files associated with the subdirectory **\secret\top** without using the password again (as long as the files are not individually password-protected).

PATH

Format

```
PATH [/?|/H] [[d:]dirpath [;[d:]dirpath]... |;]
```

Explanation

PATH sets a search path for commands or batch files not found by a search of the current directory. If a command cannot be found in the current directory, DR DOS searches the named directories in the order you entered them, but does not change the current directory. The maximum length of a search path is 123 characters.

PATH with no parameters displays the names indicated in the previous PATH command. (The search paths currently defined.)

PATH with a semi-colon:

```
C>PATH;
```

resets the search path to *no extended search path* or a "null" path. If no path is specified, DR DOS only searches in the current directory of the current drive.

Incorrect information in the paths will not be found until the specified paths are searched. If a specified path does not exist, DR DOS ignores that path and goes to the next.

If the command entered is not found in any of the directories specified in PATH, the message:

```
    Command or filename not recognized
```

is displayed.

Command switches	
/? or /H	Displays help text.

Examples

In the following example, **spell.com** is contained in directory **mydir** on drive B:, and the default drive is drive A:

```
A>PATH a:\sub1;a:\sub1\sub2;b:\mydir
```

This command sets the search path so that when you type *spell* at the command line DR DOS looks into the current directory of the current drive specified, followed by **a:\sub1**, then **a:\sub1\sub2**, then **b:\mydir**, until **spell.com** is found and loaded.

If you enter the command:

```
A>SPELL
```

DR DOS searches four directories and finds program **spell** in **b:\mydir**.

PRINT

Format

```
PRINT [/?|/H] [wildspec] [/D:device] [/B:buffsize]
[/U:busyticks] [/M:maxticks] [/S:timeslice]
[/Q:queuesize] [/T] [/C] [/P]
```

Explanation

PRINT is used to print text files. When you have created files on disk that need printing, you can use PRINT to manage the print process. PRINT creates a list of files called a *print queue* and then prints the files one by one.

PRINT runs in the background, leaving your computer free so you can carry on working as normal. PRINT is an external command in DR DOS, but part of it remains loaded in memory. When you start the PRINT command, you see the message:

```
Resident part of PRINT installed
```

Command switches

/? or /H — Displays help text.

/D:*device* — Sends the printer output to the named device. The default is PRN (which is the same as LPT1). The other valid devices you can name are: AUX, COM1, COM2, LPT1, LPT2, and LPT3. The device name can only be specified the first time the PRINT command is used during a DR DOS session.

/B:*buffsize* — Sets the size of the print buffer (in bytes). The default is 512. A larger buffer can (sometimes) improve the print speed. The buffer size can only be set the first time the PRINT command is used during a DR DOS session.

/U:*busyticks*
/M:*maxticks*
/S:*timeslice* — These parameters work together to govern what portion of computer time PRINT is allowed to use. *Timeslice* (default 8) describes how often the computer gives control to the PRINT utility; *maxticks* (default 2, range 1

Command switches (continued)

through 255) governs for how many clock-ticks PRINT runs once it has control; *busyticks* (default 1) specifies how long PRINT waits before handing back control if the printer is busy. (A clock-tick is approximately 1/18th second.)

Note

Each of these parameters can only be specified the first time you issue the PRINT command during a DR DOS session.

/Q:*queuesize* Sets the maximum number of files that can be held in the print queue. The default is 10, in the range 4 to 32. This parameter can only be defined the first time you use PRINT during a DR DOS session.

/T Deletes all the files from the print queue. The printer prints the message:

```
All files cancelled by operator
```

/C Deletes the named file(s) from the print queue. Filenames can be typed before or after this option in the command line: the option looks at the one filename immediately preceding it in the command line and at all the filenames typed after it.

/P Adds the named files to the print queue. The rules for using it are the same as those governing the /C option.

Examples

If you type this command when using PRINT for the first time after starting up DR DOS:

```
C>PRINT /Q:20 chap1.doc
```

the command sets the spool queue size to 20. As the /D option was not given, you get this message:

```
List device? [PRN]
```

You need to press **Enter** to print via the default port LPT1 (LPT1 is equivalent to PRN). Type the port name if you want to print via another port. If PRINT was already started, using the /Q option would result in an error message.

The following command deletes the files **chap2.doc** and **chap3.doc** from the print queue:

```
C>PRINT chap2.doc /C chap3.doc
```

PRINT allows you to combine several commands on one command line.

The following command deletes all the files on the print queue and then adds files **newbookl.doc** and **newbook2.doc** to the queue:

```
C>PRINT /T newbookl.doc newbook2.doc
```

PROMPT

Format

```
PROMPT [/?|/H] [promptstring]
```

Explanation

The PROMPT command modifies the command prompt. You can set the command prompt to contain any valid ASCII characters or certain special character-strings. You can also set the prompt to automatically execute a command every time it displays.

Type PROMPT with no parameters to reset the command prompt to the default DR DOS prompt. The default prompt displays the standard DR DOS prompt, C>. (The default prompt is indicated by ng where *n* is the default drive letter and *g* is the character >.)

The *prompt string* can be any of the valid ASCII characters or one of the following (you must precede it with a dollar sign, $, for example $t):

Character	Meaning
$	The $ character.
t	The time.
d	The date.
p	The current directory path of the default drive.
v	The version number.
n	The default drive letter.
g	The ">" character.
l	The "<" character.
b	The | character.
q	The = character.
h	A backspace; the previous character is erased.
e	The ESCape character.

Character	Meaning (continued)
x	Tells DR DOS to run a command every time you return to the prompt. You specify the command to run with a SET statement that sets the value of the PEXEC environment variable. Note that x itself does not cause anything to display in the command prompt; it simply points to the PEXEC value. You can put both the PROMPT command and the SET command in **autoexec.bat**.
_	The CR LF sequence (Carriage-return and line feed, go to beginning of new line of the screen).

Any other value is considered invalid and ignored by PROMPT (for example $c).

Command switches	
/? or /H	Displays help text.

Examples

The following command sets the DR DOS prompt to HELLO:

```
C>PROMPT hello
```

The following command sets the DR DOS prompt to the default drive and path plus the character >. This is the recommended prompt for use on a hard disk.

```
C>PROMPT $p$g
```

The following command sets the DR DOS prompt to display the time and date:

```
C>PROMPT time = $t$_date = $d
```

The DR DOS prompt appears as follows:

```
TIME = current time
DATE = current date
```

The next command sets the prompt to display the current directory path and the message `Wait...`, runs the command defined by the PEXEC parameter in a SET statement, and then backspaces to erase `Wait...` and display the > character:

```
C>PROMPT $pWait...$x$h$h$h$h$h$h$h$g
```

A DISKMAP command would run every time you returned to the command prompt, assuming you also specified the following in your **autoexec.bat** file:

```
SET pexec=C:\drdos\diskmap C:
```

RECOVER

Format

```
RECOVER [/?|/H] d:|wildspec
```

Explanation

The RECOVER command enables you to retrieve information from files that have been corrupted. If you have a file that is only partially readable, RECOVER removes the unreadable parts so you can access the rest. It has two forms: you can either recover individual files or groups of files, or you can recover an entire disk.

Command switches	
/? or /H	Displays help text.

WARNING Only recover a disk if your *root directory* is corrupted, because if you only specify a disk RECOVER assumes the *entire* disk is scrambled and replaces your root directory with a *new* one. All the information contained in your files is re-organised into a series of files as described above, and you then have to sort out where your data is and re-structure it.

Recovering a disk

To recover an entire disk you type:

```
RECOVER d:
```

Each recovered file is converted into a file in the root directory. These files have filenames in the form FILE*xxxx*.REC, where *xxxx* specifies a number starting at 0000.

Note

You may need to repeat the operation if there are more cluster chains to convert than there are available entries in the root directory. In this case you should *copy* the first set of recovered files to another directory on another disk, delete them from the root directory, and then re-issue the command.

Recovering individual files

You can use RECOVER to recover single files or a specific group of files (you can use wildcards in the specification). For example:

```
C>RECOVER c:\reports\invoice
```

In this case each file in the path that matches the specified filename, for example **invoice**, is checked for bad areas (or "blocks"). If DR DOS finds any bad blocks they are skipped over and marked as bad blocks in the File Allocation Table. The record of the file's length (shown when you use the command DIR) is also modified to reflect the new length of the file.

Once you have recovered the file or files, you can edit them and re-enter lost information.

Remarks

If you specify a subdirectory in the command syntax, it means that the files in that subdirectory are recovered and copied to the *root* directory, not to the original subdirectory where the files were stored.

RECOVER is most useful when you are attempting to recover *text* files.

RECOVER does not work on networked drives.

RECOVER does not work on drives affected by a JOIN or SUBST command.

Example

The following command checks whether there are any bad blocks in the files matching **salesm?.doc** in directory **reports** on drive C:. Any bad blocks are marked in the File Allocation Table and the record of the files' length is modified accordingly.

```
C>RECOVER c:\reports\salesm?.doc
```

RENAME (REN)

Format

```
RENAME [/?|/H] oldwildspec newwildspec
```

Explanation

The RENAME command lets you change the name of a file or move a file between subdirectories on the same disk. RENAME can be shortened to REN.

Command switches	
/? or /H	Displays help text.

Renaming a file

If the file to be renamed is not on the current drive or in the current directory, you must precede the old file specification with the correct drive/path specification. The following command line renames **myfile.typ** as **file1.txt**:

```
A>REN b:myfile.typ file1.txt
```

Wildcards can be used, for example the command:

```
B>REN *.in *.out
```

renames all files with the file extension **.in**, so that they have the file extension **.out**.

RENAME does not let you rename a file to an existing name. If you attempt to do this the message:

```
File already exists
```

is displayed, and the command prompt returned. Any files designated to be renamed remain unchanged.

Password-protected files

On password-protected files, you must enter the default password before renaming a file. The same file password is then required to access the new file specification. You can remove the password protection on the file before renaming it (see the PASSWORD command).

Moving files

RENAME can be used to move files between subdirectories on the same disk. For example the command:

```
A>REN myfile.lst \wordproc\memos
```

moves **myfile.lst** to the directory **a:\wordproc\memos**; **myfile.lst** no longer exists on the original directory.

Note

This operation can save time when you want to move several files. Previously you would have had to copy the files then delete the originals. Using RENAME, only the directory entries are moved, so the operation is very fast.

RENDIR

Format

```
RENDIR [/?|/H] [d:] [dirpath]olddirname
[dirpath]newdirname
```

Explanation

The RENDIR command lets you change the name of a subdirectory.

Type the command followed by the current name and the new name. You must also type the drive/path specification before the current name and the new name if the subdirectory you are renaming is not on your current drive or is not your current directory.

Note that if you try to specify a new subdirectory name that already exists, you receive an error message, the command prompt returns, and the subdirectory is not renamed.

Command switches	
/? or /H	Displays help text.

Example

The following example would rename a subdirectory **france**, which is on the current drive but which is not the current directory, as **french**:

```
C>RENDIR clients\foreign\france clients\foreign\french
```

REPLACE

Format

```
REPLACE [/?|/H] [@]wildspec dirpath [/A] [/H] [/M]
[/N] [/P] [/R] [/S] [/U] [/W]
```

Explanation

REPLACE copies selected files from a source to a destination. It is useful for updating previous versions of files. For example, you can *add* files to a destination directory or you can replace only files that have been changed since the last copy. Use the /A switch to add files to a subdirectory.

The *wildspec* is the drive, path and name of the *source* files; *dirpath* shows the drive and path where you want the files to be placed.

As well as replacing single files in one directory, you can also replace files across subdirectories using the /S switch.

Command switches

/? or /H	Displays help text.
/A	Copies only files not already on the destination.
/H	Does not ignore files with hidden or system attributes.
/M	Merges changed files on the source with unchanged files at the destination.
/N	Preview operation - does not actually copy any files.
/P	Prompts before copying each file.
/R	Overwrites read-only files on the destination with fresh copies from the source.
/S	Looks in subdirectories of the destination for files that match the *wildspec*.
/U	Only copy files that are older on the destination than on the source.

Command switches (continued)	
/W	Waits for disks to be changed before starting to look for the source files. This allows the REPLACE program to be loaded from a diskette and the diskette changed for the source diskette.

Example

The REPLACE option is especially useful for updating software. For example, you might have a program called **wcounter.bas** in various subdirectories on your hard disk. If you change the copy of **wcounter.bas** in the **\wordproc** directory, you can copy the changed version of the file to all the other places in the hard disk by using the following command:

```
C>REPLACE a:\wordproc\wcounter.bas c:\ /S
```

RESTORE

Format

```
RESTORE [/?|/H] source_drive destination_drive
[filespec] [/S] [/P] [/A:date] [/B:date]
[/E:hh:mm:ss] [/L:hh:mm:ss] [/M] [/N] [/R]
```

Explanation

Restores directories and files that were backed up with the DR DOS BACKUP utility. *Source_drive* is the drive containing the backed up files, and *destination_drive* is where you want to restore your files to. RESTORE prompts you for each backup diskette in turn until all the files you specified are restored.

Note

The RESTORE command can be used with any previous version of a DOS BACKUP utility.

Command switches

/? or /H — Displays help text.

/S — Restores all subdirectories associated with the current directory on the destination disk. RESTORE will create the required subdirectories on the destination disk if necessary.

/P — Prompts you for permission to restore read-only files, or files that have changed since the last backup.

/A:*date* — Restores all the files that have been altered on the original disk on or after the date specified. The format of the date depends on the current country code (see the DATE command).

/B:*date* — Restores all the files that have been altered on the original disk on or before the date specified. The format of the date depends on the current country code (see the DATE command).

Command switches (continued)

/E:*hh:mm:ss*	Restores all the files that have been altered on the original disk at or earlier than the time specified.
/L:*hh:mm:ss*	Restores all the files that have been altered on the original disk at or later than the time specified.
/M	Restores all the files that have been altered on the original disk, or deleted, since the backup was taken.
/N	Restores files that do not exist on the destination.
/R	Reports which files would be restored, but does not take any action. This is useful for checking that you will restore the files that you want. This option also checks the integrity of backup disks.

Remarks

RESTORE reports any errors during the restoration process using the codes shown in Table 10-14.

Table 10-14 RESTORE Error Codes

Code	Explanation
0	Normal completion.
1	No files found to restore.
2	Some files not restored due to file sharing conflict.
3	Restoration process ended by a **Ctrl+Break**.
4	Restoration process ended by an error.

This feature can be useful in creating batch files with an `IF ERRORLEVEL` statement in them to improve the process, see Chapter 9, "Grouping files in batch files".

Do not use the RESTORE program with disks affected by an ASSIGN, JOIN, or SUBST command.

Examples

This example restores a backup of all files on all directories on drive C:

```
C>RESTORE a: c:\  /S
```

This command restores the file **letter.doc** in the subdirectory **\backedup** on drive C:

```
C>RESTORE a: c:\backedup\letter.doc
```

RMDIR (RD)

Format

```
RMDIR | RD [/?|/H] [d:]dirpath
```

Explanation

The RMDIR command removes a specified subdirectory. The subdirectory to be deleted must be empty except for the (.) and (..) markers which refer to the directory itself, and its parent directory; these entries cannot be deleted. RMDIR can be abbreviated to RD.

Before you remove a subdirectory, ensure that:

- ❑ The subdirectory contains no files or additional subdirectories.
- ❑ The subdirectory is not the current directory on *any* drive.
- ❑ The subdirectory is not in use by a SUBST or JOIN command.

Command switches	
/? or /H	Displays help text.

Remarks

If you get the error message:

```
    Directory not empty or in use
```

but when you check the relevant subdirectory only the . and .. entries can be seen, check again with the XDIR command. The XDIR command will show if any hidden, system, or password-protected files are present.

Examples

This command erases subdirectory **france** on drive A:.

```
A>RMDIR clients\foreign\france
```

This command erases subdirectory **accts.rcv** on drive C:.

```
A>RD c:\finance\accts.rcv
```

SCRIPT

Format

```
SCRIPT [/?|/H] [/U] [device|filename]
[device|filename][/O=P|L][/H=nn][/T=nn][/R]
```

SCRIPT provides PostScript support for DR DOS and programs running under it.

Use the SCRIPT command to translate all characters in text files, including those conforming to the Hewlett-Packard LaserJet II standard, to PostScript.

You can use SCRIPT from the DR DOS command line to take input from a file, the system console, or another program, translate it, and send it to a printer or file.

You can install SCRIPT as a TSR, to intercept all data intended for a printer, translate it, and send the data to a PostScript printer.

You can also use SCRIPT on the command line when SCRIPT is already loaded as a TSR to change the current parameters for the specified device in the resident part of SCRIPT.

Command switches	
/? or /H	Displays help text.
/U	Uninstalls SCRIPT, if you installed it as a TSR.

Specify SCRIPT with *both* a source specification and a destination specification if you want to take input from the source and print to the destination. If you specify an output device (LPT1, 2, or 3) as the source, SCRIPT automatically installs as a TSR and intercepts all data intended for the source, processes it, and sends it to the destination.

Specify SCRIPT with a source specification *only* and LPT1 is assumed as the destination.

Specify SCRIPT with *neither* specification to process data from standard input to standard output, if SCRIPT is not loaded as a TSR.

The following table describes the options you can use in your source and destination specifications:

Options

/O[RIENTATION]=P[ORTRAIT] | L[ANDSCAPE]

PORTRAIT is the default.

/P[OINTSIZE]=*nn*

Specify *nn* where *nn* is a point size. The default point size is 11.

/T[IMEOUT]=*nn*

Sets a timeout to pre-empt the normal spooler or printer timeout and print any information remaining in the spool file or printer before the printer times itself out. Use /T when the spool is timing out too soon and transferring data to a printer before all data for the printer has been collected by the spooler. Specify *nn* where *nn* is the timeout in seconds. The default timeout is 10 seconds.

/R[ESET]

Generates a software reset before processing any data.

Example

Following are examples of source specification and a destination specification:

```
report.1st/reset
lpt1:/orientation=landscape/pointsize=8.5
```

SET

Format

```
SET [/?|/H] [name=[parameter]]
```

Explanation

SET inserts variable strings into the command processor's environment. All the strings in the environment are available to all commands and applications.

The string (beginning with *name*) is inserted into a part of memory reserved for environment strings. Lower case letters in the name are converted to upper case when added to the environment. If the name was in the environment, it is replaced by the new parameter.

If you type SET with no options you see a list of the strings already in the environment.

To delete a name from the environment, type SET with a variable name and no parameter: SET name=.

Command switches	
/? or /H	Displays help text.

Remarks

Temporary files

You can use the special environment variable TEMP in a SET command to specify a subdirectory for *te*mporary files. Commands such as DISKCOMP and DISKCOPY can use temporary files, and automatically delete them afterwards. The TEMP parameter *must* be an existing subdirectory.

Among the strings set automatically by DR DOS when DR DOS is started are:

```
COMSPEC=C:\COMMAND.COM
```

and

```
OS=DRDOS
```

COMSPEC defines the path to the command processor. (The command processor in DR DOS is called **command.com.**) COMSPEC is used by programs that sometimes look for the command processor; typically programs that allow you to access operating system commands without leaving the program.

Command.com is provided with DR DOS; you can, however, insert your own command processor instead by setting a new COMSPEC.

OS describes the operating system *type*. This environment string is useful for distinguishing between operating systems. For example, if you run two or more different operating systems at different times on one computer, you can devise a batch file that looks for the OS environment string, finds which operating system is present, and then takes appropriate action.

SET can be used in your **autoexec.bat** file to automatically set strings when you start DR DOS.

Example

With the following command, if the operating system currently running is DR DOS, it prompts for the global password. If a different operating system is running, it does nothing.

```
IF "%OS%" == "DRDOS" PASSWORD /G:
```

The next command defines the directory **c:\drdos** as the location for a temporary file:

```
TEMP=C:\DRDOS
```

SHARE

Format

```
SHARE [/?|/H] [/L:nnnn|/X] [/ML|/MH|/MU]
```

Explanation

SHARE is a DR DOS program that provides support for file locking, (by which a file, or part of a file, is reserved for unique access by one program) to allow file sharing. Share is used with networked systems and some TSR programs, where the same data and files need to be shared at the same time by different programs. SHARE, as its name implies, coordinates this *sharing* of information. When SHARE is loaded, DR DOS checks all read and write requests made.

Command switches	
/? or /H	Displays help text.
/X	Disables SHARE.
/L:*nnnn*	Allocates the number of file locks you want, where *nnnn* is a decimal number in the range 20 to 1024. The default, if you issue a SHARE command without specifying a value using this switch, is 20.
/ML|/MH|/MU	Forces the SHARE program to be loaded in a specific area of memory; /ML for conventional (lower) memory, /MH for high memory, or /MU for upper memory. Note that if there is insufficient high or upper memory for the program, it is loaded into conventional memory. Before you can use /MH or /MU, a device driver that supports high and upper memory must be loaded.

Remarks

As DR DOS has built-in support for large hard disks, it does not require SHARE to be loaded to access them.

When SHARE is loaded, it increases the resident size of DR DOS in memory.

You can put SHARE in your **autoexec.bat** file, so it is loaded every time you start up your system; see Chapter 11, "Customizing your system").

Examples

To make 40 locks available in your system type:

```
C>SHARE /L:40
```

To disable SHARE:

```
C>SHARE /X
```

SORT

Format

```
SORT [/?|/H] [/R] [/+n]
```

Explanation

SORT is a filter program that reads standard input, sorts it alphabetically, and then writes it to the standard output device.

You can use the redirection symbols <> and | to pipe data from a filename or command through the SORT program. For example:

```
C>DIR | SORT >filename.txt
```

takes the output from the DIR command, sorts it and then redirects the sorted data to **filename.txt**.

Note

The SORT command does not differentiate between upper and lower case letters.

Command switches

/? or /H	Displays help text.
/R	Sorts in reverse order, that is 9 to 0, or Z to A.
/+*n*	Makes the sort start on the "*n*th" column of a line. If omitted, the sort starts from the default, column 1.

For more information on redirection, piping and the SORT command, see Chapter 8, "Redirecting input and output".

Examples

The following example reads and sorts the file **index.dis** and writes the output to the printer.

```
A>SORT<index.dis>prn:
```

The following example uses SORT to present a directory listing in *reverse* alphabetical order and then stores it in the file **reverse.ord**.

```
A>DIR | SORT /R >reverse.ord
```

SSTOR

Format

```
SSTOR [/?|/H] [/B]
```

Explanation

The SSTOR command starts the SuperStor data compression program. You can use SuperStor to increase the capacity of a hard drive by converting it to a SuperStor compressed drive; consequently, data is automatically compressed when written to the drive and decompressed when read from the drive. The SuperStor program attempts to compress the data as much as possible, but the actual compression ratio it achieves depends on the type of data being compressed.

Command switches

/? or /H	Displays help text.
/B	Specifies a monochrome (no color) display.

Type SSTOR to display the SuperStor menu, from which you can select any of the following options:

- ❑ **Prepare** to format and initialize a compressed drive.
- ❑ **Remove** to remove a SuperStor compressed drive.
- ❑ **Statistics** to display statistics about SuperStor drives.
- ❑ **Help** to display help text.
- ❑ **Exit** to return to the command prompt.

For a complete description of SuperStor and how to use the SSTOR command, refer to Chapter 13, "Optimizing your disk".

Note

You must load the SSTORDRV.SYS device driver before you can use the SuperStor program, or access SuperStor drives; you can choose to load SSORSRV.SYS when you run INSTALL or SETUP, in which case the correct statements are automatically added to your **config.sys** file.

SUBST

Format

```
SUBST [/?|/H] [d: d:dirpath | [d:] /D]
```

Explanation

SUBST allows you to make a subdirectory appear as if it were another disk drive. You do this by replacing a drive-and-path specification with a single drive letter. So, for example, you can replace a path like **\wordproc\simon\letters** with the drive F:. You can then use drive F: in your commands as if it was a real disk drive. This saves you from having to type in long path specifications over and again, and it can be used for programs which do not recognize subdirectories.

Typing SUBST without any parameters displays the currently substituted drives.

Caution When using substituted drives you should avoid using the DR DOS commands that involve entire disks.

BACKUP
CHKDSK
DISKCOMP
DISKCOPY
FDISK
FORMAT
LABEL
RECOVER
RESTORE
SYS

Always switch off a substitution when you have finished working with it.

Command switches

/? or /H	Displays help text.
/D	Removes substitutions when you have finished working with them. /D operates on the specified drive, or on all substitute drives if none is specified on the command line.

Examples

The following example assigns drive F: to the path **\wordproc\simon\letters** on drive C:

```
A>SUBST  f:  c:\wordproc\simon\letters
```

The example below removes the assignment from drive H:

```
A>SUBST  h:  /D
```

SUPERPCK

Format

```
SUPERPCK [/?|/H] [options]
```

Explanation

The SUPERPCK command loads and unloads the Super PC-Kwik® disk cache. This is a memory-resident program that speeds up your system's performance by reducing the number of times an application has to physically access hard disks or dikettes. It also increases the speed at which data is written to disk.

Chapter 13, "Optimizing your disk" explains how to use SUPERPCK, and describes all the SUPERPCK options.

Note

You can choose to install the disk cache when you run INSTALL or SETUP, in which case the correct statements to load and run the disk cache program are automatically added to your **config.sys** and **autoexec.bat** files.

Command switches	
/? or /H	Displays help text.

Example

The following command unloads the disk cache program from memory:

```
SUPERPCK /U
```

SYS

Format

```
SYS [/?|/H] d:
```

Explanation

SYS transfers the DR DOS operating system files and **command.com** from the default drive (that is the drive on which SYS is being run) to a specified destination drive (d:).

SYS allows you to transfer the operating system files to a diskette to make a drive bootable (some program diskettes need DR DOS on them to work), or to change them to match the ones on the default (or SYS) drive.

SYS also transfers the command processor defined by COMSPEC to the root directory of the destination drive.

Command switches	
/? or /H	Displays help text.

Remarks

DR DOS does not require space to be reserved for its system files on a disk, or that they be placed at the beginning of the disk; therefore you should have no problems making a disk bootable.

SYS does not work on drives affected by a SUBST or JOIN command or across networks.

Examples

To transfer the system files from drive C: to drive A: type:

```
C>SYS a:
```

To transfer the system files from drive A: to drive C: (you must have booted from drive A:) type:

```
A>SYS c:
```

TASKMAX

Format

```
TASKMAX [/?|/H] [/D=dirpath][/X[=nnnn]] [/E[=nnnn]]
[/L=nnnn] [/C command] [/K:nn] [/N[:nn][name]]
```

Explanation

TaskMAX is a utility which allows you to switch between applications quickly and easily. You add applications as *tasks* to the TaskMAX menu, which you can display at any time. When tasks are not being used they are frozen in the background exactly as you left them. The background tasks are swapped out to a swap file in memory or to disk. If you have expanded or extended memory available, you can specify that your background tasks are swapped there. TaskMAX always swaps tasks to memory first; when no more memory is available, it swaps to disk. You can also specify the location of the swap file on the disk.

TaskMAX can be installed using the SETUP program so that it is loaded automatically every time you switch on your computer. Otherwise, you can load it directly from the command line by typing:

```
C>TASKMAX
```

followed by the appropriate parameters to configure TaskMAX the way you want it; see below.

Note

The SHARE program should always be run before loading TaskMAX from the command line; see the description of SHARE in this chapter.

A message is displayed saying that the DR DOS application switcher is loaded and you can now display the TaskMAX menu by pressing **Ctrl+Esc**. Refer to Chapter 6, "Switching tasks" for more information on how to use TaskMAX.

Command switches

/? or /H — Displays help text.

/D=*dirpath* — The programs you suspend in the background are swapped to the file in the specified location on your hard disk. The default *dirpath* is **c:\drdos\tmp**. Ensure that the drive you set your swap path to has enough free space to store the tasks you want to load.

You may want to swap files to a memory disk (see VDISK.SYS in Chapter 11, "Customizing your system") to increase the speed with which you can switch between applications; that is, the speed with which tasks are swapped. If your C: drive is nearly full, and you have a D: drive, you may want to swap tasks to a directory there.

To calculate the size of memory disk, load all the applications you would typically use and check the size of the swap file created on the disk. Set up the memory disk to be slightly larger than the size of the swap file and then configure TaskMAX to use the memory disk.

Note that /D is only applicable when you are loading TaskMAX.

/X[=*nnnn*] — Swaps your background tasks to expanded memory, where *nnnn* is the size of the expanded memory in Kbytes. If you type /X on its own, all expanded memory is allocated. Entering /X=0 disables use of this type of memory (for example, you may want to reserve all expanded memory for your applications).

Note that /X is only applicable when you are loading TaskMAX.

/E[=*nnnn*] — Swaps your background tasks to extended memory, where *nnnn* is the size of the extended memory in Kbytes. If you type /E on its own, all extended memory is allocated. Entering /E=0 disables use of this type of memory (for example, you may want to reserve all extended memory for your applications).

Command switches (continued)

For both the /X and /E switches, if you enter a value that exceeds the amount of memory available, the maximum is used. For example, if you type /E=3000 and only 2 Mbytes of memory are available, 2 Mbytes are allocated.

Note that the /E switch is only applicable when you are loading TaskMAX.

/L=*nnnn* Allows you to set a limit on the amount of expanded (EMS) memory used per task. This means that if one application attempts to allocate all expanded memory to itself, this switch will override it and allocate the value you specify per task. The value you specify will apply to all tasks you add. For definitions of, and more information on using, different types of memory with DR DOS, see Appendix C, "Memory" and Chapter 12, "Using MemoryMAX". See also Chapter 6, "Switching tasks" section 6.10.

Note that TaskMAX must already be loaded before you use the /L switch.

/C *command* Allows you to add a task to the TaskMAX menu directly from the command line. *Command* is the application you want loaded as the task; you can load the application in the usual way. For example:

```
C>TASKMAX  /C  c:\wp\wordpro
```

loads the application Wordpro from the directory **c:\wp** and adds it as a task to the TaskMAX menu. If you normally load an application for example, Wordpro, using a batch file you can enter:

```
C>TASKMAX  /C  c:\bats\wp.bat
```

The batch file, **wp.bat**, is processed as normal and Wordpro is added as a task. Note that you can also add the /N switch to the batch file to preset the taskname of the application; see below.

Command switches (continued)

Note that TaskMAX must already be loaded before you use the /C switch.

/K:*nn* Removes the specified task without prompting you first, where *nn* is the number of the task in the TaskMAX menu.

Note that TaskMAX must already be loaded before you use the /K switch.

/N[:*nn*] [*name*] Allows you to specify the name of a task, overriding the name allocated by TaskMAX. You can type up to 8 characters. The name you specify remains in place for the current TaskMAX copy. You can rename either the current task or a specific task. For example:

```
C>TASKMAX  /N  spread
```

sets the name of the current task to *spread.* The following example would rename a specific task, *task 2* on the TaskMAX menu, to *spread*:

```
C>TASKMAX  /N:2  spread
```

To restore the original TaskMAX task name, type /N if it is the current task, or /N:*nn* if it is a specific task, for example:

```
C>TASKMAX  /N:2
```

restores task 2 to its original taskname which is the name of the application's executable program in memory. Note that this only takes effect when you next switch tasks.

Note that TaskMAX must already be loaded before you use the /N switch.

Remarks

If you originally used SETUP to configure TaskMAX and later you change the TaskMAX configuration at the command line, your changes will only be effective during your currently loaded copy of TaskMAX.

If TaskMAX is already resident in memory when you try to load it, you will see a message to this effect followed by a list of existing tasks.

TaskMAX will remain resident until you switch off your computer, or remove it; see Chapter 6, "Switching tasks".

Examples

The following command limits the allocation of expanded memory for each task to 1 Mbyte:

```
C>TASKMAX  /L=1024
```

The following command allocates all expanded memory for background tasks to be swapped to:

```
C>TASKMAX  /X
```

The following command disables use of extended memory:

```
C>TASKMAX  /E=0
```

The following command changes the default swap directory to drive E:, which in this case is a memory disk:

```
C>TASKMAX  /D=e:\
```

The next command loads the application Graphics as a task:

```
C>TASKMAX  /C  graphics
```

The next command renames task 5 in the task list to **wp**:

```
C>TASKMAX  /N:5 wp
```

and

```
C>TASKMAX  /N:5
```

restores the original TaskMAX taskname. Note that this only takes effect when you next switch tasks.

TIME

Format

```
TIME [/?|/H] [hh[:mm[:ss]]] [/C]
```

Explanation

Use the TIME command to display or set the time of day. (DR DOS maintains its own internal clock.) DR DOS also uses time to record the time a file is created or updated.

You must set the time whenever you start or reset DR DOS, unless you have a real-time clock with battery backup. (When you do not enter a time, DR DOS assumes the time 00:00:00.00.) Put the TIME command in your **autoexec.bat** file to prompt you when you start DR DOS. The DR DOS clock works on a 24-hour basis. This means that 1pm is 13:00 and midnight is 00:00.

Command switches

/? or /H	Displays help text.
/C	Displays the time continuously until you press any key to exit.

To check the current time setting, invoke the command without parameters:

```
A>TIME
```

TIME displays the time and prompts you as follows.

```
    Time: hh:mm:ss.xx
    Enter time:
```

You can press **Enter** to keep the current values. To enter a new time, enter the hours, minutes and seconds and then press **Enter**. Note that the last value shown above, *xx*, indicates hundredths of a second, which cannot be changed.

The time values are defined as follows:

- ❑ *hh* is a one- or two-digit number from 0 to 23 indicating the hour

- ❑ *mm* is a one- or two-digit number from 0 to 59 indicating the minute
- ❑ *ss* is a one- or two-digit number from 0 to 59 indicating the second

TIME assumes that the first entry specifies the hour. The minutes and seconds are optional, and if they are not entered are set to zero. You must use a colon to separate the entries.

If you enter a number that is out of range, TIME returns the message and prompt:

```
Invalid time specified
Enter time:
```

Examples

The following command sets the time for 30 minutes and 59 seconds after 10 pm:

```
A>TIME 22:30:59
```

The following command sets the time at 12 minutes after midnight:

```
A>TIME 00:12
```

TOUCH

Format

```
TOUCH [/?|/H] [@]wildspec[...] [/T:hh:mm:ss]
[/D:date] [/F:E|J|U] [/P] [/R] [/S]
```

Explanation

TOUCH resets the time and date stamps for a single file or a group of files. If you change the date stamps on a group of files you can also alter the date format using the /F switch.

Command switches

/? or /H	Displays help text.
/T:*hh:mm:ss*	The time to be set. The time is set in 24 hour clock with midnight being 00:00. The default is the current time.
/D:*date*	The date to be set. The default is the current date. The date format depends on the current country code, set up in the **config.sys** file or the value specified in the /F: option.
/F:[E | J | U]	Specifies European, Japanese or USA date format for the date option before touching each file. This overrides the default format determined by the current country code.
/P	Prompts before touching each file.
/R	Includes read-only files in the touch operation.
/S	Touches files in subdirectories associated with the current directory.

Examples

This example resets the date stamps for all the **.let** files. The new date is 2 February 1989. TOUCH prompts you before changing each file.

```
A>TOUCH  *.let  /D:02-02-89  /P
```

The following command resets the date stamps for all the files **states*.*** using the US date format, overriding the date format set up in the **config.sys** file:

```
A>TOUCH  states*.*  /D:06-22-89  /F:U
```

TREE

Format

```
TREE [/?|/H] [d:|wildspec] [/B] [/F] [/G] [/P]
```

Explanation

The TREE command displays the hierarchical path of directories and subdirectories on a disk, or if used with a *wildspec* finds all the files matching the *wildspec* on the disk.

Command switches

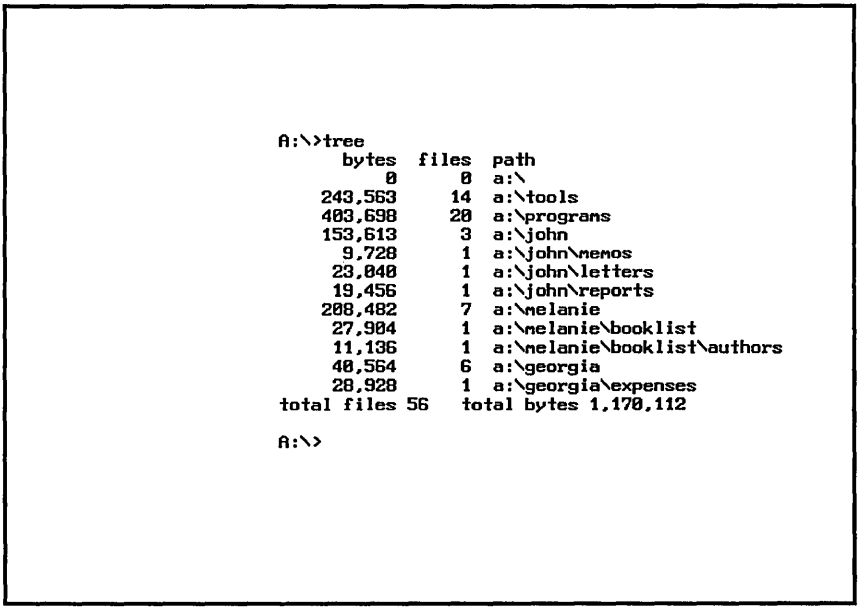
/? or /H	Displays help text.
/B	Displays in brief mode which leaves out the number of files in each directory.
/F	Displays all the files in the subdirectories, as well as the subdirectories themselves.
/G	Displays directory structure in graphical format, that is lines are drawn to show the directory structure.
/P	Pauses after each screenful in a long TREE display.

Examples

The command

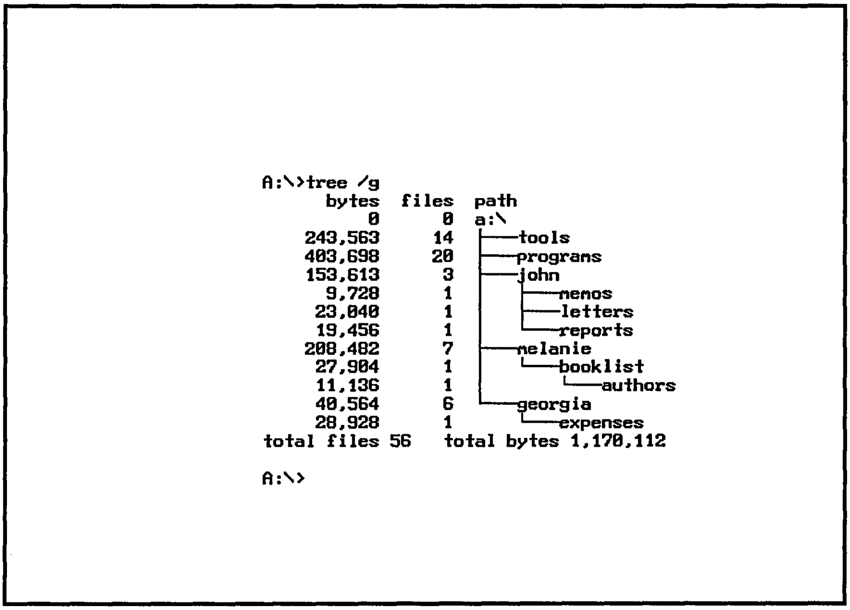
```
A>TREE
```

could generate the listing on the next page which shows that drive A: has five directories: **john**, **georgia**, **melanie**, **programs** and **tools**. It shows that **john**, **georgia** and **melanie** have subdirectories, and lists the space used and number of files belonging to each.



```
A:\>tree
      bytes  files  path
          0      0  a:\
    243,563     14  a:\tools
    403,698     20  a:\programs
    153,613      3  a:\john
      9,728      1  a:\john\memos
     23,040      1  a:\john\letters
     19,456      1  a:\john\reports
    208,482      7  a:\melanie
     27,904      1  a:\melanie\booklist
     11,136      1  a:\melanie\booklist\authors
     40,564      6  a:\georgia
     28,928      1  a:\georgia\expenses
total files 56   total bytes 1,170,112

A:\>
```

The same command with the /G switch added would display:

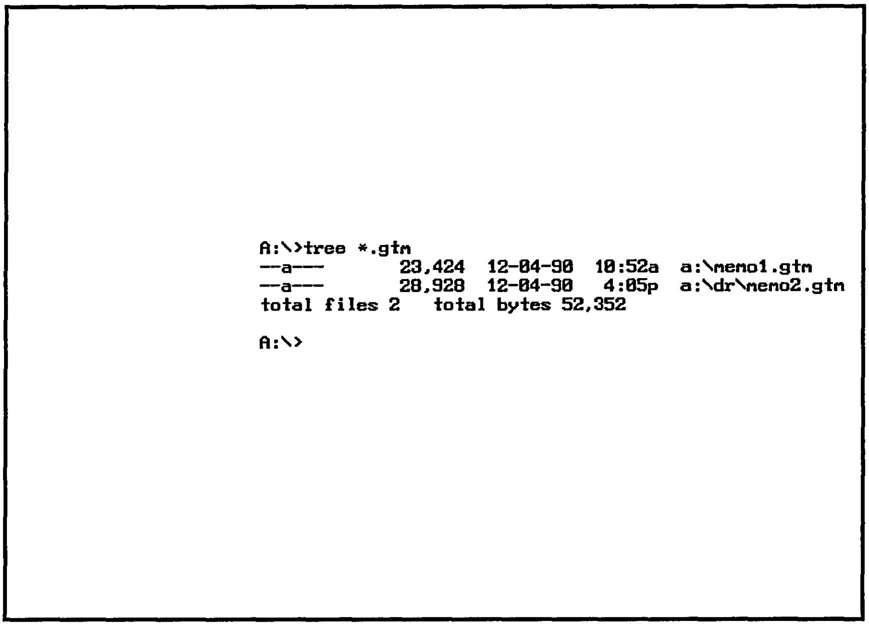


```
A:\>tree /g
      bytes  files  path
          0      0  a:\
    243,563     14  ├───tools
    403,698     20  ├───programs
    153,613      3  ├───john
      9,728      1  │   ├───memos
     23,040      1  │   ├───letters
     19,456      1  │   └───reports
    208,482      7  ├───melanie
     27,904      1  │   └───booklist
     11,136      1  │       └───authors
     40,564      6  └───georgia
     28,928      1      └───expenses
total files 56   total bytes 1,170,112

A:\>
```

The command:

```
A>TREE *.gtm
```

shows two files named ***.gtm** on drive A:. One is in the root directory, and one is in the subdirectory **\dr**.



```
A:\>tree *.gtm
--a---        23,424  12-04-90  10:52a  a:\memo1.gtm
--a---        28,928  12-04-90   4:05p  a:\dr\memo2.gtm
total files 2   total bytes 52,352

A:\>
```

The --a--- represents the file attributes, which are explained under the ATTRIB command.

Note

This form of the TREE command can be very useful when you are trying to find a particular file, but you cannot remember its exact name, or which subdirectory you placed it in.

TYPE

Format

```
TYPE [/?|/H] wildspec[;password] [/P]
```

Explanation

The TYPE command displays the contents of a *text* file on your screen (if you try to display a program or binary file, you just see unintelligible characters).

Press **Ctrl+P** to have the file contents sent to the printer as well as the screen. Press **Ctrl+P** again to stop output to the printer.

Password-protected files

If the file is password-protected for READ mode, you must set the global password, or add the password to the end of the *wildspec*. Wildcard characters can be used to display multiple files. When you type the name of the file you must enter the full name and extension.

Command switches

/? or /H	Displays help text.
/P	Causes the text to be displayed a screenful at a time; move to the next screen by pressing any key. Press **Ctrl+C** to abort the display. Alternatively you can press **Ctrl+S** or **Pause** to stop the display from scrolling. To restart scrolling, press any key.

Examples

This command displays the text file **memo.txt** on the screen a screenful at a time.

```
C>TYPE memo.txt /P
```

This command displays the contents of all files with the extension **.bat**.

```
C>TYPE *.bat
```

UNDELETE

Format

```
UNDELETE [/?|/H] [d:][dirpath][wildspec] [/A] [/B]
[/D:date|/D:-nn] [/L] [/P] [/R:method] [/S] [/T:time]
```

Explanation

Use the UNDELETE command to recover deleted files.

UNDELETE automatically determines how much information about a deleted file exists, and therefore which recovery method to use.

Recovery methods

If the file was saved by DELWATCH, it probably still exists on the disk. UNDELETE uses DELWATCH to recover the file.

If the file was not saved by DELWATCH but DISKMAP was recently run on the disk, and the file has not already been overwritten, UNDELETE uses DISKMAP to recover the file.

Note

Refer to the explanations about DELWATCH and DISKMAP commands in this chapter for more information about DELWATCH and DISKMAP.

If UNDELETE can use neither DELWATCH not DISKMAP, it uses *u*naided recovery. UNDELETE attempts to find the file's original directory entry, allocate the same space on the disk, and update the directory entry and the FAT.

You can use UNDELETE (with options) on the command line or you can use a menu version of the command. Note that only the /B switch applies if you want to use the UNDELETE menu.

Command switches	
/? or /H	Display help text.
/A	Recover all deleted files matching the wildspec but do not prompt before each file.
/B	Display a monochrome (no color) menu. Note that you cannot use any other switch with /B.

Command switches (continued)	
/D:date or /D:-*nn*	Recover (or list) all pending delete files saved by DELWATCH and modified since the specified date, if you specify a date, or files deleted during the last *nn* days, where *nn* is a number.
/L	List all deleted files matching the wildspec but do not recover.
/P	Pause after each screenful.
/R:*method*	Recover only those files with the specified method, where *method* is DELWATCH, DISKMAP, or UNAIDED. If you specify /R:*diskmap*, for example, UNDELETE will only attempt to recover the files for which it can use DISKMAP.
/S	Recover deleted files in subdirectories.
/T:*hh:mm:ss*	Recover (or list) only pending delete files saved by DELWATCH and modified since the specified time.

Using the UNDELETE menu

Type UNDELETE with no filespecs or command switches except /B.

You see a menu on your screen that consists of two elements: a title bar and a file list.

- ❑ The list contains an entry for each deleted file in the current directory, which is indicated at the top of the list. Each list entry displays the filename, size, date, time, and recovery method available for the file. For example:

  ```
  report txt       6,560   11-10-92   11:12am  unaided
  ```

  The list also contains the names of subdirectories in the current directory.

- ❑ The title bar contains the names of three menu options: `File`, `Sort`, and `Help`.

You can exit from the menu at any time by pressing the **Esc** key.

Recovering files using the menu

Select a file to recover by moving the selection bar on the list. Use the up ↑ and down ↓ keys. You can also select a file or a group of files using options in the File menu (refer to the following section, "Using menu options").

Start file recovery for the selected file by doing one of the following:

- ❑ Press **Enter**. Note that if you select a subdirectory in the list, rather than a deleted file, pressing **Enter** moves you down the directory path and displays a list of deleted files in the selected subdirectory.
- ❑ Select the File menu. Refer to the following section, "Using menu options."

If UNDELETE cannot recover the file, you see a message and prompt. Press **Enter** to answer the prompt and return to the UNDELETE menu.

If UNDELETE can recover the file, you also see a message and prompt. Select either of the following responses by using the left ← and right → keys and pressing **Enter**.

- ❑ **OK** to UNDELETE the file.
- ❑ **Cancel** to cancel UNDELETE for the file and return to the UNDELETE menu.

If you choose **OK**, the file is recovered and the file list is updated. RECOVERED replaces the forecast in the list entry.

Using menu options

Select a pull-down menu to open by pressing the **Alt** key with one of the following:

- ❑ **F** for the File menu
- ❑ **S** for the Sort menu
- ❑ **H** for the Help menu

Once a menu is open, you can move between the three menus using the left ← and right → keys.

Close a menu by either selecting an option or pressing **Esc**.

Select any of the following options in an open menu by moving the selection bar and pressing **Enter** or typing the highlighted letter in the option name. You can also select some of the options outside of the menu by pressing **Alt** with the highlighted letter, as indicated next to the option in the menu.

For example, to choose the `Change drive` option, either press **Alt+F** to open the File menu and then **Alt+D** to select `Change drive` or simply press **Alt+D**.

File menu options

Undelete (**Alt+U**)
Starts UNDELETE for the selected file. Refer to the previous section, "Recovering files using the menu" about how UNDELETE recovers a file.

Change **d**rive (**Alt+D**)
Allows you to specify a different disk.

Select (**Space**)
Selects or "tags" the file currently highlighted in the list. You can tag a number of files this way; a diamond (♦) appears beside each file you tag. When you choose the Undelete option, all the tagged files will be recovered at the same time.

Select **g**roup
Allows you to select or tag a group of files using wildcard characters.

Deselect **g**roup
Allows you to deselect or untag a group of files using wildcard characters.

E**x**it (**Alt+X**)
Cancels UNDELETE and returns you to the command prompt.

Sort menu options

Name
Sorts the file into alphabetical order.

Extension
Also sorts the list into alphabetical order, but by file extension.

Date
Sorts the list by file date stamp, from the earliest to the most recent.

Size
Sorts the list by file size, from smallest to largest.

Sort menu options (continued)	
Recovery method	Sorts the list by file recovery method, from best (Delwatch) to worst (Cannot recover).
No sort	Leaves the file list unsorted.

Help menu options	
Help for help	Displays information about using the online help.
Contents	Displays the online help table of contents.
Previous topic	Displays the last help topic that you viewed.
About ...	Displays copyright information.

Using UNDELETE from the command line

Using UNDELETE on the command line is particularly useful, and faster, when you know exactly which files you want to recover and where they were located before deletion.

Listing deleted files

Type UNDELETE followed by a filespec or one or more of the list options (/L, /P, /D, /S, or /T) to list undeleted files without recovering them. UNDELETE displays file information such as filename, size, and available recovery method. Refer to the table of command switches for more information about list options.

Recovering files

Type UNDELETE followed by a filespec to recover files. UNDELETE displays information about the specified files and a prompt before starting file recovery. Type Y to recover the file or N to cancel. Refer to the command switches table for information about using /A, /D, /R, /S, and /T with UNDELETE on the command line.

Examples

The following command would list all the undeleted files in **c:\customer** and all subdirectories of **\customer** and pause after each screen:

```
C>UNDELETE \customer  /S  /P  /L
```

The next command would recover all deleted files in **c:\customer** with extension **.doc** and a recovery method of DELWATCH, without prompting you for each file separately:

```
C>UNDELETE \customer\*.doc /A /R:delwatch
```

The following command is the same as the previous command, except that it would only recover files deleted in the past 10 days:

```
C> UNDELETE \customer\*.doc /A /R:delwatch /D:-10
```

UNFORMAT

Format

```
UNFORMAT [/?|/H] d:
```

Explanation

This command allows you to unformat a disk or diskette that has been "safe formatted" using the DR DOS FORMAT command (see this chapter). When a safe format is performed, a copy of the system information is stored on an unused area of the disk. The UNFORMAT command can locate this system information and use it to restore any files that have not been overwritten since the formatting process. You can only recover all the data from the disk if you have not copied any new files to it. If you have copied files to the disk, UNFORMAT will recover the files that have not been overwritten.

WARNING If you copy lots of files to the disk, the system information may be overwritten and you will not be able to recover any data using UNFORMAT. You will see the message:

```
UNFORMAT information has been overwritten
```

Command switches	
/? or /H	Displays help text.

Example

To unformat a diskette in drive B:, type:

```
C>UNFORMAT b:
```

You see the message:

```
Disk UNFORMATed successfully
```

and the DR DOS prompt is returned. If you now type:

```
C>DIR b:
```

you will see the names of the files that have been recovered.

UNINSTAL

Format

```
UNINSTAL [/?|/H] [/C]
```

Explanation

UNINSTAL is a command you can use to remove DR DOS from your hard disk and restore your previous operating system, whether it is an old version of DR DOS or another system.

In order to use UNINSTAL, however, you must have saved the old operating system by answering the appropriate prompt during the DR DOS INSTALL program. Refer to Chapter 1, "Getting started with DR DOS" for information about INSTALL.

After you use UNINSTAL, you can only reinstall DR DOS by using the system diskettes.

Command switches

/? or /H	Displays help text.
/C	Removes the old operating system files. Note that if you specify /C, you will not be able to "uninstall" DR DOS.

VER

Format

```
VER [/?|/H]
```

Explanation

VER displays the version of DR DOS that you are using.

Command switches	
/? or /H	Displays help text.

VERIFY

Format

```
VERIFY [/?|/H] [ON|OFF]
```

Explanation

`VERIFY ON` causes DR DOS to check that data is written to the disk correctly after each write operation. This, however, causes your system to run slightly slower and so the default value is `VERIFY OFF`.

Entering VERIFY with no parameters shows whether VERIFY is currently ON or OFF.

If you switch VERIFY on, it stays on until you or a program specifically switches it off.

Command switches	
/? or /H	Displays help text.

Examples

To check the current status of VERIFY:

```
C>VERIFY
```

To switch VERIFY off:

```
C>VERIFY OFF
```

VOL

Format

```
VOL [/?|/H] [d:]
```

Explanation

VOL displays the volume label of the specified drive, or of the default drive if no drive is given.

If the disk in the drive has no label then the message:

```
    Volume in drive d does not have a label
```

is shown.

Example

The following command displays the volume label on the current drive C:.

```
C>VOL

Volume in drive C is ACCOUNTS
```

Command switches	
/? or /H	Displays help text.

XCOPY

Format

```
XCOPY [/?|/H] [@]wildspec [dirpath] [/A] [/D:date]
[E] [/H] [/L] [/M] [/P|/C] [/R] [/S] [/V] [/W]
```

Explanation

XCOPY (or Extended Copy) selectively copies groups of files. It can be used to copy whole subdirectories.

The *wildspec* is the drive, path and name of the files to be copied.

Dirpath is the destination drive and path to which the files are to be copied.

Command switches

/? or /H — Displays help text.

/A — Copies only those files that have the archive attribute set. The attribute of the source file is not changed. (See Chapter 4, "Protecting your information", section 4.3 and the ATTRIB command explanation in this chapter, for an explanation of the archive attribute and other attributes).

/D:*date* — Copies only those files whose date is the same or later than the date specified. The format of the date depends on the current country code (see the DATE command).

/E — Creates subdirectories at the destination even if they end up being empty after all the copying is complete.

/H — Includes hidden and system files. (Files with the system attribute are explained in the ATTRIB command description.) The default is for system files to be ignored.

/L — Copies the disk label from one disk to another as well as the files specified.

/M — Copies files that have the archive attribute set and resets the attribute bit of the source file.

Command switches (continued)	
/P or /C	Prompts you to confirm each file before it is copied.
/R	Overwrites read-only files.
/S	Copies the files in the source directory and files in all the subdirectories below it.
/V	Verifies that the data written to the destination is a true copy of data in the source.
/W	Waits for a diskette to be inserted before beginning to search for source files. Press a key to continue or **Ctrl+C** to stop the copying. This allows the XCOPY program to be loaded from diskette, the diskette changed for the source diskette.

Remarks

XCOPY reports the following error codes:

Code	Explanation
0	Normal completion.
1	No files found to copy.
2	Copy process ended by **Ctrl+C**.
4	Copy process ended by initialization; not enough memory or disk space, or invalid drive name.
5	Copy process ended by disk write error.

This feature can be useful in creating batch files with an `IF ERRORLEVEL` statement in them to improve the process, see Chapter 9, "Grouping commands in batch files".

You can use XCOPY to rename files while copying them in the same way as the COPY command, by specifying a new file name for the destination file.

Using XCOPY you can duplicate a tree of subdirectories and files on another disk. If you want an *exact* copy of a tree structure, you need to specify the /E switch as well as the /S switch to ensure all subdirectories including empty ones are copied.

You can use XCOPY to copy a subdirectory from one type of disk to another disk with a different format.

Password-protected files

XCOPY can be used with password-protected files and directories; you are prompted for the password. For example, if the subdirectory **sales** was assigned the password MAGIC, and you entered the following command from the root directory:

```
C>XCOPY c:\sales a:
```

you are prompted:

```
c:\sales ...path password?
```

Type in the password (it is not shown on the screen) and press **Enter**, the contents of the subdirectory **sales** are copied to the diskette in drive A:.

Examples

Suppose you have files in a directory called **c:\wordproc** which contains two subdirectories **\wordproc\jenny** and **\wordproc\olivia**. You want to copy all the files that have the file extension **.let** from these directories on drive C: onto a diskette in drive A:. You also want the appropriate subdirectories to be created on the diskette in drive A:. You type:

```
C>XCOPY \wordproc\*.let a: /E /S
```

The next example copies all the files that are marked for archiving from a diskette in drive A: to another diskette in drive B:. (Note that the archive attribute is set by DR DOS whenever a file is written to and then closed.) You are asked to press a key to start the copying operation. The archive attribute is turned off at the source file to show that the file has been copied. You type:

```
A>XCOPY *.* b: /M /W
```

The following command copies all the files listed in the file **file.lst** stored in the directory **c:\fred** to the directory **d:\john**. Each file specification in **file.lst** must start on a new line, or be separated by a space, tab or comma.

```
C>XCOPY @c:\fred\file.lst d:\john
```

The following command copies all files from a diskette in drive A: to another diskette in drive B: including the diskette label:

```
C>XCOPY a:\*.* b:\ /S /H /L
```

You could use this command to convert an application installation diskette set from 5¼" to 3½" media.

XDEL

Format

```
XDEL [/?|/H] [@]wildspec... [/D] [/N] [/O] [/P] [/R]
[/S]
```

Explanation

XDEL is a command for deleting multiple files in subdirectories and removing empty directories.

Password-protected files

If no default password is given, you are prompted for the correct one before a password-protected file is deleted.

Command switches

/? or /H	Displays help text.
/D	Removes empty subdirectories.
/N	Deletes all the specified files without asking further permission. Use this switch with care.
/O	Overwrite files before deleting them. Normally, when you delete a file, DR DOS amends the directory entry for that file to show the space it previously occupied is now free, but keeps the actual file contents until another file requires its space. It is then overwritten by the new file. If you want to make sure the file is really erased off the face of the disk, you can specify the /O switch.

WARNING Use the /O option with great care as once a file is deleted by it, it can *never* be recovered.

/P	Prompts you before deleting each file.
/R	Delete read-only files.
/S	Deletes files in subdirectories derived from the specified directory.

Example

The following command deletes all files with the **.doc** extension from the current directory and all the subdirectories below the current directory.

```
A>XDEL *.doc /S
```

```
path: a:\
file: *.doc
warning: FILES IN SUB-DIRECTORIES MAY BE DELETED
Is this what you wish to do (Y/N) ? Y
a:\first.doc
a:\sub1\mem1.doc
a:\sub2\mem2.doc
a:\sub3\letter3.doc
```

XDIR

Format

```
XDIR [/?|/H] [+|-ADHRS] [wildspec] [/B] [/C] [/L]
[/N] [/P] [/R] [/S] [/T] [/W] [/X] [/Y] [/Z]
```

Explanation

XDIR is an expanded version of the DIR command, offering extra options for displaying the contents of directories and subdirectories.

Command switches

/? or /H Displays help text.

+ or - A,D,H,R,S Includes (+) or excludes (-) files with a certain attribute. The default is for all the files to be displayed, including system files. See the ATTRIB command for details of the different attributes.

/B Produces a brief directory display showing only filenames and paths.

/C Computes and displays a checksum for each file. A checksum is a four digit hexadecimal number calculated from the bytes within the file. It can be used to verify that two files are the same.

/L Produces a long display format. This is the default.

/N Does not sort directory alphabetically.

/P Pauses after each screenful of display.

/R Reverses the sort order: alphabetical by filename (the default), by extension (set by /X), chronological (set by /T), by file size (set by /Z), or by SuperStor compression ratio (set by /Y).

/S Displays the files in subdirectories below the specified directory.

/T Sorts the directory by date and time, rather than by filename.

Command switches (continued)

/W Displays the directory in a wide, abbreviated format. This is useful if your directory display would otherwise be too long for the screen.

/X Sorts the directory by filename extension, for example, **.exe**, grouping files with the same extension together.

/Y Sorts the directory by SuperStor file compression ratio. /Y only applies if the drive is a SuperStor compressed drive.

/Z Sorts the directory by file size.

The output of the XDIR command is controlled by the order of any sort switches specified. For example, if you type:

```
A>XDIR /Z /X
```

the directory is sorted by file size and then by extension (if two files of the same size are found).

If you type:

```
A>XDIR /X /Z
```

the display is sorted so that all files with the same extension are grouped together and are then listed according to size.

Examples

The following example displays all the read-only files in the current directory and the associated subdirectories:

```
A>XDIR +r *.* /S
```

The following example displays all the files that match the specification **memo*.*** on the current directory, and sorts them into reverse chronological order, so that the most recent are at the top of the list:

```
A>XDIR memo*.* /T /R
```

Customizing your system

YOU may want to adapt DR DOS to improve the efficiency of the programs that you use. DR DOS allows you to make adjustments to some of its standard settings or “defaults” via a program called SETUP.

SETUP has menus of options that are easy to select and change. SETUP takes your changes and when you restart your computer after using it, automatically updates your system for you.

Using SETUP you can alter such settings as:

- ❑ The allocation of computer memory as memory disks.
- ❑ The default country.
- ❑ The number of files that can be open at one time.

When you use SETUP to change default values it automatically updates the DR DOS configuration files **config.sys** and **autoexec.bat** for you. You can directly edit these configuration files using EDITOR if you prefer. This chapter describes how to use SETUP, and the DR DOS configuration files and commands. Note that some advanced features can only be set up by directly editing the configuration files.

11.1 Using SETUP

To start SETUP, type:

```
C>SETUP
```

A welcome screen appears listing all the keys you can use to move through SETUP and select options. The screens are similar to those that appear when you install DR DOS.

SETUP has full context-sensitive help. Just press the **F1** key on a highlighted option or when the cursor is positioned on the field you are interested in.

You can leave SETUP at any time and return to DR DOS by pressing **F10.** This will not do any harm to your existing system setup. You can also go back at any time in the future to change any values set.

11.2 SETUP and the DR DOS configuration files

The SETUP program changes the settings in two files that are specifically used by DR DOS to customize your system to exactly the way you want it. These files are **config.sys** and **autoexec.bat**. Every time DR DOS is started, it searches its root directory for both these files and carries out the commands contained in them.

Config.sys has its own set of commands, which are listed in section 11.3, "Config.sys commands".

When you install DR DOS a **config.sys** file is built up and copied into the root directory of the disk from which you start DR DOS (usually the hard disk).

To see how your **config.sys** file is set up, enter:

```
C>TYPE config.sys
```

You see something like this:

```
SHELL = C:\COMMAND.COM /P /E:1024
DEVICE = c:\DRDOS\EMM386.SYS
BREAK = ON
BUFFERS = 20
FILES = 20
FCBS = 4, 4
FASTOPEN = 512
LASTDRIVE = E
HISTORY = ON, 256, OFF
COUNTRY = 001,,C:\DRDOS\COUNTRY.SYS
```

The display will vary according to your individual system setup.

11.3 Config.sys commands

The **config.sys** commands are explained in detail on the following pages. Examples given under each command show how to set these parameters directly in the **config.sys** file. However, it is much easier to change your system configuration using the SETUP program.

Command conventions

The **config.sys** command descriptions use the same conventions as the DR DOS command descriptions, as detailed in Chapter 10 "DR DOS commands", section 10.1 "Command syntax explained".

Table 11-1 Config.sys commands

Command	Explanation
?	Conditional execution of commands.
:label	Defines a label that can be referenced by a GOTO **config.sys** subcommand.
BREAK	Allows you to set the default state of the BREAK command, which lets you stop a program while it is running.
BUFFERS	Sets the number of disk buffers DR DOS allocates.
CHAIN	Transfers control to another configuration file.
CLS	Clears the display screen.
COUNTRY	Selects the appropriate date and time format, currency symbol, and code page parameter for your country.
CPOS	Positions the cursor at a specified location on the screen.
DEVICE	Enables extra device drivers to be installed.
DRIVPARM	Sets the physical characteristics for a specified disk drive.
ECHO	Allows the display of messages on screen during startup.

Command	Explanation
EXIT	Forces the operating system to terminate **config.sys** execution.
FASTOPEN	Increases the speed with which files are accessed when the same files are repeatedly used.
FCBS	Sets the number of files that can be opened by older applications at the same time.
FILES	Sets the number of files that can be opened by programs at the same time.
GOSUB	Directs the operating system to jump to a label during execution and then jump back again (with a RETURN statement).
GOTO	Changes the order in which commands are executed by directing the operating system to jump to a label.
HIBUFFERS	Sets the number of disk buffers and attempts to allocate them in high memory.
HIDEVICE	Enables device drivers to be installed in upper memory.
HIDOS	Relocates as much as possible of the DR DOS operating system data into upper memory.
HIINSTALL	Installs specified programs into upper memory during startup.
HISTORY	Provides the extended line editing facility.
INSTALL	Allows you to load a program during startup.
LASTDRIVE	Sets the maximum number of drives to be accessed.
REM	During batch file execution, displays the comment given on the same line as REM.
RETURN	Directs the operating system to jump back again after a GOSUB or SWITCH command, during execution.
SET	Inserts environment variables into the command processor's master environment.
SHELL	Specifies the location of **command.com**, or loads a different command processor.
SWITCH	Allows switching between different configuration routines with the same **config.sys** file.

Command	Explanation
TIMEOUT	Forces the operating system to ignore a ? or SWITCH command if a prompt is not answered within a specified time period.

?

Format

```
? ["remark"] command
```

Explanation

If you insert a ? at the beginning of a statement in your config.sys file, you are asked if you want that particular statement executed, when **config.sys** is executed. This is most useful when applied to the DEVICE= and CHAIN= statements so you can choose whether to install specific device drivers or transfer control to a different configuration file at startup. For example, the following statement in your **config.sys** file:

```
?DEVICE = C:\DRDOS\ANSI.SYS
```

means you are queried each time you reboot your system as to whether or not you want the device driver ANSI.SYS loaded, like this:

```
     DEVICE =C:\DRDOS\ANSI.SYS (Y/N) ?
```

Type Y to load ANSI.SYS, or N to ignore it. A response is required, unless you set a limit on the amount of time allowed for a response and ignore the command if none is given. Use the TIMEOUT command in **config.sys** to specify a time limit for a ? statement.

You can place a message string between the ? and the **config.sys** statement, enclosed in double quotation marks. For example:

```
?"Do you want ANSI.SYS?"DEVICE = C:\DRDOS\ANSI.SYS
```

Then, at startup, you are prompted with the message:

```
     Do you want ANSI.SYS?
```

Answer by typing Y for Yes or N for No. This can be convenient if you have access to a network; you can set up a prompt that asks you at startup whether you want to use the network.

You can use ? with any of your **config.sys** file statements.

The maximum length allowed for a command statement including a question mark and message string is 128 characters. Any statements longer than this are ignored.

:label

Format

```
:label
```

Explanation

:label defines a label that can be referenced by a GOTO **config.sys** subcommand.

Only the first eight characters after the colon are considered part of the label by GOTO.

BREAK

Format

```
BREAK = ON|OFF
```

Explanation

DR DOS allows you to stop a program when it is running by holding down the **Ctrl** key and pressing **Break** or **C**.

If BREAK is OFF (the default), the key sequence **Ctrl+Break** or **Ctrl+C** will only stop the program when it next reads from the keyboard, writes to the screen or writes to a printer.

When BREAK is ON, DR DOS also looks for **Ctrl+Break** or **Ctrl+C** when the program is reading to or writing from a disk.

BREAK=ON lets you quickly stop programs that do not use standard input and output operations very often, such as programs that perform lengthy calculations before printing the final output.

The BREAK command in the **config.sys** file can be overruled when you type a BREAK command at the command line. (See the BREAK command entry in Chapter 10, "DR DOS commands".)

Note

Many applications disable this feature so that you can only use the method documented by the application to stop it.

BUFFERS

Format

```
BUFFERS = nn
```

Explanation

Buffers are small areas (or blocks) of memory that DR DOS uses during disk read or write operations. Buffers store the information that is being read or written. You can specify between 3 and 99 buffers; the default is 15 (if you have a 640 Kbyte system).

The number of buffers you specify generally affects the speed at which a program accesses files. The more buffers there are, the faster the system accesses files. However, each buffer occupies memory that would otherwise be used for the program itself and for your data. If you specify a large number of buffers, your program may actually run slower than if you specify a small number.

Your application documentation should suggest how many buffers you need. Between 10 and 30 is a sensible number; a few more may be needed if the program uses a large number of files. A desktop publishing program may need between 20 and 30, for example.

You can also force DR DOS to allocate your buffers in high memory, rather than conventional memory, by using HIBUFFERS (see the HIBUFFERS command).

Example

The following entry in your **config.sys** file sets the number of buffers to 20:

```
BUFFERS = 20
```

CHAIN

Format

```
CHAIN = filespec
```

Explanation

The CHAIN command transfers control to another configuration file. CHAIN checks that the filespec specified exists, if it does, it closes the current configuration file, and begins executing the statements in the filespec. It does not return to the original configuration file.

The CHAIN command is most useful for systems with read-only or ROM disks. It allows the user to change the default configuration specified in the ROM-based copy of the **config.sys** file.

CHAIN is also useful when used in conjunction with the ? conditional option (see the entry for ?). Using it, you can group together several similar **config.sys** commands into additional **config.sys** files, and execute them with one conditional statement.

Caution When you use the CHAIN command to transfer control to a configuration file on a different drive, you must specify the full path *including* the drive in all DEVICE, and HIDEVICE statements.

Example

The following CHAIN command within a **config.sys** file means that when DR DOS reaches the CHAIN statement it will look for a **config.sys** file on drive D: and, if it exists, will begin to process the commands in that configuration file instead. If the **config.sys** file on drive D: does not exist, the original **config.sys** file continues to be processed.

```
BUFFERS  =  10
COUNTRY = 44,,C:\DRDOS\COUNTRY.SYS
CHAIN = D:\CONFIG.SYS
```

CLS

Format

```
CLS
```

Explanation

The CLS command clears the display screen.

COUNTRY

Format

```
COUNTRY=nnn,cp,[d:]\path\COUNTRY.SYS
```

Explanation

The **country**.sys file provides the date and time format and currency symbol appropriate for your country. It also includes code page information which can be specified in the second parameter (*cp*). For European countries, for example, the preferred date format is day/month/year, whereas for North America it is month/day/year and for Japan, year/month/day. In the United Kingdom the currency symbol is £ rather than $ as used in the United States.

The default value may already be set for your own country and may not need to be changed.

The country code does not include information about the keyboard type; that information can be provided using the KEYB command, see Chapter 10, "DR DOS commands".

The **country.sys** file affects the commands BACKUP, DATE RESTORE, and TIME.

Options

nnn The country code of your country. Table 11-2 shows the values allowed.

cp The code page for which country information is required. If this parameter is left out, the default code page for the country *nnn* is assumed.

The first code page listed is the default code page for the country, see Table 11-2.

Table 11-2 Country Codes and Code Pages

Country	Code (nnn)	Code Pages (cp)
Australia	061	437 850
Belgium	032	437 850
Canada (French)	002	863 850
Denmark	045	865 850
Finland	358	437 850
France	033	437 850
Germany (West)	049	437 850
Hungary	036	852 850
Israel	972	862 850
Italy	039	437 850
Japan	081	932 437
Korea	082	934 437
Latin America	003	437 850
Middle East	785	864 850
Netherlands	031	437 850
Norway	047	865 850
Portugal	351	860 850
Russia	007	866 850
Spain	034	437 850
Sweden	046	437 850
Swiss (French)	041	437 850
Swiss (German)	041	437 850
Turkey	090	857 850
United Kingdom	044	437 850
United States	001	437 850

Example

This entry in your **config.sys** file converts international currency symbols, date and time to Italian formats using code page 437.

```
COUNTRY = 039,437,C:\DRDOS\COUNTRY.SYS
```

CPOS

Format

```
CPOS nn, nn
```

Explanation

Use CPOS to reposition the cursor to a specified location on the screen.

The first parameter *nn* is the row number, to specify the vertical position of the cursor. Enter a number from 1 through 25.

The second parameter *nn* is the column number, to specify the horizontal position of the cursor. Enter a number from 1 through 80.

Example

The following command causes the cursor to appear on the screen in row 10, column 10:

```
CPOS 10, 10
```

DEVICE

Format

```
DEVICE = filespec [options]
```

Explanation

The DEVICE command enables device drivers to be installed. Device drivers for the keyboard, screen, disk drives, printer and auxiliary devices are automatically loaded as part of the DR DOS startup sequence.

Whenever you add a new piece of computer hardware to your system like a printer or mouse, you should be supplied with a device driver if one is required. Device drivers are programs that are loaded alongside DR DOS to operate hardware devices that are not recognized as a default part of your system. Device drivers usually have the extension **.sys**. You need to copy the supplied device driver onto the disk from which you start DR DOS. Then use the DEVICE command in **config.sys** to ensure the driver is found and loaded each time DR DOS is started.

A number of optional device drivers are supplied with DR DOS and can be loaded with the DEVICE command:

ANSI.SYS
DISPLAY.SYS
DRIVER.SYS
EMM386.SYS
EMMXMA.SYS
HIDOS.SYS
PRINTER.SYS
VDISK.SYS

These drivers are described in section 11.4, "Installable DR DOS device drivers".

If you precede a DEVICE statement with a question mark before the device driver is loaded, you are prompted as to whether you want it installed. This gives you the option every time you start DR DOS of omitting a particular device driver.

Most device drivers are hardware-specific, but some (like VDISK.SYS) are designed for software functions; to set up a memory disk on your system in the case of VDISK.SYS.

Example

The following statement installs the device driver ANSI.SYS.

```
DEVICE = C:\DRDOS\ANSI.SYS
```

DRIVPARM

Format

```
DRIVPARM = /D:d [/C] [/F:f] [/H:h] [/N] [/S:ss]
[/T:tt]
```

Explanation

DRIVPARM sets the physical characteristics for a specified disk drive. It lets you associate a particular type of physical disk drive with a logical drive. For example, some computers do not expect to use 3½" drives, so to add a 3½" drive it may be necessary to use DRIVPARM to redefine the drive parameters for subsequent use by DR DOS.

DRIVPARM is similar to the device driver DRIVER.SYS, except it only allows you to modify a logical drive already known to DR DOS. If you want to add a *new* logical drive, you must use DRIVER.SYS, as described in section 11.4, "Installable DR DOS device drivers".

Options

/D:*d* A single digit 0, 1, 2 and so on, which corresponds to the logical drive (for example A: or B:) for which the characteristics are to be set.

/C Indicates that the drive is able to detect when a disk is changed in the drive.

/F:*f* Specifies the drive type. The possible values are 0, 1, 2 and 7, where 0 = 360 Kbyte drive, 1 = 1.2 Mbyte drive, 2 = 720 Kbyte drive and 7 = 1.44 Mbyte. The default is 2.

/H:*h* Specifies the number of drive heads. Possible values are 1 and 2.

/N Indicates that the drive is for permanent media.

/S:*ss* Specifies the number of disk sectors that the drive supports. Possible values can be between 1 and 63.

/T:*tt* Specifies the number of tracks that the drive supports. Possible values are 40 and 80.

ECHO

Format

```
ECHO [=][message]
```

Explanation

ECHO allows the display of messages on screen during startup.

It is useful for including questions and explanatory messages as part of your **config.sys** file, and these messages are displayed at startup in the order they are inserted in the **config.sys** file. The messages are displayed exactly as you type them in the ECHO statements; for example, if you leave a space between = and the string, the message appears indented one space.

Example

```
ECHO = Install code page switching?
```

would display the following line of text during startup:

```
    Install code page switching?
```

EXIT

Format

```
EXIT
```

Explanation

The EXIT statement forces the operating system to terminate **config.sys** execution.

FASTOPEN

Format

```
FASTOPEN = nnnnn
```

Explanation

FASTOPEN reduces the time taken to access files on a hard disk which are being accessed repeatedly. It does this by remembering the locations of the files on disk, so that DR DOS does not have to navigate along the path every time the file is opened.

DR DOS sets up a special table of these filenames in memory and you can set the size of this table by specifying the number of entries.

The default number is 512. The minimum number is 128 and the maximum is 32768.

This command improves disk performance at the cost of memory space. Each table entry uses 2 bytes of memory. Having a larger table than you actually need will not increase your system's performance.

Example

This command sets up a table size of 512 entries:

```
FASTOPEN = 512
```

FCBS

Format

```
FCBS=m,n
```

Explanation

This command sets the number of files that can be opened by programs at the same time, using data structures called File Control Blocks. FCBS have little effect unless the file sharing program SHARE.EXE (see Chapter 10, "DR DOS commands"), or networking support is loaded.

Options

m The number of files that FCBs can open at once in the range 1 to 255.

n The number of files opened by FCBs that are protected from automatic closure by DR DOS in the range 1 to 255. (If an application attempts to open more than *m* files by FCBs, then all but the first *n* files may be closed by DR DOS.)

Some older style applications may require you to insert an FCBS statement in **config.sys** rather than a FILES entry. (The configuration command FILES can also be used to control the number of files that can be opened simultaneously.) Unless your application documentation specifically requests you to have an FCBS entry, use FILES to control the number of open files allowed.

Example

This command sets the number of open files allowed to 8 and the number of protected files to 4.

```
FCBS = 8,4
```

FILES

Format

```
FILES = nnn
```

Explanation

This command specifies the number of files that can be opened by programs at the same time. The default is 20, in the range 20 through 255. DR DOS uses this number to determine how much memory to set aside for controlling open files.

You should only increase the default if your application documentation instructs you to, or if the program you are using returns error messages that the number of open files is insufficient.

Example

The following entry in your **config.sys** file sets the number of open files allowed to 20.

```
FILES = 20
```

GOSUB

Format

```
GOSUB label
```

Explanation

GOSUB performs the same function as GOTO, except that you use GOSUB to jump to a statement or *group* of statements (*subroutine*), execute them, and then jump back again with a RETURN statement. The *label* must be a label present in **config.sys** and identify a statement or subroutine. You must also use a RETURN statement with GOSUB to identify the end of the subroutine; execution then returns to the statement immediately following the GOSUB.

Example

In the following sequence, the operating system switches to the statement after the label, executes the following statements, and then returns to the statement after the GOSUB command:

```
GOSUB memdisk

exit

.

.

.

:memdisk

DEVICE=vdisk.sys

.

.

.

return
```

GOTO

Format

```
GOTO label
```

Explanation

GOTO is used to change the order in which commands are executed.

GOTO directs the operating system to jump to another part of **config.sys** identified by a label. GOTO is often used with a ? statement, so that if a condition is satisfied, execution jumps. The label parameter must identify a label that is present in the **config.sys** file.

To direct execution to a statement or subroutine within **config.sys** and then jump back again, use the GOSUB and RETURN statements (see GOSUB and RETURN).

To direct the operating system to jump to the end of the **config.sys** file, use the EXIT statement (see EXIT).

Example

In the following sequence, if the user types Y for Yes, the operating system switches to the command after the label, and loads **vdisk.sys**:

```
? "Do you want a memory disk? (Y/N)" GOTO memdisk
.
.
.
:memdisk
DEVICE=vdisk.sys
```

HIBUFFERS

Format

```
HIBUFFERS = nn
```

Explanation

Use HIBUFFERS to specify the number of memory buffers that DR DOS uses. HIBUFFERS performs the same function as BUFFERS, except that HIBUFFERS allocates as many buffers as possible from high memory rather than conventional memory. If there is insufficient high memory available for all the buffers specified, the rest are created in upper memory (if HIDOS is ON) or conventional memory (if HIDOS is OFF); see HIDOS.

Note that you must have a device driver that supports high memory (XMS) loaded before you can use HIBUFFERS. See Appendix C, "Memory" for information about high memory.

Example

The following entry in your **config.sys** file sets the number of buffers to 20 and forces DR DOS to allocate as many as possible from high memory:

```
HIBUFFERS = 20
```

HIDEVICE

Format

```
HIDEVICE [SIZE=nnn] filespec
```

Explanation

The HIDEVICE command loads specified device drivers into upper memory. If there is insufficient upper memory for the specified driver, it is loaded into conventional memory. Before you can use HIDEVICE, a device driver that supports upper memory must be loaded. See section 11.4, "Installable DR DOS device drivers" and Chapter 12, "Using MemoryMAX" for descriptions of device drivers and memory management.

You can specify HIDEVICE with the SIZE option; in this case, HIDEVICE does not attempt to load the device driver into upper memory unless the amount of space you specify (at least) is available.

You can use the MEM command with the /D switch (see Chapter 10, "DR DOS commands") to check whether there is enough upper memory available to load the device driver, then add the HIDEVICE command to **config.sys** and reboot your system. Alternatively, you can use MEM after **config.sys** has been executed to check whether there was enough space to load the device driver in upper memory.

Options	
filespec	The device driver you want to load into upper memory.
SIZE=*nnn*	The minimum number of bytes in upper memory that must be available before HIDEVICE loads the device driver, where *nnn* is the number of bytes (in hexadecimal format).

The following DR DOS device drivers, described in section 11.4, "Installable DR DOS device drivers", are suitable for loading into upper memory:

ANSI.SYS
DISPLAY.SYS
PRINTER.SYS
VDISK.SYS

Device drivers that require more memory than the device driver image should not be loaded into upper memory without using the SIZE option, because they cannot determine the amount of free memory available for their use. If the device driver is loaded in lower memory, the MEM /D command will indicate the size value to use (see the MEM command description in Chapter 10, "DR DOS commands").

Note

You cannot use HIDEVICE if a third-party memory manager has been loaded, because these memory managers tend to use a different technique to DR DOS for mapping upper memory.

Example

The following command loads the ANSI.SYS device driver into upper memory:

```
HIDEVICE =C:\DRDOS\ANSI.SYS
```

HIDOS

Format

```
HIDOS = ON|OFF
```

Explanation

The HIDOS command relocates as much of the DR DOS operating system data structures as possible into upper memory. The default is OFF.

Before you can use HIDOS, a device driver that supports upper memory must be loaded. See section 11.4, "Installable DR DOS device drivers" and Chapter 12, "Using MemoryMAX" for descriptions of device drivers and memory management.

Note

You cannot use HIDOS if a third-party memory manager has been loaded, because these memory managers tend to use a different technique to DR DOS for mapping upper memory.

HIINSTALL

Format

```
HIINSTALL = filespec [options]
```

Explanation

The HIINSTALL command loads specified TSRs into upper memory. If there is insufficient upper memory for the specified TSR, it is loaded into conventional memory. Before you can use HIINSTALL, a device driver that supports upper memory must be loaded. See section 11.4, "Installable DR DOS device drivers" and Chapter 12, "Using MemoryMAX" for descriptions of device drivers and memory management.

You should consider loading the following DR DOS programs into upper memory:

CURSOR.EXE
GRAPHICS.COM
GRAFTABL.COM
JOIN.EXE

Options indicates the options to pass to the program given in the *filespec*.

You can use the MEM command with the /B switch (see Chapter 10, "DR DOS commands") to check whether there is enough upper memory available to load the TSR, then add the HIINSTALL command to **config.sys** and reboot your system. Alternatively, you can use MEM after **config.sys** has been executed to check whether there was enough space to load the TSR in upper memory.

Note

You cannot use HIINSTALL if a third-party memory manager has been loaded, because these memory managers tend to use a different technique to DR DOS for mapping upper memory.

HIINSTALL performs the same function as HILOAD, but loads TSRs while **config.sys** is being processed.

Example

The following command installs the CURSOR program into upper memory:

```
HIINSTALL =C:\DRDOS\CURSOR.EXE
```

Notice that you must give the full path to the program and its extension (**.com** or **.exe**).

HISTORY

Format

```
HISTORY = ON[,nnnn[,ON|OFF]]|OFF
```

Explanation

DR DOS provides an extended command line editing facility. This facility stores commands you previously issued in a memory buffer. You can then recall any previously entered commands and re-issue or modify them. If HISTORY is OFF the extended command line editing is disabled. If HISTORY is ON you can use the function keys described in Chapter 10, "DR DOS commands", section 10.2 to recall and modify the command lines.

Options

nnnn

Allows you to specify the size of the memory buffer in which the command lines are stored. The size of the memory buffer determines how many command lines you can recall. The minimum size you can specify is 128 bytes; the maximum size is 4096 bytes. If you do not specify one, a default size of 512 bytes is used.

works with 256, not 128

Note

The system uses two memory buffers of the size you specify. The first memory buffer stores commands entered from the command line. The second memory buffer stores commands entered while running an application.

[,ON|OFF]

This sets the insert mode on or off. The default is off. If insert mode is on, characters are inserted in front of the cursor. If it is off, characters at the cursor position are overwritten.

Example

To switch on insert mode and set a memory buffer of 512 bytes:

```
HISTORY = ON,512,ON
```

INSTALL

Format

```
INSTALL = filespec [options]
```

Explanation

The INSTALL command allows you to load a program during startup. You can install the following in this way:

CURSOR.EXE
GRAFTABL.COM
GRAPHICS.COM
KEYB.COM
NLSFUNC.EXE
PRINT.COM
SHARE.EXE

Options indicates the options to pass to the program given in the *filespec*.

Example

The following INSTALL statement in a **config.sys** file, loads the KEYB program during **config.sys** processing.

```
BUFFERS = 10
INSTALL = C:\DRDOS\KEYB.COM UK
FILES = 20
```

LASTDRIVE

Format

```
LASTDRIVE = drive_letter
```

Explanation

DR DOS uses letters to identify particular drives. It automatically assigns a drive letter to each *actual* drive that it finds during the startup procedure (including any virtual disk drives). It then sets the *last drive* as being the drive letter assigned to the last drive it finds. All the remaining drive letters then become invalid.

The LASTDRIVE command enables you to set a *different* last drive, so that DR DOS will recognize non-assigned drive letters.

Example

Suppose your computer has a diskette drive (drive A:), and a hard disk drive (drive C:). You have added two virtual disks (drives D: and E:) and you have a program that requires you to use the SUBST command with drives F: and G:. To use LASTDRIVE type LASTDRIVE followed by the highest drive letter you will need:

```
LASTDRIVE = g
```

This means that DR DOS recognizes drives F: and G: as well as drives A: through E:.

REM

Format

```
REM | ; [comment]
```

Explanation

REM allows comments to be inserted into a **config.sys** file (REM is short for REMARK). REM can be replaced by a semicolon (;).

Use REM with a comment to clarify what a command does, or without a comment to add spacing for readability.

Examples

To clarify what an entry does:

```
REM * Load the ANSI driver for Wordsmaster
DEVICE = ANSI.SYS
```

RETURN

Format

```
RETURN
```

Explanation

RETURN is always used with a GOSUB or a SWITCH statement, to direct execution of **config.sys** back to the statement following the GOSUB or SWITCH command. See the GOSUB and SWITCH explanations for more information.

SET

Format

```
SET name=parameter
```

Explanation

The SET statement inserts environment variables into the command processor's master environment. The variable *name* is set depending on the different commands executed and drivers loaded during system configuration. Applications and batch files such as **autoexec.bat** can then test variable values and process commands accordingly.

Example

The following series of statements includes a SET statement that specifies a value for the VDISK variable:

```
?"Create a 1Mb RAM Disk" gosub ramdisk
exit
.
.
.
:ramdisk
set VDISK=1024
device=c:\drdos\vdisk.sys 1024 /X
return
```

The next series of statements is from the **autoexec.bat** file, and test the value of VDISK and change the location of TEMP according to the value:

```
SET TEMP=C:\TEMP
if NOT "%VDISK%" == " " SET TEMP=D:\TEMP
MD %TEMP% >NUL
```

SHELL

Format

```
SHELL = filespec [/E:nnnnn] [/P[:filename]]
[/ML|/MH|/MU]
```

Explanation

Using SHELL you can specify what options the default DR DOS command processor **command.com** will use and where it should be loaded from. (SHELL must be used if **command.com** is located in a directory other than the root.) The command processor or "shell" as it is also known, carries out user commands and starts programs.

You can also use SHELL to load a different command processor. (System programmers can write their own command processors and then use SHELL to load them.)

Some of the most useful options to use with SHELL are listed below. See also the explanation of **command.com** in Chapter 10, "DR DOS commands".

Options

/E:*nnnnn* Specifies the size of the environment in bytes where *nnnnn* is in the range 512 (minimum) to 32,751 maximum). If you specify a value outside this range, the command defaults to the minimum value.

/P[:*filename*]] /P fixes this copy of the command processor permanently in memory. The EXIT command is disabled.

/P also causes **autoexec.bat** file to be executed when **command.com** starts. Specify a *filename* with /P to execute a file other than **autoexec.bat**.

The /P switch is mandatory if a SHELL statement is used, so that an EXIT command from the default shell **command.com** will not exit from the **command.com** and halt the system.

Options (continued)

/ML I /MH I /MU	Forces **command.com** to be loaded in a specific area of memory; /ML for conventional (lower) memory, /MH for high memory, or /MU for upper memory. Note that if there is insufficient high or upper memory available, the command processor is loaded into conventional memory. Before you can use /MH or /MU, a device driver that supports upper or high memory must be loaded.
/R	Loads **command.com** into high or upper memory, rather than conventional memory. You must have a device driver that supports upper or high memory loaded before you can use /R.

Example

By default, **command.com** is located in the root directory, and the environment variable COMSPEC, which defines the path to the command processor, is set to C:\COMMAND.COM.

The following command loads **command.com** from directory **c:\drdos**, and sets COMSPEC=C:\DRDOS\COMMAND.COM:

```
SHELL=C:\DRDOS\COMMAND.COM C:\DRDOS /P /E:512
```

SWITCH

Format

```
SWITCH label1, label2[, labeln]
```

Explanation

Use the SWITCH statement to allow switching between different configuration routines within the **config.sys** file.

SWITCH causes the operating system to prompt you to select a configuration subroutine. Execution then jumps to the selected subroutine.

Each subroutine in the **config.sys** file must start with a label and end with a RETURN statement.

You can specify up to 9 different subroutine labels.

Example

The following series of statements specifies a SWITCH between three subroutines:

```
echo = CONFIGURATION 1
echo = CONFIGURATION 2
echo = CONFIGURATION 3
echo =Please select CONFIGURATION 1 2 or 3 ?
switch config1, config2, config3
echo CONFIGURATION COMPLETE
exit
:config1
echo CONFIGURATION 1 selected
return
:config2
echo CONFIGURATION 2 selected
return
:config3
echo CONFIGURATION 3 selected
return
```

When executed, the statements would cause the following to display:

```
    CONFIGURATION 1

    CONFIGURATION 2

    CONFIGURATION 3

Please select CONFIGURATION 1 2 or 3 ?
```

The valid responses to the SWITCH prompt are:

□ 1 through 9, to select a CONFIGURATION (1 through 9)

□ Just the **Enter** key to automatically select CONFIGURATION 1

□ No response (but with a TIMEOUT condition specified) to automatically select CONFIGURATION 1 after the TIMEOUT limit is reached. (See the TIMEOUT command).

Assuming you type 2 and press **Enter**, you then see:

```
    CONFIGURATION 2 selected
```

Then:

```
    CONFIGURATION COMPLETE
```

TIMEOUT

Format

```
TIMEOUT [=] n
```

Explanation

Use TIMEOUT to set a time limit for the amount of time allowed to respond to a ? or SWITCH prompt (see the ? and SWITCH explanations); the command is then ignored (for ?) or the first option is assumed (for SWITCH) if you do not enter a response within the TIMEOUT limit.

Specify *n* seconds, where *n* is a number, or accept the default of 0, which forces the system to wait indefinitely until a key is pressed.

Note that a TIMEOUT setting applies to all ? and SWITCH statements that follow it.

Example

The following statement sets a time limit of 10 seconds for ? and SWITCH prompts:

```
TIMEOUT 10
```

11.4 Installable DR DOS device drivers

This section describes the additional device drivers supplied with DR DOS; they are summarized in Table 11-3.

Table 11-3 Installable Device Drivers

Device driver	Explanation
ANSI.SYS	Provides extra options for the keyboard and screen.
DEVSWAP.COM	Changes drive assignments for SuperStor compressed drives.
DISPLAY.SYS	Enables code page switching to be used on EGA and VGA displays.
DRIVER.SYS	Sets the characteristics to be associated with a physical disk drive specified in the command line.
EMM386.SYS	Enables applications to use LIM 4.0 expanded memory on Intel 386 and i486-based computers. Allows TSRs, device drivers and operating system data structures to be loaded into upper memory. Relocates the BDOS to upper or high memory.
EMMXMA.SYS	Enables applications to use LIM 4.0 expanded memory on computers with XMA-compatible memory cards.
HIDOS.SYS	Supports the relocation of the DR DOS kernel to high memory on 80286-based computers, and allows computers based on certain chipsets to access a portion of upper memory.
PRINTER.SYS	Enables code page switching on a printer.

Table 11-3 Installable Device Drivers (continued)

Device driver	Explanation
SSTORDRV.SYS	Enables the SuperStor Block Driver, to attach compressed partitions and allocate drive letters. Refer to Chapter 13, "Optimizing your disk" for information about the SuperStor disk compression program.
VDISK.SYS	Sets aside a portion of RAM memory to act as if it was a very fast disk.

Note

The EMM386.SYS driver must always be loaded first and the VDISK.SYS driver after it. If you are installing the SuperStor program, SSTORDRV.SYS must be loaded next. The INSTALL and SETUP programs set this up automatically.

ANSI.SYS

ANSI.SYS is a device driver for the keyboard and screen. It provides extra options for programs that need to move the cursor, alter the screen display, or assign keyboard use. These options take the form of ANSI escape sequences which all start with the [Esc] character (1B hex). The ANSI.SYS driver can only detect escape sequences sent via the normal DOS function calls, and not character output performed using the ROM BIOS directly.

Certain commercial applications that you run on your computer may expect ANSI.SYS to be present; refer to the application's documentation.

Loading ANSI.SYS

For programs to use ANSI escape sequences, the ANSI.SYS driver must be loaded. You can use the SETUP or INSTALL programs to do this automatically. Alternatively you can place the following command in the **config.sys** file:

```
DEVICE = c:\drdos\ansi.sys
```

This ensures that ANSI.SYS is loaded each time DR DOS is started.

The general form of the ANSI escape sequence is:

- ❑ A one byte ESC character (hexadecimal 1BH).
- ❑ A one byte [character (hexadecimal 5BH).
- ❑ One or more decimal numbers separated by semicolons.
- ❑ A single command character.

Creating an escape sequence

To create an ANSI escape sequence you need a word processor or editor, such as EDITOR, that allows you to input the ESC and [control codes. For example, in EDITOR, the sequence **Ctrl+P** followed by the **Esc** key creates the ESC code.

Below is a list of the ANSI escape sequences, divided into the following types:

- ❑ Cursor positioning.
- ❑ Erasing.
- ❑ Changing the display mode.

In the sequences, note that:

1. *y*, *x* indicate row and column numbers, which range between 1 and 50 rows (with 1 at the top of the screen) and 1 and 79 columns (1 at the left of the screen). The numbers are specified in ASCII digits. If you omit the value (or specify zero), the default value is used.
2. Spaces are shown only to make the sequences clearer to read.

Escape sequences for positioning the cursor

Cursor position (CUP) ESC [*y*;*x* H

Horizontal and vertical position (HVP) ESC [*y*;*x* f

CUP and HVP position the cursor according to the co-ordinates issued. The default (and null) value is the top left corner of the screen. CUP and HVP are equivalent.

Cursor up (CUU) ESC [*y* A

Moves the cursor up y rows without changing columns. The sequence is ignored if the cursor is already at the top line.

Cursor down (CUD) ESC [*y* B

Moves the cursor down y rows without changing columns. The sequence is ignored if the cursor is already at the bottom line.

Cursor forward (CUF) ESC [*x* C

Moves the cursor to the right without changing rows. The sequence is ignored if the cursor is already in the far right column.

Cursor backward (CUB) ESC [*x* D

Moves the cursor to the left without changing rows. The sequence is ignored if the cursor is already in the far left column.

Device status report (DSR) ESC [6 n

Cursor position report (CPR) ESC [*y*;*x* R

The DSR sequence instructs the console driver to return the CPR sequence. The CPR sequence reports the cursor position in terms of row and column. It is returned via the keyboard input.

Save cursor position (SCP) ESC [s

Restore cursor position (RCP) Esc [u

SCP stores the current cursor position. RCP restores the cursor to that position. The default for RCP, if no SCP was issued, is 0,0. These escape sequences cannot be nested.

Escape sequences for erasing

Erase display (ED) ESC [2 J

Erases the display and moves the cursor to the top left of the screen.

Erase line (EL) ESC [K

Erases at the cursor position and to the end of the row.

Escape sequences for changing the display modes

Set graphics rendition (SGR) ESC [*n* ; ... ; *n* m

Sets various screen modes which remain until a different SGR sequence is set; screen modes (*n*) are shown in Table 11-4.

Table 11-4 Screen Modes

Parameter (n)	Description
0	Return to default (white characters on black background)
1	High intensity (bold)
4	Underscore (monochrome only)
5	Flashing
7	Inverse video (black characters on white background)
8	Invisible (black on black)
30	Black foreground
31	Red foreground
32	Green foreground
33	Yellow foreground
34	Blue foreground

Table 11-4 Screen Modes (continued)

Parameter	Description
35	Magenta foreground
36	Cyan foreground
37	White foreground
40	Black background
41	Red background
42	Green background
43	Yellow background
44	Blue background
45	Magenta background
46	Cyan background
47	White background

Set screen mode (SM) ESC [= *n* h 1 *or* ESC [= h 1 *or*
ESC [? *n* h
Reset screen mode (RM) ESC [= *n* 1 *or* ESC [= 1 *or*
ESC [? *n* 1

These sequences set and reset the screen operating mode. *n* may take the values shown in Table 11-5.

Table 11-5 Screen Operating Modes

Parameter	Description
0	40 x 25 monochrome
1	40 x 25 color
2	80 x 25 monochrome
3	80 x 25 color
4	320 x 200 monochrome
5	320 x 200 color
6	640 x 200 monochrome

Table 11-5 Screen Operating Modes (continued)

Parameter	Description
7	Enables (SM) or disables (RM) wrapping at the row end. When this parameter is enabled, extra characters wrap on to the next line. When disabled, they are lost.

Assign a key (KR) ESC [n ; n ;... n p

Changes the function of a key. First define the key to be changed and then define the change.

You can use extended character codes (ECCs) by typing the first variable in the sequence as 0 (NUL) followed by the second key. For example, 0;68; is the F10 key. The last variables in the escape sequence are the character codes for the values you are assigning to the key. An alternative to the sequence of numbers is a string defined as characters enclosed in double quotation marks. For example, 65;66;67 can be shown as "ABC";. Table 11-6 lists the codes for function keys F1 to F10, used alone or with the **Shift**, **Ctrl** or **Alt** keys.

Table 11-6 Decimal Key Codes

Key	Code	+ Shift	+ Ctrl	+ Alt
F1	59	84	94	104
F2	60	85	95	105
F3	61	86	96	106
F4	62	87	97	107
F5	63	88	98	108
F6	64	89	99	109
F7	65	90	100	110
F8	66	91	101	111
F9	67	92	102	112
F10	68	93	103	113

Examples

You can demonstrate the following examples by using EDITOR to create a file with the appropriate escape sequence, and then entering that filename as a parameter to the TYPE command. Remember that you have to enter the control code for ESC, rather than the character itself (in EDITOR, **Ctrl+P** followed by the **Esc** key produces the ESC code).

To position cursor at row 10, column 10, and print an x at that point:

```
ESC [10;10Hx
```

To set screen mode to 40x25 color:

```
ESC [?1h
```

To assign XDIR C: to the F10 key:

```
ESC [0;68;"XDIR C:";13p
```

To reset current attributes and then set green as foreground color on a white background:

```
ESC [0;32;47m
```

DEVSWAP.COM

DEVSWAP.COM is a device driver required by the SuperStor program to change drive letter assignments for SuperStor compressed drives on your system. Because SSTORDRV.SYS automatically assigns drive letters at the end of the drive letter chain to compressed drives, DEVSWAP.COM reassigns drive letters so that the data you compressed (on drive C:, for example) is accessed as drive C: instead of the drive letter assigned by SSTORDRV.DRV.

DEVSWAP.COM must be loaded after SSTORDRV.SYS. This is done automatically by the SETUP and SSTOR programs. You can also edit your **config.sys** file directly and insert a DEVICE = statement.

Before you load DEVSWAP.COM, read Chapter 13, "Optimizing your disk" for a complete description of the SuperStor program.

DISPLAY.SYS

This driver enables code page switching for EGA and VGA displays.

To use the DISPLAY.SYS driver, put the following statement into your **config.sys** file:

```
DEVICE=c:\drdos\display.sys CON=(type,hwcp,n|(n,m))
```

This statement is included automatically if you use the SETUP program to install code page switching.

Refer to Chapter 14, "Setting up code page switching", for more information about code pages and code page switching.

Options

type — The type of active display. Enter EGA for EGA and VGA displays. If you do not include a type, DISPLAY.SYS checks your hardware to select the display type you are currently using.

hwcp — The hardware resident code page. DISPLAY.SYS cannot check this parameter, so do not include it if you are not sure what to enter.

n — The number of additional code pages you want to prepare, usually 1. The maximum number you can enter is 12.

Note that preparing a code page allocates space in memory for it; the amount of memory available for applications is therefore reduced by each code page preparation you specify.

Also note that this parameter only *prepares* for additional code pages. You prepare and select specific code pages using the MODE command, either in your **autoexec.bat** file or in the command line. Refer to the MODE command description in Chapter 10, "DR DOS commands".

Options (continued)

m The number of subfonts stored in memory. The default is for all font sizes to be resident in memory, but you can use *m* to reduce the number and increase the amount of memory available to applications. The possible values for *m* are as follows:

Display Type	Number of Subfonts	Supported Fonts
EGA	2 (default)	8*14 and 8*8
EGA	1	8*14
VGA	3 (default)	8*16, 8*14, and 8*8
VGA	2	8*16 and 8*8
VGA	1	8*16

Examples

The following command defines EGA (or VGA) as the display type, Code Page 437 as the hardware code page, and one additional code page (to be defined by the MODE command).

```
DEVICE=c:\drdos\display.sys CON=(EGA,437,1)
```

The next command is the same as the previous one, except that it restricts the number of subfonts stored in memory to 1 (8*14):

```
DEVICE=c:\drdos\dislplay.sys CON=(EGA,437,(1,1))
```

DRIVER.SYS

DRIVER.SYS sets the characteristics to be associated with a physical disk drive specified in the command line. Use this device driver if you want to add a *new* logical drive of a different type to an existing DR DOS system (see also your hardware documentation). For example you may want to add a 3½" disk drive to a system which uses 5¼" drives. Note that this only applies if the computer's internal ROMBIOS does not support 3½" disk drives. The full command is:

```
DEVICE =c:\drdos\driver.sys /D:n [/C] [/F:f] [/H:h]
[/N] [/S:ss] [/T:tt]
```

If you want to modify the characteristics of a logical drive already known to DR DOS use DRIVPARM, as described in section 11.3, "Config.sys commands".

Options

/D:*n* Specifies the physical drive (for example 0,1,2) for which the characteristics are set.

/C Indicates that the drive is able to detect when a disk is changed.

/F:*f* Specifies the drive type. The possible values are 0, 1, 2 and 7, where 0 = 360 Kbyte 5¼" drive, 1 = 1.2 Mbyte 5¼" drive, 2 = 720 Kbyte 3½" drive and 7 = 1.44 Mbyte 3½" drive. The default is 2.

/H:*h* Specifies the number of drive heads. Possible values are 1 and 2.

/N Indicates that the drive is for permanent media.

/S:*ss* Specifies the number of disk sectors per track that the drive supports. Possible values can be between 1 and 63.

/T:*tt* Specifies the number of tracks that the drive supports. Possible values are 40 and 80.

Example

For a 720 Kbytes 3½" drive as the second drive, insert the statement:

```
DEVICE = driver.sys /D:1 /C /F:2 /H:2 /S:9 /T:80
```

EMM386.SYS

EMM386.SYS is an additional device driver that allows DR DOS to make better use of the larger amounts of memory available on Intel 386, 386SX® and i486-based computers.

EMM386.SYS offers the following features:

❑ Supports the LIM 4.0 expanded memory specification without the need for special add-on cards (/FRAME and /KB options).

❑ Relocates the operating system kernel to upper or high memory (/BDOS option).

❑ Enables upper memory to be used for TSRs, device drivers and operating system data structures (/AUTOSCAN, /INCLUDE, /EXCLUDE and /USE options).

❑ Adds 64Kb or 96Kb to conventional memory if an MDA, Hercules™, CGA, EGA (text mode) or VGA (text mode) video adaptor is being used (/VIDEO option).

❑ Copies code and data from ROM to RAM to speed the execution of certain processes (/ROM option).

❑ Provides an extra 256Kb of extended memory on Compaq® machines (/COMPAQ option).

❑ Loads itself in conventional (/LOWEMM option) or upper memory.

❑ Provides an option to disable the relocation of the extended BIOS data area from the top of conventional memory (/XBDA option).

❑ Provides full XMS support.

❑ Works with Microsoft Windows 3 in Real, Standard, and Enhanced mode.

❑ Supports the Virtual Control Program Interface (VCPI).

The full **config.sys** command is:

```
DEVICE = c:\drdos\emm386.sys [/FRAME=AUTO|NONE|nnnn]
[/KB=0|AUTO|nnnn] [/AUTOSCAN=start-end]
[/INCLUDE=start-end] [/EXCLUDE=start-end]
[/VIDEO[=[start]-end]] [/BDOS=AUTO|FFFF|nnnn]
[/USE=start-end] [/ROM=AUTO|start-end] [/COMPAQ]
[/LOWEMM] [/XBDA] [/WINSTD]
```

Defaults

If you do not specify any command options, EMM386.SYS automatically scans upper memory between C000H and FFFFH to see which 4 Kbyte blocks are free and establishes a 64Kb LIM window in upper memory.

Note

EMM386.SYS must be installed before any other drivers that use extended memory.

It cannot be used with third-party memory managers.

Support for LIM 4.0 is optional; it may be that there is not enough contiguous memory within upper memory to provide the 64 Kbyte window necessary, or alternatively this feature may not be required. If LIM support is not enabled, all other features of the device driver are still available.

SETUP

The easiest way to configure EMM386.SYS is to use the SETUP program (although if you prefer you can edit the **config.sys** file directly and insert a DEVICE = statement using the options described on the following pages). There are two EMM386.SYS screens in SETUP. The first screen will be of interest to most users, and contains options to which you answer Yes or No. The second screen contains more advanced options.

The EMM386.SYS options are detailed below. Note that you can abbreviate the keywords to one letter; for example /FRAME=AUTO can be shortened to /F=AUTO.

Options

/FRAME=*xxxx* This option controls the LIM 4.0 emulation, where *xxxx* can be:

AUTO	Performs an automatic search through upper memory for a free 64 Kbyte window. This is the default.
NONE	Disables support for expanded memory. Use this option if you do not use expanded memory on your computer, but you want to take advantage of other features of EMM386.SYS.

Options (continued)

nnnn Specifies an address for the location of the 64 Kbyte window, provided 64 Kbytes of contiguous memory are available at that address. For example, /F=C000 forces the LIM window to be allocated at segment C000h. If 64 Kbytes of contiguous memory are not available at the address specified, you get a message reporting the conflict.

/KB = *xxxx* This option controls the amount of memory to be used for LIM memory, where *xxxx* can be:

0 or AUTO Use all extended memory (the default).

nnnn Use the amount of memory specified by *nnnn* in kilobytes. Note that this value must be given in decimal, whereas most of the other options are given in hexadecimal.

/AUTOSCAN=*start-end*

This option automatically scans 4Kb blocks of upper memory to see whether they are free. It is the most sensitive of the EMM386.SYS tests, and is the one most likely to define an area of upper memory as unavailable. *start* and *end* specify the start and end segment addresses of the memory to be scanned. If you do not include /AUTOSCAN in the EMM386.SYS command, then by default EMM386.SYS autoscans segment addresses C000H to FFFFH.

/INCLUDE=*start-end*

This option is similar to AUTOSCAN, but tells EMM386.SYS to test a portion of upper memory that AUTOSCAN would otherwise ignore. The /INCLUDE option uses less severe tests than the AUTOSCAN option, and can be used in place of AUTOSCAN to test an area of upper memory that is known to be available. *start* and *end* specify the start and end

Options (continued)

segment addresses of the memory included in the scan.

/EXCLUDE=*start-end*

This option is the opposite of /INCLUDE and can be used to specifically exclude an area of upper memory from the area being checked for usability. In this way you can exclude an area of upper memory that /AUTOSCAN detects as being available, but which is being used by, for example, an application or network card. *start* and *end* specify the start and end segment addresses of memory excluded from the scan.

/VIDEO[=[*start*]-*end*]

This option extends conventional memory by releasing unused space reserved in upper memory for the Video Display Adaptor. The additional space is contiguous to the 640Kb of conventional memory.

If your computer has a MDA or Hercules video adaptor, /VIDEO increases conventional memory by 64Kb, while on a CGA system, conventional memory increases by 96Kb.

When you have a VGA or EGA video adaptor, /VIDEO reserves the additional memory, but does not add it to conventional memory until you run MEMMAX +V. MEMMAX adds 64Kb to conventional memory if your system is set up for monochrome text mode, and 96Kb if it is set up in colour mode. Note that you cannot use graphics options on your monitor while the additional memory is enabled; you must first disable the memory by running MEMMAX -V, which releases the 64Kb or 96Kb block of memory back to the Video Display Adaptor.

start and *end* specify the start and end segment addresses of the video memory you want to reserve. If you specify *start*, it must be A000.

Options (continued)

/BDOS=*xxxx* Relocates the DR DOS kernel, freeing space for use by applications. *xxxx* can be:

AUTO	Automatically scans upper memory for a contiguous area of memory to map the kernel into. If no upper memory is found, segment FFFF is used. AUTO is the /BDOS default.
FFFF	Relocates the kernel to segment FFFF in high memory, freeing more space for use by applications.
nnnn	Relocates the kernel to the specified segment address in upper memory. This option works only if sufficient contiguous upper memory is available at the specified address.
NONE	The kernel is not relocated.

/USE=*start-end*

This option overrides all the EMM386.SYS memory tests and makes the specified upper memory available. *start* and *end* specify the start and end segment addresses of the memory that is to be used.

Caution The /USE option should only be specified by advanced users with experience of using and allocating upper memory.

/ROM=*xxxx* This option copies data and code from ROM, which is slow to execute, into RAM, which is fast. It then arranges for the RAM to have the same addresses as the ROM (that is, it maps the RAM over the ROM). This option is useful, for example, with video cards that call code in ROM to output characters. If the ROM is copied to RAM, character output is much faster.

Options (continued)

xxxx can be:

start-end Specifies the start and end segment addresses of the ROM that you want to copy.

AUTO Copies all ROMs that can be found in the upper memory of the system to RAM.

NONE No ROM is copied to RAM (default).

/COMPAQ Makes an extra 256 Kbytes of extended memory available on Compaq 386 and 486 machines configured with over 1Mb of memory.

/LOWEMM By default, EMM386.SYS relocates most of itself (approximately 24Kb) to upper memory, and leaves only about 1Kb of code in conventional memory.

The /LOWEMM option forces EMM386.SYS to load in conventional memory. This option may be useful if you have insufficient space in upper memory for EMM386.SYS together with, for example, a network adaptor and a page frame.

/XBDA By default, EMM386.SYS relocates any extended BIOS data area from the top of conventional memory to the bottom of the free area of conventional memory. The /XBDA option disables the relocation, and is used when a program requires the extended BIOS data area to be at the top of conventional memory.

Note that if you keep the extended BIOS data area at the top of conventional memory, any additional memory provided by the /VIDEO option will not be contiguous to conventional memory.

Options (continued)

/WINSTD	Forces compatibility with Windows 3.0 in Standard mode. Note that using /WINSTD will reduce the amount of conventional memory available to applications because upper memory is disabled. For more information about using Windows with DR DOS, refer to Appendix E, "Using Windows with DR DOS".

VCPI support

VCPI (Virtual Control Program Interface) is supported by EMM386.SYS, which runs in protected mode in order to access certain features of the Intel 386 and i486 processors. EMM386.SYS can co-exist with other protected-mode programs that conform to the VCPI interface.

Examples

The following command autoscans memory from C000 to FFFF (the default), but specifically excludes the area E800 to FFFF. No LIM page frame is set up, and the DOS kernel is loaded into upper memory or, if there is not enough upper memory, it is loaded into high memory.

```
DEVICE = c:\drdos\emm386.sys /F=NONE /B=AUTO /E=E800-FFFF
```

The following command specifies C400 as the address for the LIM window, with 2048 Kbytes of LIM memory to be used, and the BDOS located to segment FFFF in high memory. Upper memory from C000 to FFFF is autoscanned.

```
DEVICE = c:\drdos\emm386.sys /FRAME=C400 /KB=2048 /BDOS=FFFF
```

For further examples of how to use EMM386.SYS, refer to Chapter 12, "Using MemoryMAX".

EMMXMA.SYS

EMMXMA.SYS is an additional device driver that provides LIM (Lotus, Intel, Microsoft) 4.0 expanded memory for applications that can take advantage of it. The driver is for use on IBM PS/2 computers. It converts memory on cards compatible with the IBM XMA standard from extended to expanded memory. This memory is mapped to a 64 Kbyte window positioned between the addresses C000H and DFFFH. To use the device driver, insert the following line in the **config.sys** file:

```
DEVICE = c:\drdos\emmxma.sys [/FRAME=xxxx] [/KB=ddd]
```

Options

/FRAME=*xxxx*	*xxxx* specifies the hexadecimal segment address of the start of the 64 Kbyte mapping window. If you omit this option, the driver searches automatically for a free 64 Kbyte window between segment addresses C000H and DFFFH.
/KB=*nnnn*	*nnnn* specifies the total amount of memory to be allocated for LIM memory. If you omit this option, the default is to use all available memory.

Note

You cannot use EMMXMA.SYS with third-party memory managers.

Example

The following **config.sys** line specifies that the mapping window for LIM memory begins at hexadecimal address D000H and that 2096 Kbytes of memory is allocated for LIM use:

```
DEVICE = c:\drdos\emmxma.sys /FRAME=D000 /KB=2096
```

HIDOS.SYS

On an Intel 80286-based computer with extended memory, HIDOS.SYS relocates the DR DOS operating system kernel to high memory (see the /BDOS option).

On computers based on the following chip sets HIDOS.SYS offers more features:

- ❑ Chips and Technologies LeAPSet™ and LeAPSetsx™ CHIPSet™.
- ❑ Chips and Technologies NEAT™ and NEATsx™ CHIPSet.
- ❑ Chips and Technologies SCAT™.

If your computer has one of these chip sets HIDOS.SYS offers the following features:

- ❑ Enables upper memory to be used for TSRs, device drivers, and operating system data structures (/AUTOSCAN, /INCLUDE, /EXCLUDE and /USE options).
- ❑ Adds 64Kb or 96Kb to conventional memory if an MDA, Hercules, CGA, EGA (text mode), or VGA (text mode) video adaptor is being used (/VIDEO option).
- ❑ Copies code and data from ROM to RAM to speed the execution of certain processes (/ROM option).
- ❑ Relocates the operating system kernel to upper or high memory (/BDOS option).
- ❑ Provides an option to disable the relocation of the extended BIOS data area from the top of conventional memory (/XBDA option).
- ❑ Enables you to specify the type of chip set, or to disable the mapping of SHADOW RAM (/CHIPSET option).

HIDOS.SYS also supports EMS 4.0 upper memory blocks (or EEMS) and any permanent unused areas or upper RAM.

You use the HIDEVICE, HIINSTALL, HIDOS and HILOAD commands to load the drivers, TSRs, networking software and data structures into upper memory.

If you are running a third-party memory manager that is XMS compatible, you can still use HIDOS.SYS to relocate the operating system kernel to upper or high memory; see the section "Third-party memory managers" in Chapter 12, "Using MemoryMAX" for details.

The full **config.sys** command is:

```
DEVICE = c:\drdos\hidos.sys [/AUTOSCAN=start-end]
[/INCLUDE=start-end] [/EXCLUDE=start-end]
[/VIDEO[=[start]-end]] [/BDOS=AUTO|FFFF|nnnn]
[/USE=start-end] [/ROM=AUTO|start-end|NONE] [/XBDA]
[/CHIPSET=AUTO|chipset|NONE]
```

The HIDOS.SYS options are listed below. Note that you can abbreviate keywords to one letter; for example /VIDEO can be shortened to /V.

Options

/AUTOSCAN=*start-end*

This option automatically scans 4Kb blocks of upper memory to see whether they are free. It is the most sensitive of the HIDOS.SYS tests, and is the one most likely to define an area of upper memory as unavailable. *start* and *end* specify the start and end segment addresses of the memory to be scanned. If you do not include the /AUTOSCAN option in the HIDOS.SYS command, the default addresses for the HIDOS.SYS tests are *start* = C000H and *end* = FFFFH.

/INCLUDE=*start-end*

This option is similar to AUTOSCAN, but tells HIDOS.SYS to test a portion of upper memory that AUTOSCAN would otherwise ignore. The /INCLUDE option uses less severe tests than the AUTOSCAN option, and can be used in place of AUTOSCAN to test an area of upper memory that is known to be available. *start* and *end* specify the start and end segment addresses of the memory included in the scan.

/EXCLUDE=*start-end*

This option is the opposite of /INCLUDE and can be used to specifically exclude an area of upper memory from the area being checked for usability. In this way you can exclude an area of upper memory that /AUTOSCAN detects as

Options (continued)

being available, but which is being used by, for example, an application or network card. *start and end* specify the start and end segment addresses of memory excluded from the scan.

/VIDEO[=[*start*]-*end*]

This option extends conventional memory by releasing unused space reserved in upper memory for the Video Display Adaptor. The additional space is contiguous to the 640Kb of conventional memory.

If your computer has a MDA or Hercules video adaptor, /VIDEO increases conventional memory by 64Kb, while on a CGA system, conventional memory increases by 96Kb.

When you have a VGA or EGA video adaptor, /VIDEO reserves the additional memory, but does not add it to conventional memory until you run MEMMAX +V. MEMMAX adds 64Kb to conventional memory if your system is set up for monochrome text mode, and 96Kb if it is set up in colour mode. Note that you cannot use graphics options on your monitor while the additional memory is enabled: you must first disable the memory by running MEMMAX -V, which releases the 64Kb or 96Kb block of memory back to the Video Display Adaptor.

start and *end* specify the start and end segment addresses of the video memory you want to reserve. If you specify *start*, it must be A000.

/BDOS=*xxxx* Relocates the DR DOS kernel, freeing space for use by applications. *xxxx* can be:

AUTO Automatically scans upper memory for a contiguous area of memory to map the kernel into. If no upper memory is found, segment FFFF is used. AUTO is the /BDOS default.

Options (continued)

FFFF	Relocates the kernel to segment FFFF in high memory, freeing more space for use by applications.
nnnn	Relocates the kernel to the specified segment address in upper memory. This option works only if sufficient contiguous upper memory is available at this address.
NONE	The kernel is not relocated.

/USE=*start-end*

This option overrides all the HIDOS.SYS memory tests and makes the specified upper memory available. *start* and *end* specify the start and end segment addresses of the memory that is to be used.

Caution The /USE option should only be specified by advanced users with experience of using and allocating upper memory.

/ROM=*xxxx* This option copies data and code from ROM, which is slow to execute, into RAM, which is fast. It then arranges for the RAM to have the same addresses as the ROM (that is, it maps the RAM over the ROM). This option is useful, for example, with video cards that call code in ROM to output characters. If the ROM is copied to RAM, character output is much faster.

xxxx can be:

start-end	Specifies the start and end segment addresses of the ROM that you want to copy.
AUTO	Copies all ROMs that can be found in the upper memory of the system to RAM.
NONE	No ROM is copied to RAM (default).

Options (continued)

/XBDA

By default, HIDOS.SYS relocates any extended BIOS data area from the top of conventional memory to the bottom of the free area of conventional memory. /XBDA disables the relocation, and is used when a program requires the extended BIOS data area to be at the top of conventional memory.

Note that if you keep the extended BIOS data area at the top of conventional memory, any additional memory provided by the /VIDEO option will not be contiguous to conventional memory.

/CHIPSET=*chipset*

This option specifies the chip set, or EMS or RAM, in your computer. HIDOS.SYS normally detects the type of chip set (or EMS or RAM) and, if the type is supported, uses this information to initialize (map) Shadow RAM in upper memory. The /CHIPSET option is for use when you have difficulty mapping Shadow RAM on a computer fitted with one of the supported chip sets (or EMS or RAM), or when you want to prevent mapping. *chipset* can be:

AUTO HIDOS.SYS automatically detects the type of chip set (the default).

NEAT Chips and Technologies NEAT, NEATsx, LeAPSet, or LeAPSetsx CHIPSet™.

SCAT Chips and Technologies SCAT™ chip set.

EMSUMB

EMS 4.0 or EEMS upper memory blocks. (You must load an EMS 4.0 or EEMS device driver first.)

EMSALL

All EMS upper memory including the page frame. This option disables EMS for all other programs.

Options (continued)

RAM	Permanent upper RAM. You must use the /USE option to tell HIDOS.SYS where the RAM is.
NONE	No mappable Shadow RAM is available, or it is not to be used.

Examples

The following command relocates the operating system kernel on an Intel 80286-based computer with extended memory, to high memory. (Note that the operating system kernel can be relocated to high memory on any Intel 80286, 386 or i486-based computer.)

```
DEVICE = c:\drdos\hidos.sys /BDOS=FFFF
```

The following command automatically detects a supported chip set, and maps Shadow RAM in all available upper memory except D000-DFFF. It automatically copies all ROMs to RAM, and relocates the BDOS to upper memory.

```
DEVICE = c:\drdos\hidos.sys /E=D000-DFFF /ROM=AUTO /BDOS=AUTO
```

For further examples of how to use HIDOS.SYS, refer to Chapter 12, "Using MemoryMAX".

PRINTER.SYS

This driver enables code page switching for printers.

To use the PRINTER.SYS driver, put the following statement in your **config.sys** file:

```
DEVICE=c:\drdos\printer.sys LPTn=(type,hwcp,n)
```

This statement is included automatically if you use the SETUP program to install code page switching.

Refer to Chapter 14, "Setting up code page switching", for more information about code pages and code page switching.

Options

LPT*n* — The port to which the printer is connected, where *n* is 1, 2, or 3. Note that PRN is equivalent to LPT1, and you can use PRN instead.

You must define each printer on your system that supports code page switching, up to a limit of three entries in the same DEVICE statement.

type — The type of printer. Enter 4201 for IBM 4201 Proprinter and IBM 4202 Proprinter XL, 4208 for IBM Proprinter 4207 XL24 and 4208 XL24, 5202 for IBM Quietwriter III, 1050 for Epson FX 850 and FX 1050 printers, or a type compatible with one of these printers.

hwcp — The hardware resident code page. This is an optional parameter.

n — The number of additional code pages you want to prepare for the printer usually 1 or 2. The maximum number you can enter is 12.

Note that preparing a code page allocates space in memory for it; the amount of memory available for applications is therefore reduced by each code page preparation you specify.

Example

The following command prepares two printers with an additional code page:

```
DEVICE=c:\drdos\printer.sys PRN=(4201,437,1) LPT2=(5202,437,1)
```

SSTORDRV.SYS

SSTORDRV.SYS is a device driver for the SuperStor program. SSTORDRV.SYS is the Block Driver that manages compressed disk partitions (created by SuperStor). The driver also allocates drive letters.

SSTORDRV.SYS occupies a minimum of 28 Kbytes. The Block Driver will increase in size by about 4 Kbytes for each compressed drive allocated in your system.

Note

SSTORDRV.SYS must be installed before any other drivers except memory managers (EMM386.SYS, for example).

The easiest way to configure SSTORDRV.SYS is to use the SETUP program, although if you prefer you can edit the **config.sys** file directly and insert a DEVICE = statement.

Before you load SSTORDRV.SYS, read Chapter 13, "Optimizing your disk" for a complete description of the SuperStor program.

VDISK.SYS

VDISK.SYS lets you set aside a specified portion of your computer's memory to act as if it were a fast disk. This area of memory is called a memory disk.

A memory disk has a very fast access time, but it is only in effect while your computer is switched on. When you switch off or reset your computer everything stored in a virtual disk is lost. This means that anything stored on it that you want to keep must be copied to a hard disk or diskette before switching off.

VDISK.SYS can be useful for storing temporary files used by some programs, or by programs that need to access a lot of information quickly and often, like a database or spreadsheet.

If you have extended or expanded memory, you can set up a virtual disk using the switches described below. Otherwise your virtual disk is automatically set up in conventional memory.

The full command is:

```
DEVICE = c:\drdos\vdisk.sys [disksize]
[sectsize][maxfiles] [/E:sectors] [/X]
```

WARNING Remember that any files stored on virtual disks are lost when the computer is switched off or when DR DOS is restarted. Any information you need to keep must be stored on diskette or on your hard disk.

Options

disksize The size of the disk in Kbytes. The default is 64 Kbytes, and the range 1 through 256 Kbytes unless your system includes extended memory.

sectsize The smallest amount of space reserved on the disk. The default value is 128 bytes; alternative values are 256 or 512.

maxfiles The maximum number of files to be written to the root directory of the virtual disk and can be between 2 and 512; the default is 64. The number governs how much virtual disk space is reserved for the disk directory. Specifying a

Options (continued)

	large file number wastes space unless you plan to copy large numbers of files onto the virtual disk.
/E:*sectors*	This option is applicable if your computer has additional memory (called extended memory). The option locates the virtual disk in this extended memory. *Sectors* define how many sectors (1 through 8) are transferred at a time between ordinary memory and extended memory when the virtual disk is being used. The default is 8.
/X	Locates the virtual disk in expanded memory. The virtual disk can be up to 32 Mbytes.

VDISK must be installed before any other drivers which use extended memory are installed, but after EMM386.SYS which must always be loaded first. (The INSTALL and SETUP programs set this up automatically.)

If VDISK.SYS is loaded into upper memory with the HIDEVICE command, then the /E switch must be used to create the memory disk using extended memory.

Example

The following command leads to a virtual disk of 200 Kbytes with a sector size of 256 bytes and a maximum of 32 files.

```
DEVICE = c:\drdos\vdisk.sys 200 256 32
```

The following command leads to a virtual disk of 10240 Kbytes located in expanded memory.

```
DEVICE = c:\drdos\vdisk.sys 10240 /X
```

11.5 The autoexec.bat file

The **auto**exec.bat file is a batch file that DR DOS runs each time you start your computer to set it up the way you want it. When you install DR DOS, an **autoexec.bat** file is automatically built for you, and stored in the root directory of the disk from which you load DR DOS. The file must always be stored on this disk, usually your hard disk.

Following are brief descriptions of some of the commands you can put in your **autoexec.bat file**, or which you may already find set up in the file. All of these commands are described in more detail in Chapter 10, "DR DOS commands".

Note that the DR DOS SETUP and INSTALL programs automatically put two special labels in the **autoexec.bat** file, :DRDOSBEG and :DRDOSEND. SETUP processes all statements between these labels, so place any additional statements for DR DOS between the labels. Other statements, for other operating systems for example, should be placed outside the labels.

KEYB

KEYB tells DR DOS what kind of keyboard you are using. If you have a national keyboard (that is, one different from the standard keyboard used in the US), you *must* run the KEYB command each time you start DR DOS, to ensure that the characters you type are the same as those displayed on the screen.

PATH and APPEND

PATH tells DR DOS where to look for command, program, and batch files.

APPEND tells DR DOS where to look for any other type of file.

Placing PATH and APPEND commands in **autoexec.bat** automatically tailors the DR DOS search path to your own directory hierarchy, every time you start DR DOS.

SET

SET inserts strings of information into the command processor's environment. These strings are then available to all programs and applications you run.

TIME and DATE

TIME and DATE set the system time and date. You need to use TIME and DATE if your computer does not have a battery-powered clock. TIME and DATE prompt you to type in the correct time and date, and then set the system clock for you.

Placing TIME and DATE commands in **autoexec.bat** means that you are prompted to set the system clock every time you start DR DOS.

Note that if you do have a battery-powered clock, DR DOS reads it automatically at startup.

PROMPT

PROMPT changes the default DR DOS system prompt. For example, you could change it to the following:

```
Have a nice day!
```

You could (more usefully!) change it to show the current directory:

```
C:\WILLIAM\WUNDAWRD\REPORTS
```

You can change the prompt at any time, but placing the PROMPT command in **autoexec.bat** saves you having to change the prompt every time you start DR DOS.

Starting applications

Automatically start your favourite software applications by loading them from your **autoexec.bat** file.

For example, assume you have a word processor called WundaWord that is started by typing ww from a directory called **\wundawrd**. Normally, you would need to type both the following commands:

```
C>CHDIR \wundawrd

C>ww
```

Put both commands in **autoexec.bat,** to load WundaWord automatically every time you start your computer.

Example

```
KEYB UK
DATE
TIME
PROMPT $p$g
CD \wundawrd
WW
CD \
```

where:

Statement	Meaning
KEYB UK	Specifies a UK keyboard.
DATE	Prompts you for the date.
TIME	Prompts you for the time.
PROMPT pg	Shows the current drive and directory.
CD \wundawrd	Makes **\wundawrd** the current directory.
WW	Starts WundaWord.
CD \	Returns you to the root directory when you leave WundaWord.

12

Using MemoryMAX

THIS chapter describes the features offered by the DR DOS memory management tools, and explains how to use them to maximize the memory available to your applications.

To understand this chapter, you need to be familiar with the following terminology: lower, conventional, upper, high, extended, and expanded memory. If you are not sure about the meaning of any of these terms, refer to Appendix C, "Memory". In addition, you will occasionally need to refer to the detailed descriptions of MemoryMAX commands and device drivers in Chapters 10, "DR DOS commands" and 11, "Customizing your system".

12.1 What is MemoryMAX?

The term MemoryMAX describes the collection of device drivers and commands that are used for memory management under DR DOS. To take advantage of the features offered by MemoryMAX, your computer must have a 286, 386, 386sx, or 486 processor, and at least 1 Mbyte of memory.

Why use MemoryMAX?

The purpose of MemoryMAX is to give you more free memory in which to load your applications. Without a memory manager and without loading any device drivers, you will probably have about 580 Kbytes of free memory. However, modern applications frequently require 540 Kbytes memory, or more, in order to load. There is thus insufficient memory to load a large application such as Ventura Publisher, Lotus 1-2-3®, or dBase IV at the same time, for example, as network drivers. In addition, you might also want to run disk cache software or a TSR. MemoryMAX extends the amount of memory available to applications, so that even with network drivers and TSRs loaded, there can still be 620 Kbytes or more of conventional memory available. However, the exact amount of additonal memory that MemoryMAX provides, however, is dependent on your particular configuration.

MemoryMAX consists of three device drivers and five commands.

MemoryMAX drivers

The MemoryMAX device drivers are summarized below and detailed in Chapter 11, "Customizing your system".

EMM386.SYS

Use this driver on computers with 386, 386sx or 486 processors.

EMM386.SYS does the following:

- ❑ Supports LIM 4.0 expanded memory without special add-on cards.
- ❑ Relocates the operating system kernel to upper or high memory.
- ❑ Enables upper memory to be used for device drivers, TSRs and operating system data structures.
- ❑ Copies data from ROM to RAM.

HIDOS.SYS
On computers with the 286 processor and extended memory, HIDOS.SYS relocates the operating system kernel to high memory.

On computers that support Shadow RAM, HIDOS.SYS relocates the kernel to upper memory, and enables you to load TSRs, drivers, and operating system data structures into upper memory.

EMMXMA.SYS
Use this driver on computers with XMA memory cards, to convert the memory from extended to expanded memory.

MemoryMAX commands

The commands associated with MemoryMAX are summarized below. You use them after you have loaded EMM386.SYS or HIDOS.SYS.

HIBUFFERS
Creates DR DOS buffers in high or upper memory. HIBUFFERS is executed from **config.sys**. See Chapter 11, "Customizing your system" section 11.3, for a detailed command description.

HIDEVICE
Loads specified device drivers into upper memory. HIDEVICE is executed from **config.sys**. See Chapter 11, "Customizing your system", section 11.3, for the detailed command description.

HIDOS
Relocates operating system data structures into upper memory. HIDOS is executed from **config.sys**. See Chapter 11, "Customizing your system", section 11.3, for the detailed command description.

HIINSTALL
Loads specified TSRs into upper memory. HIINSTALL is executed from **config.sys**. See Chapter 11, "Customizing your system", section 11.3, for the detailed command description.

HILOAD
Loads specified TSRs into upper memory. HILOAD is executed from the command line or a batch file. See Chapter 10, "DR DOS commands" for the detailed command description.

MEMMAX Disables newly mapped areas of RAM if they confuse an application. MEMMAX also enables extra memory if it has been reserved from the video adaptor by EMM386.SYS or HIDOS.SYS. MEMMAX is executed from the command line or a batch file. See Chapter 10, "DR DOS commands" for the detailed command description.

12.2 MemoryMAX and your computer platform

As indicated in the previous section, your computer hardware determines which MemoryMAX drivers and commands you can use. It also affects the extent to which you are able to gain additional memory for your applications. Read your computer's documentation to determine which of the following categories your computer falls into:

□ 386, 386sx, or 486 processor.

□ 286 processor.

□ IBM PS/2 with XMA memory.

□ Uses one of the following chip sets:

  – Chips and Technologies NeAT, NeATsx, LeAPSet, or LeAPSetsx.

  – Chips and Technologies LeAPSet or LeAPSetsx.

  – Chips and Technologies SCAT.

□ EMS or EEMS expanded memory board with driver.

□ Permanent upper RAM.

Now read the appropriate subsections below to find out which MemoryMAX drivers and commands you should use.

Table 12-1 summarizes the interaction between device drivers, hardware platforms, and the resultant memory management features.

386, 386sx, or 486 processor

Use the EMM386.SYS device driver to "fine tune" your memory. Refer to Chapter 11 "Customizing your system", section 11.4, for a detailed description of the EMM386.SYS features and options. After you have loaded EMM386.SYS, you can use the HIBUFFERS (high memory only), HIDOS, HIDEVICE, HIINSTALL and HILOAD commands to take advantage of the upper memory initialized by EMM386.SYS.

Using SETUP

Instead of using EMM386.SYS and the MemoryMAX commands directly, you can use SETUP to maximize the memory available to your applications. Make the following configuration changes with SETUP:

1. Change the `Relocate DR DOS data areas and device drivers` option to `Yes`. This moves all DR DOS device drivers and data structures into high or upper memory.

2. In the EMM386.SYS `Advanced Options` menu, change the BDOS entry from AUTO to FFFF. This forces the operating system kernel to be moved to high memory, freeing upper memory for device drivers, networking software, and TSRs.

Disabling lower memory

It is likely that applications will load in lower memory as a result of the above reconfiguration. If you subsequently encounter problems when running an application, use the MEMMAX command to temporarily disable lower memory. See Chapter 10, "DR DOS commands" for details of MEMMAX.

Note

If you have a Compaq 386 or 486, you can use the EMM386.SYS /COMPAQ option to gain an extra 256 Kbytes of extended memory.

286 processor

You can use the HIDOS.SYS device driver to relocate the operating system to high memory. This will provide additional memory for applications. Refer to Chapter 11 "Customizing your system", section 11.4, for a detailed description of the HIDOS.SYS /BDOS option, which specifies the relocation.

PS/2 with XMA memory

You can use the EMMXMA.SYS driver to convert memory on cards compatible with the XMA standard from extended to LIM expanded memory. Do this if you want to run an application that can take advantage of LIM memory. After you have loaded EMMXMA.SYS, you can also load HIDOS.SYS to relocate the operating system kernel to high memory, providing additional conventional memory. Refer to Chapter 11, "Customizing your system", section 11.4, for detailed descriptions of EMMXMA.SYS and HIDOS.SYS.

Chips and Technologies, chip sets, expanded memory boards, or permanent upper RAM

You can use the HIDOS.SYS device driver to fine tune your memory. Refer to Chapter 11 "Customizing your system", section 11.4, for a detailed description of the HIDOS.SYS features and options. After you have loaded HIDOS.SYS, you can use the HIBUFFERS (high memory only), HIDOS, HIDEVICE, HIINSTALL and HILOAD commands to take advantage of the upper memory initialized by HIDOS.SYS. If your computer has a 286 processor, you can also use HIDOS.SYS to relocate the operating system kernel to high memory (see the previous subsection).

Using SETUP

Instead of using HIDOS.SYS and the MemoryMAX commands directly, you can use SETUP to maximize the memory available to your applications. Make the following configuration changes in SETUP:

1. Change the `Relocate DR DOS data areas` and `device drivers` option to `Yes`. This moves all DR DOS device drivers and data structures into high or upper memory.
2. In the HIDOS `Advanced Options` menu, change the BDOS entry from AUTO to FFFF. This forces the operating system kernel to be moved to high memory, freeing upper memory for device drivers, networking software, and TSRs.

Disabling lower memory

It is likely that applications will load in lower memory as a result of the above reconfiguration. If you subsequently encounter problems when running an application, use the MEMMAX command to temporarily disable lower memory. See Chapter 10, "DR DOS commands" for details of MEMMAX.

Note

HIDOS.SYS can only initialize upper memory if you have enabled Shadow RAM and if Shadow RAM is available. Your machine's hardware setup program, which is normally invoked by a special keyboard sequence during reset, enables Shadow RAM. However, the setup program usually allows you to move ROM into RAM as well; ensure you leave Shadow RAM available for HIDOS.SYS. Otherwise, the only HIDOS.SYS feature you will be able to use is the relocation of the operating system kernel to high memory (using the /BDOS option).

Table 12-1 Features Available With Various Hardware/ Device Driver Combinations

Applicable Hardware	Driver	LIM	DR DOS kernel relocation	XMS	DR DOS Upper memory
386, 386sx & 486	EMM386.SYS	✓	✓	✓	✓
IBM PS/2 80286 with XMA card	EMMXMA.SYS	✓	✗	✗	✗
any 80286 with extended memory	HIDOS.SYS	✗	✓	✓	✗
80286 with mappable Shadow RAM, no extended memory	HIDOS.SYS	✗	✓	✗	✓
80286 with mappable Shadow RAM, + extended memory	HIDOS.SYS	✗	✓	✓	✓
?	HIDOS.SYS with third party XMS memory manager	?	✓	✓	✗

✓ Supported feature

✗ Unsupported feature

? This entry depends on the features offered by the third party memory manager

Note: If upper or high memory is not supported, HIDOS=ON | OFF, HIDEVICE, HIINSTALL and HILOAD commands have no effect.

12.3 Experimenting with MemoryMAX

This section describes a sequence of changes you can make to **config.sys**, and the effects that these changes have on the configuration of your system's memory. The examples use EMM386.SYS, and thus assume that you have a 386sx machine, or higher. The section does not demonstrate all the options of the MemoryMAX drivers and commands, but aims to give you the confidence to investigate MemoryMAX features for yourself, and so get the most out of your system.

The easy way to configure your system is to use the DR DOS SETUP program. To understand more about MemoryMAX, however, you need to bypass SETUP and experiment with different commands in your **config.sys** file.

Switching MemoryMAX features off

1 To establish a base configuration on your system, enter SETUP and make a note of the settings on the screen showing optional device drivers and utilities (the screen on which the first option is `Extended display (ANSI.SYS) [YES/NO]`).

Set all these options to NO. (If necessary, you can return to the original settings later.) Do not alter any other SETUP values.

2 Exit SETUP and reboot your system.

3 Now use the MEM command to look at your memory configuration:

```
C>MEM /A /P
```

Make a note of the address shown in the table for the DR DOS kernel. The last lines of the MEM display tell you the size of the largest block of available conventional memory; it may be around 565Kb, but the exact figure is not important; what is important (for this example) is the change that occurs to available conventional memory when you use the MemoryMAX drivers and commands.

Relocating the kernel; creating a LIM window

4 Use the EDITOR to add the following command to your **config.sys** file:

```
DEVICE = C:\DRDOS\EMM386.SYS  /BDOS=FFFF
```

5 Reboot your system. The EMM386.SYS device driver relocates the operating system kernel to high memory. By default, EMM386.SYS also creates a 64Kb window in upper memory for LIM memory.

6 Run MEM again with the following command:

```
C>MEM /A /P
```

Note that the DR DOS kernel code has moved to address FFFF:0000. (Notealso that the address of the disk buffers have not moved yet but will be relocated by the next command in this example.) MEM will report that the largest available block of conventional memory has increased by about 35Kb. The graphical display produced by MEM will show the LIM window in upper memory; the window is labelled in the display as EMS memory.

Relocating operating system data structures

7 Next, you can relocate operating system data structures to upper memory. Do this by using EDITOR to change the **config.sys** line:

```
HIDOS=OFF
```

to

```
HIDOS=ON
```

8 Reboot and run MEM again. You see in the table that the disk buffers have been relocated to upper memory. The largest available block of conventional memory will have increased by about 20Kb; it may now be around 620Kb.

The preceding examples show that you can experiment safely with the EMM386.SYS options. If a particular option does not bring any benefits, simply revert to the previous setting for that option. For details of all the EMM386.SYS options, refer to the description of EMM386.SYS in Chapter 11, "Customizing your system".

You have now used EMM386.SYS to increase the amount of conventional memory available to your applications. However, you don't want to use this conventional memory unless you really need to; so when you load TSRs or additional device drivers, you should try to load them into upper memory rather than conventional memory.

Loading device drivers into upper memory

For example, you can add the following command (to **config.sys**) to load device driver ANSI.SYS into conventional memory:

```
DEVICE = C:\DRDOS\ANSI.SYS
```

You will find that the largest available block of conventional memory drops by about 4Kb. Replace the DEVICE statement in **config.sys** with the HIDEVICE command as follows:

```
HIDEVICE = C:\DRDOS\ANSI.SYS
```

This loads ANSI.SYS into upper memory if there is sufficient space. When you reboot and run MEM, you will find that the largest available block of conventional memory is the same size it was before you used the DEVICE command to load ANSI.SYS; you have loaded ANSI.SYS without any impact on conventional memory.

Loading TSRs into upper memory

Similarly, you can load TSRs into upper memory. To see the effect of TSRs on conventional memory, run GRAPHICS, for example, by typing the following:

```
C>GRAPHICS
```

If you now run MEM, this will show that conventional memory has dropped by about 2Kb. To load GRAPHICS into upper memory, add the following command to **config.sys**:

```
HIINSTALL=C:\DRDOS\GRAPHICS
```

When you reboot and run MEM, available conventional memory will be as it was before you first ran GRAPHICS; you have loaded GRAPHICS without any impact on conventional memory.

If you return to SETUP, and go to the screen of optional device drivers and utilities that begin with `Extended display (ANSI.SYS)`, you will see that the changes you have made to **config.sys** are reflected in the settings of the options.

Note

When you load a program into upper memory, it cannot displace a program already there. However, when you change your system configuration by relocating a TSR or device driver to upper memory and rebooting DR DOS, you may find that the largest block of available conventional memory is reduced in size. This will be because no more upper memory is available, and the relocation has displaced something else from upper memory back to conventional memory. Assume, for example, that 8Kb of upper memory is free and you attempt to load two device drivers, one 5Kb and one 7Kb, into upper memory. If the 7Kb driver is already in upper memory and is displaced back to conventional memory when you relocate the 5Kb driver to upper memory, your conventional memory will fall by 2Kb.

12.4 Potential memory conflicts

Applications

A few applications become confused if they find RAM memory in upper or lower memory. The DR DOS MEMMAX command enables you to temporarily disable these areas of memory while you load sensitive applications. The MEMMAX -U option temporarily locks upper memory, while the +U option reverses this temporary lock. The -L option temporarily locks lower memory, while the +L option reverses this temporary lock.

To load applications that are sensitive to the additional areas of RAM provided by MemoryMAX, DR DOS automatically adds the line MEMMAX -U>NUL to **autoexec.bat**. This locks upper memory; the redirection prevents any message from printing on the screen. You must reverse the lock with the command MEMMAX +U before you use HILOAD to load TSRs into upper memory.

Refer to Chapter 10, "DR DOS Commands" for a detailed description of the MEMMAX command.

Hardware devices

A further potential memory conflict relates to the use of upper memory by various hardware devices, such as high resolution graphics cards, Token Ring network cards, and hard drive controllers with onboard BIOS. Such hardware uses upper memory in a manner that could conflict with MemoryMAX, and you should exclude the memory the hardware occupies by using the /EXCLUDE option of EMM386.SYS or HIDOS.SYS. For example, some adaptor cards in upper memory only become visible to MemoryMAX when you run the associated hardware device driver. If you run EMM386.SYS or HIDOS.SYS without using the /EXCLUDE option, the MemoryMAX driver may overlay the upper memory required by the hardware device driver and the adapter card becomes inaccessible to its driver.

12.5 Third-party memory managers

Relocating the kernel

When you use a memory manager from another vendor, it is still possible to gain additional conventional memory space by relocating the DR DOS kernel to upper or high memory. The third-party memory manager must support the XMS specification, and upper or high memory must be available.

To relocate the kernel, insert the following statement after the statement that loads the third-party memory manager.

```
DEVICE=C:\DRDOS\HIDOS.SYS
```

HIDOS.SYS attempts, by default, to relocate the DR DOS kernel to upper memory. If upper memory is not supported or not available, the kernel is placed in high memory. You can force the kernel to high memory by using the following DEVICE statement:

```
DEVICE=C:\DRDOS\HIDOS.SYS  /BDOS=FFFF
```

The following examples show how and where to add the statement when loading Qualitas 386^{MAX}, or the Quarterdeck QEMM or QRAM drivers.

Qualitas 386^{MAX}

```
DEVICE=C:\386MAX\386MAX.SYS
DEVICE=C:\DRDOS\HIDOS.SYS
DEVICE=C:\386MAX\386LOAD.SYS......etc.
```

Quarterdeck QEMM.386

```
DEVICE=C:\QEMM\QEMM.SYS
DEVICE=C:\DRDOS\HIDOS.SYS
DEVICE=C:\QEMM\LOADHI.SYS......etc.
```

Quarterdeck QRAM (with Intel Above board)

```
DEVICE=C:\ABOARD\EMS.SYS AT 208 MC
DEVICE=C:\QRAM\QRAM.SYS
DEVICE=C:\DRDOS\HIDOS.SYS
DEVICE=C:\QRAM\LOADHI.SYS......etc.
```

Notes

- ❑ Neither EMM386.SYS or EMMXMA.SYS can be used with other memory managers.
- ❑ When a third-party memory manager is loaded, the following **config.sys** statements have no effect:

  HIINSTALL

  HIDEVICE
- ❑ When a third-party memory manager is loaded, the HILOAD command has no effect.

13

Optimizing your disk

This chapter describes three utilities you can use to make best the use of you hard disk.

- □ DISKOPT is a menu-driven utility that "optimizes" disk performance by reorganizing the arrangement of data on the disk.
- □ The Super PC-Kwik Disk Accelerator is a disk cache program that dramatically decreases the time it takes your system to read information from diskettes and the hard disk. Super PC-Kwik also increases the speed at which data is written to the disk.
- □ The SuperStor program compresses data stored on a disk; this considerably increases the storage capacity of the disk.

13.1 DISKOPT

Use DISKOPT to optimize the performance of a hard disk.

DISKOPT changes the position of data on a disk as follows:

- ❑ Makes files contiguous
- ❑ Moves all free space to the end of the disk
- ❑ Sorts directories (optional)

Before using DISKOPT

Before you use DISKOPT, be aware of the following:

- ❑ DISKOPT can take considerable time to run. Once you start disk optimization, however, you can stop it at any time by pressing the **Esc** key.
- ❑ Ensure that no other program uses the disk while DISKOPT is in progress. This is especially important to remember if you are using TaskMAX or TSRs, for example.
- ❑ Although DISKOPT checks the disk before optimizing it, you should run CHKDSK before using DISKOPT, and correct any errors CHKDSK finds. For more information about CHKDSK, refer to the CHKDSK command explanation in this chapter. You should also run CHKDSK if DISKOPT does not terminate normally, because of a power failure for example.
- ❑ Backup a disk before optimizing it. Refer to the BACKUP command explanation in this chapter.
- ❑ DISKOPT ignores system files. You can only include them in optimization by resetting the system attribute. Copy-protected files should generally *not* be moved by DISKOPT.

Using the DISKOPT menu

Type DISKOPT on its own followed by the disk drive. If you do not specify a drive, a list of valid drives is displayed from which you can choose the drive to optimize.

Once a drive is selected, you see a menu on your screen that consists of two elements: a title bar and a graphical representation of how data is arranged on the disk.

- ❑ The title bar contains the names of three menu options: `Optimize`, `Sort`, and `Help`.
- ❑ The rest of the screen is a kind of table, to illustrate how space on the disk is currently allocated. Each block in the table represents a data cluster, an unused space, or an unmoveable (or bad) cluster. Once you start to optimize the disk, this table will change to indicate how data is being read and then rewritten by DISKOPT.

You can interrupt optimization by pressing **Esc**, and you can exit from the DISKOPT menu at any time by pressing **Alt+X**.

Select a pull-down menu to open by pressing the **Alt** key with one of the following:

- ❑ **O** for the Optimize menu
- ❑ **S** for the Sort menu
- ❑ **H** for the Help menu

Once a menu is open, you can move between the three menus using the left ← and right → keys.

Close a menu by either selecting an option or pressing **Esc**.

Select any of the following options in an open menu by moving the selection bar and pressing **Enter** or typing the highlighted letter in the option name. For example, to choose the `Cluster` sort option, press **Alt+S** to open the Sort menu and then **Alt+C** to select `Cluster`.

Starting optimization

Optimize menu options	
Change **d**rive (**Alt+D**)	Allows you to specify a different disk drive.
Optimize disk	Starts disk optimization for the selected disk drive. You see a message indicating the DISKOPT is analysing the disk. The table reflects data movement while optimization is in progress. You see another message when optimization is finished.
E**x**it (**Alt+X**)	Cancels DISKOPT and returns you to the command prompt.

Sort menu options	
Name	Sorts disk entries alphabetically by name.
Extension	Sorts disk entriescontents alphabetically by file extension.
Date	Sorts disk entries by date, from oldest to most recent.
Size	Sorts disk entries by size, from smallest to largest.
Cluster	Sorts disk entries by starting cluster. Choosing this sort option usually means that optimization will take less time.
No sort	Leaves disk entries unsorted.

Help menu options	
Help for help	Displays information about using the online help.
Contents	Displays the online help table of contents.
Previous topic	Moves you back through previous screens of online help.
About ...	Displays copyright and version information.

13.2 Super PC-Kwik Disk Accelerator

Introduction

A disk cache is a collection of recently-used data from your disk, kept in Random Access Memory (RAM). A disk cache can reduce the number of times applications physically access the disk, and dramatically speed up the performance.

A disk cache works by accumulating copies of frequently-used sectors (disk locations) in memory. As new sectors are used, the cache adds copies of them as well. There may not always be room for a copy of a new sector and the cache must determine which sector copy to remove; the cache replaces the copy of the least recently used sector with a copy of the most recently used sector.

The Super PC-Kwik Disk Accelerator decreases the amount of time it takes your system to read information from diskettes and hard disks. Super PC-Kwik also increases the speed at which data is written to the disk. Super PC-Kwik is designed to make the most efficient use of whatever type of memory you have available on your system. Depending on your system configuration, Super PC-Kwik can lend that memory to other applications you are running.

Loading Super PC-Kwik

Super PC-Kwik is a memory-resident program; it must be loaded into memory in order to run. After you start it, however, it runs invisibly, speeding up the performance of your system.

You are prompted to specify whether you want the Super PC-Kwik cache loaded when you run the DR DOS INSTALL or SETUP programs. If you answer Yes, Super PC-Kwik is loaded and automatically configures itself for your system, choosing the options and settings to maximize your performance and work with the hardware and software you use. The appropriate statements are automatically added to your **config.sys** and **autoexec.bat** files.

Generally, you don't have to do anything else to get the most use from the Super PC-Kwik program. If, however, you want to change any of the options that INSTALL or SETUP automatically select for you, you can specify Super PC-Kwik at the command line, using the SUPERPCK command and the appropriate options. The most common options are described later in this chapter, in the section called "Super PC-Kwik options".

Note that if you want to change Super PC-Kwik options after you have already loaded it with INSTALL or SETUP, you must unload Super PC-Kwik first with the following command:

```
SUPERPCK /U
```

Notes about using Windows with Super PC-Kwik

Super PC-Kwik is compatible with Microsoft Windows. Ensure that you always load Super PC-Kwik before loading Windows, however; do not load Super PC-Kwik while already running Windows.

When you use the INSTALL or SETUP program to load Super PC-Kwik, you are prompted about whether you are using Windows in Standard or Enhanced mode. If this is the case, make sure you answer Yes. INSTALL or SETUP will then set your system correctly, loading the PCKWIN.SYS device driver and loading the cache into extended memory. Remember that you can use the SETUP program at any time to configure Super PC-Kwik for Windows in Standard or Enhanced mode.

Windows SETUP

By default, the Windows SETUP program loads the device drivers HIMEM.SYS and SMARTDRV.SYS. To avoid these changes to your **config.sys** files, remove the modifications during Windows SETUP or remove the statements with the DR DOS Editor after loading Windows.

To allow Super PC-Kwik to correctly report the use of advanced support for both diskettes and hard disks, you may want to change the section called `[386enh]` in the Windows **system.ini** file by adding the following line:

```
VirtualHDIrq=FALSE
```

This parameter is not case-sensitive.

Super PC-Kwik options

When you configure Super PC-Kwik with INSTALL or SETUP, all the necessary default values, such as a default cache size and a default memory type, are automatically assigned.

You can specify different options, however, by loading Super PC-Kwik from the command line; simply type SUPERPCK followed by the options you want to specify. You can specify multiple options in one command; there is no limit to the number of options, except for the DR DOS limit of 128 characters per command line. For each option, + turns it on and - suppresses it. The default is +, unless indicated otherwise.

Options used when loading Super PC-Kwik

These options are available when you load Super PC-Kwik on your system.

Options

/-*d* Do not cache drive *d*.

You may wish to restrict caching for certain drives. To do so, add the /- parameter, where *d* is a drive letter for each drive that should not be cached. No colon is needed. You may specify as many /- commands as you need, using one for each disk that is not cached.

Ensure the - character is before the drive letter.

Default: All local diskette and hard disk drives are cached. RAM disks and remote network drives are not cached.

/&U- Do not automatically move the parts of Super PC-Kwik that usually reside in conventional memory into upper memory. Normally, if you specify that the cache buffer go into expanded memory, only a small part of the code for the cache moves into upper memory blocks (UMBs); the track buffers, tables and remainder of the code reside in expanded memory. If you specify that the cache buffer go into extended memory, the track buffers, tables, and all code move to UMBs. If you specify that Super PC-Kwik run in conventional memory, everything moves to UMBs if there is enough upper memory available.

Default: /&U+ (load into UMBs if possible)

/A+ Use expanded memory for the Super PC-Kwik cache. Expanded memory is memory that conforms to the LIM (Lotus/Intel/ Microsoft) Expanded Memory Specification.

Default: /A+ when expanded memory is present and no other memory type is specified.

/EM+ Use extended memory for Super PC-Kwik's cache and minimize the amount of conventional memory used.

Options (continued)

If your applications use extended memory, the cache automatically handles the lending of memory to these programs. If your extended memory programs do not appear to be borrowing memory from Super PC-Kwik properly, you may wish to use the /R option.

Default: /EM when no expanded memory is available, extended memory is present, and no other memory type is specified.

/I+ Display drive table during installation.

/L:*nnnnn* Enable the memory lending feature, and allow up to *nnnnn* Kbytes of expanded or extended memory to be lent to other programs. The colon is required in this option. The maximum value for this option is determined dynamically by Super PC-Kwik. If there is not enough extended or expanded memory, Super PC-Kwik takes whatever is available and lends that amount instead of the amount specified.

Default: One-half the memory available for the disk cache.

/L- Disable lending.

/P+ On startup, display all options in effect for Super PC-Kwik.

Because the Super PC-Kwik program usually sets the default values for most of its options, it is a good idea to display a list of the options in effect when it is loaded. If you prefer, however, you can suppress this display by using the /P- option.

Default: /P-

/R:*nnnnn* Reserve *nnnnn* Kbytes of the memory type used by the cache for programs loaded after Super PC-Kwik. Allocate the rest of the memory to the cache. Note that the /L:*nnnn* option is usually a better way of lending memory to other programs.

Default: /R:480 for conventional memory. /R:0 for UMB, expanded and extended memory.

Options (continued)

/? Display a summary of these options. If you use the /? option with other options when you load Super PC-Kwik, the help summary displays, but the program does not load.

Example

The following command loads the Super PC-Kwik cache in expanded memory, using a small amount of conventional memory for code. It allocates as much expanded memory as is found for the cache. It lends to other programs until the minimum cache size is reached.

```
SUPERPCK /A+
```

Options used while Super PC-Kwik is running

The following options can be used after Super PC-Kwik has been loaded and is running:

Options

/D Disable the cache and reset its measurements to zero, leaving the cache in memory and turning off all caching functions.

/E Enable the cache after previously being disabled. There must be one enable command for each disable command issued in order for the cache to be enabled.

/F Flush the cache, clearing the memory of any disk data; reset the cache measurements to zero.

Super PC-Kwik always keeps your disks up to date. Flushing the cache does not cause data from your disks to be lost, nor do you need to flush the cache to keep your disks up to date. /F is useful when you disable or enable the cache. Because it only works when the cache is running, you get an error message if the cache is not yet loaded.

Options (continued)	
/M	Display disk cache measurements: the number of disk transfer requests, the number of physical transfers, the number of disk transfers saved, and the percentage of transfers saved.
/P	Display all options in effect. If you wish to try different options, you must unload the program and reload it with the new options. DR DOS requires that resident programs be unloaded last in, first out.
/U	Unload Super PC-Kwik.
/?	Display a summary of options.

Advanced options

In addition to the basic options described earlier in this section, there are advanced options that allow you to customize the Super PC-Kwik program to your system's requirements. These advanced options are available only while loading the Super PC-Kwik program. If you don't use an option or don't give a value with it, the program uses a default.

Options

/B+ When copying data to and from the cache, copy batches of sectors at a time. This option increases performance, especially when the cache is in extended memory.

When extended memory is used for the cache, batch copying may increase the length of time interrupts are turned off. If you use extended memory for your cache and you have trouble with high-speed data communication, for example over a 2400 baud modem, use /B-. If your cache is in conventional memory, expanded memory, or extended memory that has been converted to expanded memory by a memory manager, use /B+.

Default: /B+

/D+ Use advanced support to increase the performance of writing to diskette drives.

If /D- is specified, Super PC-Kwik uses generic cache support for your diskette drives, but not advanced support for increasing the performance of writing to the diskettes.

To take advantage of advanced cache support for diskette transfers, your diskette controller must be 100% IBM compatible.

Default: /D+ for IBM and 100% compatibles; /D- for unrecognized compatibles. Super PC-Kwik checks your system to determine the default.

Options (continued)

/Ex:*nnnnn*

Set the lowest address available to the cache in extended memory at memory location *nnnnn*K. X is either M or P, depending on whether you want to minimize the conventional memory used, or increase the performance of the cache.

It is seldom necessary to use this option, because Super PC-Kwik automatically detects most programs that use extended memory, and it automatically lends as much extended memory as is available to other programs that request it.

If you are using a program in extended memory that Super PC-Kwik does not seem to recognize, use this option to tell Super PC-Kwik to use only memory above location *nnnnn*K bytes. Be certain you know what location to give Super PC-Kwik, so that it runs above the other program. We recommend that you use the /R or /S options to restrict the amount of extended memory used by Super PC-Kwik.

Super PC-Kwik allocates extended memory from the top of memory down. When you use the /Ex:*nnnnn* option, Super PC-Kwik may not actually reach the location you specify, but it never allocates extended memory below the location you specify.

Default: /E:1024 when extended memory is used for the cache.

/G+

Use the boot sector information found on the drive when that information contradicts the information the BIOS has about the drive.

Use /G- if the information from the BIOS is to be used instead of the boot sector information.

Options (continued)

Occasionally, the boot sector on your disk contains information about the number of heads or the number of sectors per track that differs from the information the BIOS provides about the disk. When Super PC-Kwik detects this difference, it issues a warning message. This option lets you choose which set of information to use when the two sets contradict each other. You can choose to have Super PC-Kwik pause when it displays the warning message and wait for you to strike a key before proceeding; see the /K+ option below.

Default: /G+

/H+ Use advanced cache support to enhance the performance of writing to hard disk drives.

If /H- is specified, Super PC-Kwik provides cacheing for hard disks, but does not provide advanced support for writing to the disks.

To take advantage of advanced support, your hard disk controller needs to be 100% IBM compatible. ATs and compatibles can use advanced support of hard disk transfers even if the controller is not IBM compatible.

Default: /H+ for IBM ATs™, most 386s, and compatibles. /H- for others. Super PC-Kwik checks your system to determine the default.

/K+ Super PC-Kwik pauses and waits for you to type a key if a warning message displays while you are loading the program. Use /K- if you would prefer that loading proceed without any pauses.

Default: /K+

Options (continued)

/O+

Use the optional algorithm for advanced support of disk transfers.

This gives disk reads priority over disk writes. It affects diskette transfers when using the /D+ option and hard disk transfers when using the /H+ option. Depending on your applications and the speed of your disk drives, it may provide an additional performance gain.

This parameter usually improves the effect of the /Q+ option (see below) when you copy files from one hard disk directory to another, or from your hard disk to a diskette.

Default: /O-

/Q+

Quickly return DR DOS prompt.

When you are using /D+ or /H+, this option often allows a faster return to the DR DOS prompt while disk transfers associated with the previous command or application are still being completed. Returning the prompt allows you to enter additional commands while the disk transfers are taking place.

Be sure to wait for the lights on any diskette drive to go off before changing diskettes when using this option. Also, wait for the activity indicator on your hard disk to stop before you turn off or reboot your system.

If you select this option, you may wish to specify the /O+ option also (see above).

Default: /Q-

Options (continued)

/S:*nnnnn*

Allocate *nnnnn* Kbytes of RAM for Super PC-Kwik. Unless you need a specific cache buffer size, it is usually better to allow Super PC-Kwik to determine the amount of memory it needs and lend memory to other programs. See also the /L and /R options.

If you set up the cache in conventional memory, *nnnnn* should be between 64 and 512. If you set up the cache in expanded or extended memory, *nnnnn* should be between 64 and 16384.

Default: None. If the /S option is not specified, the /R option default applies.

/T+

Whenever an application requests a disk read, read the rest of the track into cache memory.

Disk space is divided into tracks, which are composed of sectors. There are typically 9 to 36 sectors per diskette track and 17 or more sectors per hard disk track. Because DR DOS frequently requests information from nearby sectors soon after requesting a given sector, reading the rest of the track in advance increases performance, but it uses about 8 Kbytes of memory to do so.

You can disable this feature by using /T- if you need to save the extra memory, or if your application accesses many small files scattered throughout the disk. You may also wish to disable full-track buffering if your disk has many bad sectors that have been mapped out.

If your system has a disk with a large number of sectors per track and you have the memory to allocate a large cache, you may wish to increase performance by using /TL, which defaults to the largest track size on your disk. You can also specify a track size to use with /T:*nn*.

Default: /T+, with a track size of 17 or the largest number of sectors per track, whichever is smaller.

Options (continued)

/TL

Perform track buffering using a buffer that holds up to the largest number of sectors per track detected.

Default: /T+

/T:*nn*

Perform track buffering using a buffer that holds up to *nn* sectors per track.

When you use the /T+ option, Super PC-Kwik automatically chooses a buffer size. The track buffer size chosen by the program usually produces the best performance for the least memory, but if you have additional memory to use and a disk with a larger track size, use the /T:*nn* option.

Default: /T+; the maximum is 72 sectors

/V+

Use volume-change detection hardware to detect the change of diskettes in a diskette drive.

High-capacity and 3½ inch diskette drives often have hardware that detects when a drive door has been opened. This volume-change detection hardware reduces software overhead. See the section "Super PC-Kwik and diskette drives" below to determine if your high-capacity drives have volume change detection hardware.

This option does not affect how 360K drives are handled as they never include volume-change detection hardware.

Default: /V+

/W+

Check write requests against the cache to determine if the disk already has identical data. If so, suppress the write.

Default: /W+

Lending memory

Super PC-Kwik needs memory to store:

- ❑ Its own program code.
- ❑ Portions of the disk track before and after the specific sector on the disk that your application has requested (track buffer).
- ❑ Data from the disk (cache buffer).

When you load Super PC-Kwik, it checks your system to see what kinds of memory are available. It uses different types of memory differently, and chooses memory based on what kind you have available and how it can best use that memory to reduce the amount of conventional memory it uses. It also shares and lends memory depending on what kind of memory you have available.

If the cache finds expanded memory in your system, it uses expanded memory for most of the program code and for the cache buffer. The track buffer and a small amount of code are in conventional memory. Your expanded memory manager decides where in expanded memory Super PC-Kwik resides. Super PC-Kwik lends out expanded memory allocated to its cache buffer to other application programs you are running that might need expanded memory.

If the cache finds extended memory in your system, it uses extended memory only for the cache buffers. The Super PC-Kwik program code and the track buffer reside in conventional memory. Super PC-Kwik uses extended memory from the top down, and lends out extended memory allocated to its cache buffer to other programs you are running that might need extended memory. If your extended memory programs don't seem to borrow memory from the cache buffer properly, you may need to use the /R option to reserve some memory for that application.

If you have the choice between expanded and extended memory, expanded is usually the better choice because Super PC-Kwik occupies less conventional memory and performs better.

When the cache finds only conventional memory, it uses conventional memory for the program code, the track buffer and the cache buffers. It uses conventional memory from the bottom up, starting at the first available address.

Super PC-Kwik and diskette drives

When you are using Super PC-Kwik to cache diskette drives, an application program sometimes asks you for additional information before the diskette drives have quit operating and the lights have turned off. Feel free to begin entering your next response or command whenever the application indicates that it is ready for it.

WARNING If your application asks you to change diskettes, wait until the diskette drive lights have turned off before removing the existing diskette from its drive.

The high-capacity diskette drive available on the IBM AT usually detects when the disk drive door has been opened. To double-check that your high-capacity drive supports this function:

1. Load Super PC-Kwik with the /V+ parameter.
2. Place a diskette in the high-capacity drive.
3. Use the DIR command to obtain a directory of the diskette.
4. Remove the diskette from the drive.
5. Place a different diskette with a different set of files in the drive.
6. Use the DIR command to obtain a directory of this diskette.

Observe whether the directory obtained in step 6 accurately reflects the contents of the diskette in the drive. If it is a directory listing for the previous diskette, the volume-change hardware is not working correctly. If so, unload Super PC-Kwik and reload it with the /V- option so that the program detects the volume change for you.

Some high-capacity drives depend on the spring-loaded ejection action of the drive to detect the diskette change. When the diskette does not eject as a result of the spring action, the drive may not detect that you have pulled the diskette out of the drive and replaced it with another. To test for this case, repeat the steps above with the following modification: use your thumb to prevent the diskette from springing out of the drive in step 4, then remove it by pulling it from the drive with your fingertips. If this change causes step 6 to produce the wrong directory listing, use the /V- option.

Super PC-Kwik and Bernoulli drives

IOMEGA Bernoulli drives are supported by Super PC-Kwik by placing a special driver in the **config.sys** file of your boot disk. This driver allows the Super PC-Kwik program to monitor transfers to and from Bernoulli drives.

To install the Super PC-Kwik driver:

1. Identify the line in your **config.sys** file that specifies the Bernoulli driver (IDRIVE.SYS, RCD.SYS, or equivalent) and place the following line directly after it:

   ```
   DEVICE=C:\DRDOS\PCKWIK.SYS
   ```

   If you placed the file **pckwik.sys** in a directory other than the **\drdos** subdirectory, specify the correct subdirectory.

   Note that both HIDOS.SYS and EMM386.SYS allow you to load device drivers in their upper memory area. Whether or not you load your Bernoulli driver high, you must load the Super PC-Kwik driver in low memory.

2. Reboot your system. You should notice a sign-on message from the PCKWIK.SYS device during the boot process.

3. Start Super PC-Kwik as you normally would.

Note

Add the special PC-Kwik driver even if you do not wish to cache your Bernoulli drives, because Super PC-Kwik detects the drive and tries to test it, even if it is not to be cached. It needs the driver to do so. If you don't wish to cache your Bernoulli drive, use the Super PC-Kwik /- option to suppress caching.

When Super PC-Kwik is loaded, it first checks for the Super PC-Kwik driver. If Super PC-Kwik has trouble with this driver, it issues one of the following two messages:

```
WARNING: PCKWIK.SYS placed after non-disk driver.

/-d Drive d cannot be cached -- physical unit unknown.
```

If you receive the first message, check to see that you have placed **pckwik.sys** after the Bernoulli driver and that your Bernoulli box is turned on. If you receive the second message and *d* is the drive letter for one of your Bernoulli drives, check to see that you have placed the **pckwik.sys** file either in the root directory of the boot disk or in the **\drdos** subdirectory on the boot disk. Also check to see that you have correctly spelled the name of the driver in your **config.sys** file.

Note

You may have received a program called IOMCACHE with your Bernoulli box. Because you may only run one disk caching program at a time, remove IOMCACHE from your system before you use Super PC-Kwik. Super PC-Kwik replaces the functions of IOMCACHE.

Using Super PC-Kwik with RLL, ESDI, IDE and SCSI disk controllers

The Super PC-Kwik program has been tested with several Run Length Limited (RLL), ESDI, and IDE hard disk controllers. Generally Super PC-Kwik automatically detects the larger number of sectors per track provided by these controllers. The size of the track buffer is shown on the screen display. If you prefer a different track buffer size, use the /T:*nn* option.

Super PC-Kwik supports Adaptek, Future Domain and Allways SCSI disk controllers. It may not recognize the partitioning schemes for other SCSI controllers, so you may need to either partition the disk as one large partition, or load the PC-Kwik device in your **config.sys** file as follows:

```
DEVICE=\DRDOS\PCKWIK.SYS
```

Accelerator hardware on PC and XT® computers

If you have increased the clock rate of your PC, XT, or compatible above the standard 4.77 MHz, you may be aware that DR DOS refuses to perform certain functions such as FORMAT and VERIFY unless you switch back to the standard speed. Super PC-Kwik includes a correction to this problem that allows you to format and verify at the higher clock speed. If you are already doing this because you loaded another software program, such as FASTVRFY, unload that program before using Super PC-Kwik.

Communication software and extended memory

Access to extended memory on the IBM AT and compatibles must be accomplished with interrupts turned off. By most standards, the period in which interrupts are turned off is quite short. However, there is a chance that high-speed data communication, as with a 2400-baud modem, may not be possible if the disk is being accessed at the same time. If you experience difficulty, we suggest you specify the /B- option for the Disk Accelerator when you use extended memory. Turning off batch copying limits the period when interrupts are turned off, but slightly reduces the performance gain. Interrupts are not a consideration if your extended memory is being used as expanded memory on an 80386 or 80486 with an expanded memory manager such as EMM386.SYS.

Using Super PC-Kwik with TaskMAX

TaskMAX may switch applications by moving them out of conventional memory into expanded or extended memory. If you are using TaskMAX, be sure to load Super PC-Kwik before TaskMAX. Switching the cache out and bringing it back in later can cause it to contain data that is seriously out of date, and may compromise the integrity of the data brought in from the disk.

Invalid boot sector errors

In rare circumstances, Super PC-Kwik may issue the following warning message during its installation:

```
WARNING: Boot sector for drive d invalid;
previous information used.
```

The first sector of drive or partition has an invalid signature. When the boot sector has a valid signature, Super PC-Kwik reads certain drive parameters from the boot sector and compares them to values obtained previously from the system BIOS or previous disk partitions. Some format programs may not put the IBM-standard signatures in the boot sector while they otherwise format the disk correctly. In this case, you can allow Super PC-Kwik to continue by using the information it has obtained from the BIOS and/or previous disk partitions.

If you wish to double check the drive parameters that Super PC-Kwik is using, add a /I option when you load Super PC-Kwik. This provides drive table at the end of the loading display. You can compare the Track Size (in sectors per track) and Heads information to the dimensions of your drive(s). You can also see the Logical Sector Size (LSS) and Physical Sector Size (PSS). The former is the sector size used by DOS for the particular driver and the latter is the sector size used by the BIOS.

Note

The Track Size shown for diskette drives is the default value; Super PC-Kwik automatically updates this value as you access various diskette types.

Troubleshooting

This section discusses some common problems you might encounter. The Super PC-Kwik error messages and warnings are listed in Appendix A, "Error messages and troubleshooting".

Hard disk is not cached

If Super PC-Kwik won't cache one of your drives, it is probably because:

1. The drive is a RAM drive. In this case, the drive already exists in memory and it would be redundant for the cache to transfer data to its cache buffer in memory.
2. The drive is a remote network drive. Caching might violate file or record locking protocols, so Super PC-Kwik won't cache remote drives.
3. The drive is a Bernoulli drive and you do not have the Super PC-Kwik driver in your **config.sys** file. Place this line in your **config.sys** file, immediately after the device driver for the Bernoulli drive:

```
DEVICE=C:\DRDOS\PCKWIK.SYS
```

Not enough conventional memory

If your memory-resident applications give you errors indicating there is not enough conventional memory to run them, you can run Super PC-Kwik using the least amount of conventional memory. We suggest that you use either /A or /E to choose expanded or extended memory for the cache. If upper memory is available, Super PC-Kwik defaults to using it, so you need not specify anything on your command line to take advantage of this memory-saving feature.

When the cache is in expanded memory and has upper memory blocks available, it uses no conventional memory. When the cache is in extended memory, it minimizes the use of conventional memory by default.

If you are willing to give up a little performance, you can reduce the amount of conventional memory used by specifying a track buffer of 4 or 8, using the /T:*nn* option.

The DR DOS BUFFERS command allocates memory for buffers to speed up disk accesses. Many application programs recommend that you set the DR DOS BUFFERS command for 15 or more buffers, each of which may use conventional memory. Super PC-Kwik makes disk accesses fast enough that you won't usually need as many buffers as this. You can do a little experimenting with the applications you use most to determine the number of buffers needed to achieve the best performance while using the least conventional memory.

We recommend setting BUFFERS equal to 3 or 4, unless you are using Windows, in which case you may wish to increase the number of buffers to 10; we recommend you experiment to see what produces the best performance with the least amount of memory.

Warnings

Super PC-Kwik displays warning messages when the first sector of the drive or partition has an invalid signature. When the boot sector has a valid signature, Super PC-Kwik reads certain drive parameters from the boot sector and compares them to values obtained previously from the BIOS or from previous disk partitions. Some format programs may not put IBM-standard signatures in the boot sector even though they otherwise format the disk correctly, and the information obtained from the boot sector does not agree with the information obtained from the BIOS. Using the /G option determines which set of information Super PC-Kwik uses.

Super PC-Kwik automatically pauses when a warning message displays and waits for you to strike a key before it continues loading. Use the /K- option when invoking Super PC-Kwik if you want to it to continue loading without pausing.

uperStor (SSTOR.EXE)

The SuperStor program (and device driver) compress data for more efficient storage on hard disks. Using SuperStor, you can considerably increase the capacity of your hard disk, depending on the type of files on the disk. Once SuperStor is installed and running, no additional steps are required; you use compressed drives just like you used them before installing SuperStor. Data is automatically compressed when you write to the drive and decompressed when you read from the drive. You can create up to 8 SuperStor partitions on your system. The maximum size for a SuperStor partition is 512 Mbytes.

Compression rates

The following are some typical compression ratios achieved by SuperStor:

Type of Data	Ratio
Executable program files	1.4:1 to 2:1
Word processing files	2:1 to 4:1
Database files	2:1 to 8:1
Spreadsheet files	2:1 to 4:1
Video image files	2:1 to 8:1
CAD/CAM	3:1 to 8:1

In addition to compressing the data in files, SuperStor can save space on your disk because space is reserved in sector units (usually 512 byte) while a standard DOS disk reserves space in 2048 to 4096 byte units. This means that on files that are less than 512 bytes long, SuperStor will use 4 to 8 times less space on your disk, regardless of the type of data in the file.

Preparing for SuperStor

❏ Backup all important data on the disk drives you intend to compress with SuperStor. Converting your disk data to compressed data involves altering the structure of portions of your disk drive. While this process has been made as safe as possible, a disk or power failure during critical portions of the conversion process could result in loss of data. Use the DR DOS BACKUP command; see Chapter 10, "DR DOS commands".

❏ If you have any copy-protected software on a drive you intend to prepare as a SuperStor drive, remove the software and reload it after preparing your SuperStor drive.

❏ Remove any WINDOWS permanent swap files.

❏ Verify the integrity of your system; in order to successfully use SuperStor, the internal structures of your hard disk must be consistent. Run the DR DOS command CHKDSK and correct any errors found. Refer to the CHKDSK command description in Chapter 10, "DR DOS commands".

❏ Exit any operating environment (Windows, for example) before using SuperStor. Re-enter your operating environment after preparing your SuperStor drives and use your disks just as you did before.

❏ The conversion process requires a minimum of 1½ Mbyte of free space on the disk. If there is not enough free space, SuperStor will abort. Use the DR DOS DIR command to verify that at least 1½ Mbyte of free space remains before starting the conversion process.

Installing SuperStor

The DR DOS INSTALL and SETUP programs will optionally load the SuperStor driver and place the appropriate statements in your config.sys file. Specify that you want the SuperStor driver (SSTORDRV.SYS) loaded when you are prompted for the DiskMAX options during SETUP; SSTORDRV.SYS must be loaded before you can use SSTOR.EXE or access SuperStor drives. Once you have used SETUP to load the driver, it will automatically be loaded every time you start DR DOS. Refer to the description of installable device drivers in Chapter 11, "Customizing your system".

information obtained from the BIOS. Using the /G option determines which set of information Super PC-Kwik uses.

Super PC-Kwik automatically pauses when a warning message displays and waits for you to strike a key before it continues loading. Use the /K- option when invoking Super PC-Kwik if you want to it to continue loading without pausing.

13.3 SuperStor (SSTOR.EXE)

The SuperStor program (and device driver) compress data for more efficient storage on hard disks. Using SuperStor, you can considerably increase the capacity of your hard disk, depending on the type of files on the disk. Once SuperStor is installed and running, no additional steps are required; you use compressed drives just like you used them before installing SuperStor. Data is automatically compressed when you write to the drive and decompressed when you read from the drive. You can create up to 8 SuperStor partitions on your system. The maximum size for a SuperStor partition is 512 Mbytes.

Compression rates

The following are some typical compression ratios achieved by SuperStor:

Type of Data	Ratio
Executable program files	1.4:1 to 2:1
Word processing files	2:1 to 4:1
Database files	2:1 to 8:1
Spreadsheet files	2:1 to 4:1
Video image files	2:1 to 8:1
CAD/CAM	3:1 to 8:1

In addition to compressing the data in files, SuperStor can save space on your disk because space is reserved in sector units (usually 512 byte) while a standard DOS disk reserves space in 2048 to 4096 byte units. This means that on files that are less than 512 bytes long, SuperStor will use 4 to 8 times less space on your disk, regardless of the type of data in the file.

Preparing for SuperStor

- ❑ Backup all important data on the disk drives you intend to compress with SuperStor. Converting your disk data to compressed data involves altering the structure of portions of your disk drive. While this process has been made as safe as possible, a disk or power failure during critical portions of the conversion process could result in loss of data. Use the DR DOS BACKUP command; see Chapter 10, "DR DOS. commands".
- ❑ If you have any copy-protected software on a drive you intend to prepare as a SuperStor drive, remove the software and reload it after preparing your SuperStor drive.
- ❑ Remove any WINDOWS permanent swap files.
- ❑ Verify the integrity of your system; in order to successfully use SuperStor, the internal structures of your hard disk must be consistent. Run the DR DOS command CHKDSK and correct any errors found. Refer to the CHKDSK command description in Chapter 10, "DR DOS commands".
- ❑ Exit any operating environment (Windows, for example) before using SuperStor. Re-enter your operating environment after preparing your SuperStor drives and use your disks just as you did before.
- ❑ The conversion process requires a minimum of 1½ Mbyte of free space on the disk. If there is not enough free space, SuperStor will abort. Use the DR DOS DIR command to verify that at least 1½ Mbyte of free space remains before starting the conversion process.

Installing SuperStor

The DR DOS INSTALL and SETUP programs will optionally load the SuperStor driver and place the appropriate statements in your config.sys file. Specify that you want the SuperStor driver (SSTORDRV.SYS) loaded when you are prompted for the DiskMAX options during SETUP; SSTORDRV.SYS must be loaded before you can use SSTOR.EXE or access SuperStor drives. Once you have used SETUP to load the driver, it will automatically be loaded every time you start DR DOS. Refer to the description of installable device drivers in Chapter 11, "Customizing your system".

About disk compression

You can use SuperStor to compress an entire existing disk partition, in which case all existing files on the drive are converted into compressed format. You can also leave a specific amount of uncompressed space on the drive, for data that you do not want stored in compressed format. In either case, you use your disks exactly as you did prior to preparing them with SuperStor. Remember that you can run SuperStor to convert uncompressed drives to SuperStor drives at any time; you do not need to initially prepare all your existing partitions as SuperStor drives.

Note

If you convert your C: drive to a SuperStor drive, you can no longer uninstall DR DOS with the UNINSTAL command.

Running SuperStor

Start the SuperStor program by typing the following in the DR DOS command line:

```
C>SSTOR
```

When you press **Enter**, you see a full-screen menu that allows you to select one of the following options:

Option	Description
Prepare	To format and initialize a compressed fixed partition.
Remove	To remove a SuperStor compressed drive.
Statistics	To display statistics about SuperStor drives.
Help	To display online help about a selected item.
Exit	To return to the DR DOS prompt.

Use the following keys to make selections on the SuperStor menu:

- ❑ The arrow keys to select from a list of choices presented in menu and selection boxes.
- ❑ The **Enter** key to execute the currently selected list choice or end other program data input.
- ❑ The **Esc** key when you are prompted for input and decide that you do not want to continue. **Esc** is also used to return from a menu to the previous menu.
- ❑ **F1** to display help on selected operations.

Preparing a hard disk for compression

Prepare option

To prepare a drive for use as a compressed drive, select the Prepare option on the SuperStor menu. SuperStor converts the existing, uncompressed partition of your choice (drive C:, for example) to a SuperStor drive. After converting an existing drive to a SuperStor drive using this option, you will use the drive exactly as you did before using SuperStor. The only difference is that every time data is written to the drive, it is automatically compressed by SuperStor and every time data is read from the drive it is automatically decompressed.

If there is more than one hard disk partition on your system, the SuperStor program will ask you to select the drive you want to convert to a compressed partition. The available drives will be listed for you to select using the arrow keys on your keyboard. You can choose any drive listed.

Note

If you choose your bootable C: drive and you intend to directly edit your **config.sys** file, you will need to know how SuperStor saves **config.sys** information in uncompressed format. Read the section in this chapter called "Using your SuperStor disk".

After you have selected the drive to prepare with the arrow keys on your keyboard, press **Enter**. SuperStor then asks you how much space, if any, you want to reserve on the uncompressed drive (to be accessed by a different drive letter). If the drive contains files that need to remain uncompressed in order for specific applications to work properly (such as Windows swap files, for example), you must leave enough uncompressed space on the drive. If you do not have any such files stored on the drive, compress the entire drive.

Note

When you leave part of the disk uncompressed, the uncompressed space will appear as a new drive letter on your system. Refer to the section called "Using your SuperStor disk" for more information about accessing uncompressed disks.

The SuperStor drive preparation program will now convert the drive you selected to a compressed drive. If there are no existing files on the drive, the conversion process will be completed in a few seconds. If, however, you chose to convert existing files, the SuperStor program will verify the structure of the selected disk, convert existing data into compressed format, and defragment all existing files. Depending on the amount of data on the drive you are converting and the speed of your computer, this process will take from a few seconds to several minutes. The completion bars in the display information area indicate how close to completion each operation is.

SuperStor automatically attempts to compress the data on a disk as much as possible. The resulting compression ratio will vary according to the type of data on the disk.

WARNING It is not possible to interrupt the program during the conversion. Aborting the conversion by rebooting or turning off power to your computer will cause loss of data on the drive being converted.

Exit option

When the SuperStor drive preparation is complete, exit the SuperStor program. There are two options on the Exit menu:

- ❑ RESTART automatically reboots your computer. You *must* reboot before DR DOS will recognize an existing drive that has been compressed.
- ❑ EXIT returns you to the DR DOS command prompt but you must manually reboot your system.

Displaying compressed disk statistics

Statistics option

Select the Statistics option on the SuperStor menu to display the compression statistics about a compressed partition. If you have more than one SuperStor drive, you will be prompted to select the drive for which you want the statistics displayed.

The Compression Statistics display shows the logical size of the SuperStor drive, the actual physical size of the partition, the Compression Ratio, and a graph showing the number of sectors stored in each disk cluster.

Statistic	Meaning
SSTOR Bytes Used	The total number of bytes allocated by DR DOS to store data on the compressed drive. This number is calculated by multiplying the number of clusters allocated by the cluster size in bytes.
SSTOR Free (est.)	An estimate of the number of bytes that can be stored on the compressed drive, assuming the new data has the same compression ratio as the data already stored on the drive. This number is the difference between **SSTOR Total (est.)** and **SSTOR Bytes Used**.
SSTOR Total (est.)	The total number of bytes the SuperStor driver estimates can be stored on the compressed drive. This number is initially 2x (3x, etc.) the physical size of the drive. After the drive contains data, **SSTOR Total (est.)** is calculated using the compression ratio of data already stored on the drive.
Actual Total	The physical size of the disk partition.
Actual Bytes Used	The number of bytes used by the driver to store the files on the physical partition. **Actual Bytes Used** are in units of the physical sector size (typically 512 bytes).
Actual Free	The number of bytes left to store data on the physical partition. This number is the difference between **Actual Total** and **Actual Bytes Used**.

Statistic	Meaning (continued)
Compression Ratio	Shows the efficiency of the SuperStor compression algorithm in storing your data. This number is derived by dividing **SSTOR Bytes Used** by **Actual Bytes Used.**
Space Savings	Another way of stating the compression ratio. It is calculated by dividing **Actual Bytes Used by SSTOR Bytes Used** and expressing the result as a percentage. The result of the division operation is then subtracted from 100%.

Removing a compressed drive

Remove option

Select the Remove option on the SuperStor menu to delete a compressed drive. Note that you cannot remove your current drive.

WARNING If you delete a compressed drive, all data contained on the drive will be lost. Make sure you have backed up any data you want to keep before selecting the Remove option.

Using your SuperStor disk

You use a SuperStor fixed partition drive just as you used the drive before running SuperStor. The advantage is that now you can store up to twice as much data on the drive, depending on the compressibility of your files.

Drive letter switching

The only other difference you may need to be aware of is how drive letters can change.

When you compress an entire drive, without leaving any uncompressed space, the drive letter after you restart the system is the same as before.

When you compress a drive and leave space for uncompressed data, the original drive will appear to be split into two separate drives: one compressed drive (accessed with the original drive letter) and one uncompressed drive, which is assigned a new drive letter. For example, if you convert drive D: to a SuperStor drive but leave space for uncompressed files, the result is a compressed drive D: and an uncompressed drive E:, to contain the uncompressed data. Reassignment of drive letters is done automatically by the SETUP program, and each time you start DR DOS.

Notes

The device driver DEVSWAP must be loaded after SSTORDRV.SYS to change the drive assignments. This will be done automatically by the SETUP and SSTOR programs.

When you compress your bootable drive (usually C:), uncompressed space is *automatically* reserved for the **config.sys** information needed by the operating system to boot. The only situation in which you need to be aware of this is when you want to manually edit your configuration file, to load a driver before SSTORDRV.SYS for example. SuperStor copies configuration statements (to load the memory manager and SuperStor device drivers, swap drive letters, and to "chain" to the **config.sys file** in the compressed drive) into a file called **dconfig.sys** on the original uncompressed drive. When you want to edit this data, you can only access it in the uncompressed area created by SuperStor, and with a different drive letter.

Also note that when you use applications that create Fixed length files, unexpected errors may occur while working to a file on a SuperStor drive, because the amount of free space on the drive decreases as real data is written to the file.

Notes about compatibility

Compatibility between operating systems

You cannot access a SuperStor partition created in DR DOS from other non-DOS operating systems.

Using SuperStor with Windows

SuperStor is compatible with Microsoft Windows. However, SuperStor is not a Windows program. You should run SuperStor from the DR DOS prompt and not from within Windows. You can save Windows programs and files to a SuperStor compressed partition but permanent Windows swap files cannot be stored on SuperStor volumes.

Setting up code page switching

14

DR DOS allows you to use the range of characters required in a number of national languages, by interpreting the different language-specific characters using a mechanism called code page switching.

Usually you only need to set your computer up for code page switching if you:

- ❑ Write documents which are to be used on a computer in a different country.
- ❑ Receive foreign language documents in which some letters become unintelligible characters when displayed or printed.
- ❑ Use an application that is difficult to read on the screen because some graphics characters, such as boxes, appear as letters.

If you do not meet any of these situations, you probably do not need to use code page switching, or to read the information in this chapter.

14.1 What does code page switching mean?

When you press a key on a keyboard (a letter, symbol or number) the computer receives a numeric code that represents the way in which that keystroke is to be displayed, printed and stored.

Up to 256 codes are available to represent lower and upper case letters, numbers, punctuation marks and all the mathematical symbols on your keyboard. This may seem to be more than enough, but it is not possible to represent all the letters and characters used in every language in one character set of 256 codes. For example:

- ❑ Some languages have a much larger alphabet than others and include many accented characters. On most keyboards these accented characters are produced by adding extra single keystroke characters. These key combinations are known as *dead keys.*
- ❑ Many graphics characters such as ╙ and ╓ can be entered.

Character sets

To accommodate these differences, a number of character sets are available. The numeric code entered when you press a key represents a different character depending on which character set is in use.

The most common character set (known as 437) caters for several languages including English, German and French. Character sets 865 and 860 contain the characters needed for Danish and Portuguese respectively. In these character sets, some symbols available in 437 have been removed to make room for the extra letters needed. For example, the code for *f* (franc) has been used for Ó in 860.

Definition of a code page

The character sets are stored in a table called a *code page*. DR DOS allows you to change the code page that your computer uses, so allowing you to view in its original form a document created using a different code page. Appendix B, "Character sets for code pages supported by DR DOS", lists the characters available from different code pages.

Hardware code page

The hardware code page is the standard code page built into your computer. Additional code pages are provided in the Code Page Information or **.cpi** files in your software, for example Code Page 860 for Portugal. Table 14-1 shows the national prepared code pages supported by DR DOS. There is also an international code page, Code Page 850, that can be used in any of the countries (see Table 14-1).

Table 14-1 National Prepared Code Pages

Country	Code pages used	
Australia	437	850
Belgium	437	850
Canada (French)	863	850
Denmark	865	850
Finland	437	850
France	437	850
Germany	437	850
Hungary	852	850
Israel	862	850
Italy	437	850
Latin America	437	850
Middle East	864	850
Netherlands	437	850
Norway	865	850
Portugal	860	850
Russia	866	850
Spain	437	850
Sweden	437	850
Swiss (French)	437	850
Swiss (German)	437	850
Turkey	857	850
United Kingdom	437	850
United States	437	850

Code Page 850

To simplify the exchange of computerized documents between countries, the International Standards Organization (ISO) defined a new code page called 850, for use across national boundaries. This standard, ISO 8859/1, can be used instead of local code pages and reduces the need for code page switching because the sender and the recipient always use the same code page.

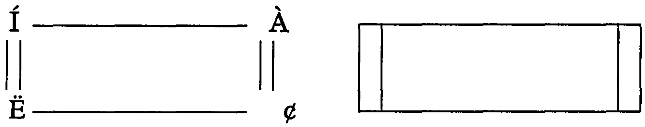
Limitations of Code Page 850

In Code Page 850 it was not possible to keep the original code for every character in every local code page. For example, two characters in 437, the Yen ¥ and Cent ¢ signs, have been moved and some graphics characters have been removed to make room for characters frequently used in the other languages.

In most cases you should be able to use either your local code page or the international Code Page 850, but there may be occasions when you need to switch code pages while you work.

Some applications are written specifically for Code Page 437, especially those that use line-drawing characters. These applications may produce displays which are difficult to read when run under Code Page 850. Typically graphics corners and joints become accented letters.

The following is an example of a display that was intended for use in Code Page 437 and how it looks in Code Page 850.



Code Page 850 Code Page 437

If you are running an application that uses line-drawing characters that are not available in the code page delivered on your computer, run the application under the code page in which it was written. By switching to the original code page, you get a display that is easier to read.

In summary

- ❑ If the local code page is not supported by your hardware, you can use code page support to remedy this.
- ❑ If you need to work in more than one language, use Code Page 850 for all new work.
- ❑ If you receive a document in a language that was created in a local code page, use that code page. If your local language is not covered by a specific code page, use Code Page 850.
- ❑ If you use an application that was written for a particular code page, or that uses line-drawing characters, use the code page that it was written in.

Hardware supported

DR DOS supports code page switching on EGA and VGA displays and on the following printers:

- ❑ IBM Proprinter (4201) & Proprinter XL (4201).
- ❑ IBM Proprinter X24 (4207) & XL24 (4208).
- ❑ IBM Quietwriter III (5202).
- ❑ Epson FX850 and FX1050 printers.
- ❑ Printers compatible with the printers listed above.

14.2 Setting up for code page switching

You can set up your system initially to allow code page switching either when you install DR DOS or later using the SETUP program (see Chapter 11, "Customizing your system"). Either way you configure your system to use code page switching simply by answering a few questions. You need to:

- ❑ Say that you want code page switching.
- ❑ Define which printer(s) will be configured for code page switching if required.
- ❑ Select the code page DR DOS is to use as the default (that is, the one that will be active when the computer is switched on).

Just follow the on-screen prompts provided.

When you finish installation make sure all printers supporting code page switching are switched on and online before you reboot your system.

When you set up code page switching certain commands are automatically placed in the **config.sys** and **autoexec.bat** files, reflecting the parameters that you selected. The keyboard, display and printer are set up to use code page switching each time the system is turned on. Fonts for the default code page (which are held in the **.cpi** files) are downloaded to all the devices that support code page switching.

14.3 Example config.sys and autoexec.bat files

This section provides an example of a **config.sys** file and **autoexec.bat** file for a system with one printer set up to use Code Page 850 as the default code page.

Note

The hardware code page (that is the character set available as standard with your computer), is always prepared for use. The INSTALL program and SETUP configure your system to allow one alternative code page. The example files described in this section are set up to allow you to switch from Code Page 850 to the hardware code page simply by entering the CHCP (Change Code Page) command at the command prompt (see section 14.4, "Working with code pages" for a description of CHCP).

A brief explanation of the commands is given below the example. There is a detailed description of all the parameters in Chapters 10, "DR DOS commands" and 11, "Customizing your system".

Example config.sys file

```
SHELL = C:\DRDOS\COMMAND.COM /P /E:512
BREAK = ON
BUFFERS = 15
FILES = 20
FASTOPEN = 512
LASTDRIVE = E
HISTORY = ON, 256
COUNTRY = 044,850,C:\DRDOS\COUNTRY.SYS
DEVICE = C:\DRDOS\DISPLAY.SYS CON=(EGA,,1)
DEVICE = C:\DRDOS\PRINTER.SYS LPT1=(4201,,1)
```

COUNTRY.SYS, PRINTER.SYS and DISPLAY.SYS are the drivers that support code page switching.

COUNTRY

Sets up the country information. The code page parameter allows the system to convert characters from upper to lower case correctly (and vice versa) and to sort them in the correct order.

When you set up your system for a national language, DR DOS checks the screen, printer and keyboard codes you asked for to see if they will work together. The valid combinations of prepared code pages, country and keyboard codes are defined using the KEYB command and **country.sys** file. See the entries for these commands in Chapters 10, "DR DOS commands" and 11, "Customizing your system" for the valid code combinations.

DEVICE

Loads the driver for the display/printer. The drivers define the number of code pages (in addition to the hardware code page) that can be used (1 in the example). Memory is set aside for each of these code pages, reducing the memory available to run applications. Therefore, although the maximum is twelve, using a smaller number is more efficient.

Example autoexec.bat

```
@ECHO OFF
PATH C:\DRDOS
VERIFY OFF
KEYB UK+,850
MODE CON: CODEPAGE PREPARE=((850) C:\DRDOS\EGA.CPI)
MODE LPT1: CODEPAGE PREPARE=((850) C:\DRDOS\4201.CPI)
NLSFUNC
CHCP 850
```

KEYB, MODE, NLSFUNC and CHCP are the commands that support code page switching.

KEYB

Sets the keyboard nationality and the code page that will be used for the keyboard.

MODE device PREPARE

Prepares a code page for a display or printer. Code pages must be prepared before they can be selected. Several code pages can be prepared for one device in one statement, the maximum being the number defined in the DEVICE = statement.

You do not need to prepare the hardware code page. Note that there are other MODE code page options, described in the next section.

NLSFUNC

Provides support for extended country information and enables you to use the CHCP command to select code pages for all devices set up to use them.

CHCP

Selects a code page for all the devices that are set up for code page switching. In the example, 850 will be the active code page when the system is turned on. The keyboard, display and printer will be set up for Code Page 850.

14.4 Working with code pages

Changing between code pages

When you have set the system up for code page switching, it is simple to switch from the default code page to any other that was prepared (that is, the hardware code page and any other code page defined in a MODE PREPARE command). You do this using the CHCP command.

Loading the NLSFUNC utility

However, before you can use CHCP, the utility NLSFUNC must be loaded. This can be done at installation. Typing NLSFUNC on its own at the command line loads the utility if it is not already installed.

To change the code page on all devices, from the command prompt type:

```
CHCP cp
```

where *cp* is the number of the code page that you want to use now. When the operation is complete, the command prompt is returned.

The code page is changed on all the devices (displays, keyboards and printers) that were defined as supporting code page switching in the **config.sys** file.

Note

All printers supporting code page switching must be switched on and be online when the CHCP command is used. The system may take some time to report an error if the printer is not switched on or is off-line.

Using MODE SELECT

You can use the MODE SELECT command to change the code page on a single device. If you do this, however, you will find that the country information and some other information used by the device is set up differently to the rest of the system.

Changing a code page temporarily

You may find the MODE SELECT command useful if you want to change a code page temporarily on only part of the system. For example, you may want to display a document in a local code page without printing the document.

Listing your current code pages

You can display currently prepared and selected code pages for screens or printers using the MODE command as follows:

```
C>MODE device CODEPAGE [/STATUS]
```

The selected code page number is displayed, followed by a list of hardware code pages defined in the **config.sys** file and code pages prepared using the MODE PREPARE command.

For example if you type:

```
C>MODE lpt1 CP
```

You see a display similar to this:

```
Active codepage for device LPT1 is 850

hardware codepages:
codepage 437
prepared codepages:
codepage 437
codepage 850

Completed codepage status function
```

For more information see Chapter 10, "DR DOS commands".

Refreshing lost code pages

If you select a code page for a printer, then turn the printer off, the code page information may be lost. If this happens, you can reload the fonts for the active code page on to the printer by using the MODE CODEPAGE REFRESH command:

```
C>MODE device CODEPAGE REFRESH
```

For example:

```
C>MODE lpt1 CODEPAGE REFRESH
```

refreshes the code page on the printer connected to port LPT1. For more information see Chapter 10, "DR DOS commands".

A

Error messages and troubleshooting

This appendix is divided into two sections:

- A.1 Error messages. An explanation of the DR DOS error messages.
- A.2 Troubleshooting. What to do if you have problems.

A.1 Error messages

This section lists DR DOS error messages in alphabetical order, giving the origin of the message followed by an explanation of it and where possible, suggestions for correcting the error.

This appendix contains the majority of error messages you are likely to come across, but is not exhaustive. However, if you encounter an error message not listed here, it may well have been generated by one of your applications; consult your applications' documentation.

Note that DR DOS commands have a /? or /H(elp) option, which displays a screen of help text on the command; see Chapter 10, "DR DOS commands".

Reformatting

In some cases this appendix suggests reformatting and/or repartitioning as a remedy for some hard disk error conditions. Keep in mind that such action is an *extreme corrective measure*, because it destroys ALL the files within the partition. If possible, back up all the files within the partition first.

Performing a system reset

Error conditions that require you to perform a system reset, by pressing **Ctrl+Alt+Del**, are rare. If such a condition occurs, make every effort to leave your program normally before resetting your system. If you have no alternative but to reset your system, try to close any open files before doing so.

```
/4 requires a high density 5.25 inch drive
```

Origin FORMAT

Explanation You have specified the /4 switch on a drive which is not high-density (1.2 Mbytes); see Chapter 10, "DR DOS commands", for a description of the FORMAT command switches.

```
/A+ parameter ignored -- conflicts with /E+
```

Origin SUPERPCK

Explanation You may not select both expanded memory (/A+) and extended memory (/E+). Super PC-Kwik ignores the error, and continues to load.

```
/E+ parameter ignored -- not 80286/80386 system
```

Origin SUPERPCK

Explanation Support of extended memory is only provided on IBM PC ATs and other systems with an 80286 or 80386 processor and BIOS support for extended memory. Super PC-Kwik ignores the error, and continues to load.

```
/F Flush request ignored -- nothing to flush
```

Origin SUPERPCK

Explanation You have requested that the cache be flushed when loading the Super PC-Kwik program. This parameter is used to flush the cache after you have loaded the program.

```
/S option may not be used with this operating system
```

Origin FORMAT

Explanation The /S switch, which copies the DOS system files, cannot be used with the operating system you are currently running.

`/U parameter ignored -- PC-Kwik is not loaded`

Origin	SUPERPCK
Explanation	You have requested that the Super PC-Kwik program be unloaded when it currently is not loaded. Super PC-Kwik ignores the error, and continues to load.

/-d Drive *d* cannot be cached -- physical unit unknown

Origin	SUPERPCK
Explanation	If drive *d* is a RAM disk, Super PC-Kwik is simply informing you that it cannot cache a RAM disk. There is no error to correct. If drive *d* is a remote drive in a network, it is inappropriate to cache this drive. You may want to install a cache on the system with the remote drive. Super PC-Kwik ignores the error, and continues to load. If drive d is installed in your system, it is installed in such a way that DR DOS accessses it without using the BIOS. Super PC-Kwik ignores the error, and continues to load.

/-d Drive *d* cannot be cached; maximum drive number exceeded

Origin	SUPERPCK
Explanation	The Super PC-Kwik program supports up to eight hard disk drives. With most drive partitioning schemes, this does not affect the number of partitions per drive. If you receive this message, your system has more than eight drives; only the first eight are cached.

/d Drive *d* cannot be cached -- sector size incompatible

Origin	SUPERPCK
Explanation	Drive *d* has a sector size that is not a multiple of 512 bytes. Super PC-Kwik ignores the error, and continues to load.

```
Abort, Retry, Ignore, Fail ?
```

Origin Several commands

Explanation Typically this message follows an error in a read or write operation. You can respond by typing **A**, **R**, **I** or **F**.

If the reason for the error is obvious (such as no diskette in the target drive) then correct the error and type **R** to try again. **A** (for abort) stops the current operation. **I** results in the next stage of the current operation continuing, as if the error had not been detected. **F** lets the current operation know that there is an error, and it may therefore continue or stop. In both the last two cases there is a risk of data being corrupted.

```
Access denied
Access denied for file
Access denied on source directory
Access to file ... denied
```

Origin Several commands.

Explanation The file is probably either a read-only file or a password-protected file.

Read-only files can be changed to read-write via the ATTRIB command. To access a password-protected file you need to type the filename followed by a semicolon and then the password.

```
Advanced support request ignored for Drive d
```

Origin SUPERPCK

Explanation You have requested advanced support for your hard disk drives with the /H+ option. Your hard disk controller appears not to be IBM compatible. Generic support is used for this drive. Super PC-Kwik ignores the error, and continues to load.

```
Attribute A,H,R or S expected to follow ....
```

Origin ATTRIB, XCOPY, XDIR

Explanation You need to name the attribute explicitly when you type the command line.

Retype the command line and include a letter to name the attribute. You can find out what the letters stand for by typing the command name followed by the /? or /H(elp) option.

```
BACKUP file sequence error
```

Origin RESTORE

Explanation You have probably inserted a diskette from the wrong set of diskettes. If you have two sets of backup diskettes you may have restored from diskette 1 in the first set and then tried diskette 2 from the second set.

Try again with the correct diskette.

```
Bad command or parameter; press ? for help
```

Origin SID

Explanation You have typed an incorrect command line. If you enter ? you will get a list of the valid SID commands. Typing ?? gives a more detailed listing with command syntax.

```
Bad file name
```

Origin SID

Explanation A filename in an E, R, or W command is incorrectly specified.

```
Bad media type
```

Origin Any application

Explanation The diskette you have inserted into the diskette-drive is not of the expected format.

Try another diskette.

```
Bad or missing file
```

Origin — DR DOS startup procedure.

Explanation — DR DOS either cannot find or cannot read the **config.sys** file on the root directory of your startup disk.

Alter the **config.sys** file and then restart DR DOS. See the description of **config.sys** in Chapter 11, "Customizing your system".

```
Bad track detected while formatting, aborting
```

Origin — DISKCOPY

Explanation — A bad track has been detected on the disk. The disk cannot be used for DISKCOPY, but you may be able to use it for other purposes.

```
Bad track in system space, disk unusable
```

Origin — FORMAT

Explanation — The disk you have tried to format is faulty or is of the wrong type.

Use another disk.

```
Batch file ... missing retry (Y/N)?
```

Origin — Batch files

Explanation — DR DOS was unable to find the batch file that it was in the midst of processing. The batch file may have been deleted by one of the commands it contained.

Check your backup copy and correct the command that caused the problem.

```
Batch files nested too deep
```

Origin Batch files

Explanation One batch file can be called from within another batch file using the CALL command. The first batch file is then said to be *nested* within the second. This error message tells you that too many CALL statements have been made within the file.

Check the batch files and correct the command causing the error.

```
Binary reads from a device are not allowed
```

Origin COPY

Explanation You have attempted to copy from a device using the binary switch /B.

Re-enter the COPY command without the /B switch. See Chapter 10, "DR DOS commands" for a full explanation of the COPY command.

```
Cache size requested too small
```

Origin SUPERPCK

Explanation You have requested a cache size less than the minimum supported by Super PC-Kwik.

```
Cannot access drive
```

Origin BACKUP

Explanation The target disk may not be inserted correctly, or it may be corrupt.

```
Cannot CHKDSK network drive ...
```

Origin CHKDSK

Explanation CHKDSK does not work on networked drives.

```
Cannot close
```

Origin SID

Explanation The disk file written by a W command cannot be closed.

```
Cannot compare two image files
```

Origin DISKCOMP

Explanation You have tried to compare two image files, one of the elements to be compared must be a drive.

```
Cannot copy from image file to image file
```

Origin DISKCOPY

Explanation You have tried to copy an image file to an image file. One of the elements specified must be a drive.

```
Cannot copy to/from a reserved device
```

Origin REPLACE

Explanation REPLACE cannot be used to copy files to devices such as modems or printers.

Use the COPY command or the PRINT command if either of these are appropriate.

```
Cannot open ... (device)
```

Origin FIND

Explanation The FIND command cannot be used with devices. It is only used with files kept on disk.

Use the COPY command to copy the relevant files to disk first, and then perform the FIND.

```
Cannot unload -- other programs above
```

Origin SUPERPCK

Explanation You have requested that the Super PC-Kwik program be unloaded after loading one or more programs above it. You need to exit or unload these programs before unloading the Super PC-Kwik program.

```
Cannot XCOPY to/from a reserved device
```

Origin XCOPY

Explanation You have attempted to XCOPY to a device such as a modem or printer. XCOPY should only be used for copying to disks.

Use the COPY command instead.

```
Can't find file ... in directory ...
```

Origin FC

Explanation The file you specified does not exist in the directory specified.

Check the path that you typed and try again.

```
Can't install: swap file error
```

Origin TASKMAX

Explanation TaskMAX cannot store the swap file on your hard disk as it is full.

Try removing some data from your hard disk.

```
Can't install: the swap path
```

Origin TASKMAX

Explanation The path you specified for the TaskMAX swap file is invalid.

Use SETUP to check the swap path you set.

```
Can't load DOS file
```

Origin DR DOS startup procedure

Explanation The file may be missing or corrupt.

Start DR DOS from your startup diskette. You will then need to copy the system files from that startup diskette back to the hard disk or diskette that failed. You can do this with the FDISK or FORMAT command. Remember, however, that both these commands erase the information already existing on the disk. See the descriptions of FDISK and FORMAT in Chapter 10, "DR DOS commands".

```
CGA display adaptor not found
```

Origin MODE

Explanation You have specified a mode display option that is not relevant to your computer.

```
Codepage not prepared
```

Origin MODE PREPARE

Explanation You have specified an additional code page in the DEVICE command but have not prepared it yet.

To prepare the code page use the MODE PREPARE command (see Chapter 10, "DR DOS commands"). Include the code page in the *cplist* parameter.

```
Codepage operation not supported on this device
```

Origin MODE PREPARE

Explanation: DISPLAY.SYS or PRINTER.SYS is not installed for this device.

Add or correct the DEVICE statements in the **config.sys** file using the SETUP program, or by directly modifying **config.sys**.

`Command or filename not recognized`

Origin: Any command

Explanation: You have typed an invalid command.

Check the spelling (Chapter 10, "DR DOS commands", section 10.2, explains how to recall, edit and reissue commands) and re-enter the command, or read Chapter 10, "DR DOS commands" for a full list of valid DR DOS commands. Also check the path to ensure that the required program, if correctly spelt, should be found.

`Could not create new volume label on destination disk`

Origin: BACKUP

Explanation: The disk is probably physically write-protected. Remove the write-protection, as described in Chapter 1, "Getting started with DR DOS", section 1.1.

`Could not delete destination disk volume label`

Origin: BACKUP

Explanation: The disk is probably physically write-protected. Remove the write-protection, as described in Chapter 1, "Getting started with DR DOS", section 1.1.

`Could not load FORMAT`

Origin: BACKUP

Explanation: **Format.com** is the file containing the DR DOS FORMAT command. BACKUP tried to run FORMAT because it detected that the target diskette was unformatted.

Ensure that the two files **backup.com** and **format.com** can both be accessed by DR DOS before you repeat the command. On a diskette-only computer this may mean having both files present on the same diskette in the drive. Alternatively repeat the BACKUP command with a blank formatted disk in the target drive.

```
Could not update destination disk backup file
Could not update destination disk control file
```

Origin BACKUP

Explanation Your disk may have been corrupted.

```
Could not update log file
```

Origin BACKUP

Explanation The BACKUP log file (if you have chosen to create one) is kept on the root directory of the disk from which you start DR DOS. If you have started DR DOS from diskette, make sure the diskette is in the current drive.

```
Current keyboard does not support this codepage
```

Origin MODE SELECT

Explanation The code page is still selected, but this message is displayed as a warning if the MODE SELECT command was used before the country has been changed on the keyboard with the KEYB command. The display and keyboard are therefore using different code pages. (The keyboard code page is changed automatically, but the country must be changed using the KEYB command.)

Use the KEYB command to set the new country for the keyboard, for example KEYB UK+,437 (see Chapter 10, "DR DOS commands").

```
Current path invalid
```

Origin FIND

Explanation Retype the command with the correct path.

```
Cyclic copy not allowed
```

Origin XCOPY

Explanation You have attempted to copy subdirectories using the /S switch. However, some of the subdirectories that would be copied are named as the target directories to receive the copy.

Copy to different directories, or to a different drive.

```
Data error
```

Origin Any application

Explanation There is an error accessing the device specified.

Check the device (for example make sure it is connected properly). If the device being accessed is a disk drive (hard or diskette), this error implies that data on the disk is corrupt. Make backups of all non-corrupt files on the disk, and then either reformat the disk, or use a different disk.

```
Date is not valid
```

Origin BACKUP, RESTORE

Explanation The command cannot be executed with the date you have specified. The date may be typed wrongly, or it may be in the wrong format.

Retype the command with a valid date. You may find the DR DOS DATE command helpful. Using it you can display the DR DOS current date in the preferred format.

`Date must also be specified`

Origin: BACKUP

Explanation: You need to include the relevant date when you specify a time in BACKUP.

`Delete failed`

Origin: XDEL

Explanation: Some or all of the files may be password-protected or read-only.

See the descriptions of PASSWORD and ATTRIB in Chapter 10, "DR DOS commands" for details of handling protected files.

`Destination disk is full`

Origin: RESTORE

Explanation: There is no further room for files on this disk.

Restore the files to a different disk.

`Destination drive is locked by another program`

Origin: DISKCOPY, DISKCOMP

Explanation: Another program has requested sole access and use of the drive specified and has not released it.

`Destination drive type is different`

Origin: DISKCOPY, DISKCOMP

Explanation: The source and destination diskettes you have specified are of different formats and so cannot be copied or compared.

```
Destination file content lost during copy
```

Origin COPY

Explanation You have used the + feature within COPY to try to concatenate a file onto itself.

Copy the original file to a different filename. Then use the + feature to create a new file made from the original file concatenated with the renamed copy.

```
Destination must be a valid disk drive
```

Origin BACKUP

Explanation Valid drives are the physical diskette and hard disk drives. The destination drive must be different from the source drive.

Retype the command with the correct drives.

```
Destination path not found
```

Origin BACKUP, RESTORE

Explanation You have probably mistyped the path. Try typing the command path again.

```
Device error during ...
```

Origin MODE PREPARE, REFRESH, SELECT, or STATUS command

Explanation The printer was off-line or not switched on when the command was given.

Check the printer status before re-entering the command.

```
Disk does not contain UNFORMAT information
```

Origin UNFORMAT

Explanation You have tried to unformat a disk which was not "safe formatted" with the DR DOS FORMAT command. This means that the disk does not contain the information UNFORMAT requires to restore data from it; see Chapter 10, "DR DOS commands".

```
Disk failure, or write protected
```

Origin FORMAT

Explanation Your disk drive has returned an error which may be due to a faulty disk or drive, or the disk may be write-protected. Try another disk or remove the write-protection; see Chapter 1, "Getting started with DR DOS" section 1.1.

Disks are usually write-protected with good reason. Always check the disk's contents before formatting it. Note that some disks do not have a write-protect notch which means that they are *permanently* write-protected. If this is not the problem, contact your dealer.

```
Disk full
```

Origin Any command

Explanation Erase some unwanted files from your disk and try again.

```
Disk is write protected
```

Origin FORMAT

Explanation You have tried to format a diskette which is write-protected. If you want to format the disk remove the write protection; see Chapter 1, "Getting started with DR DOS" section 1.1.

Disks are usually write-protected with good reason. Always check the disk's contents before removing them. Note that some disks do not have a write-protect notch which means that they are *permanently* write-protected.

```
Disk not present, or drive error
```

Origin FORMAT

Explanation You have not inserted a diskette to be formatted or there is a problem with your disk drive.

Try reinserting the diskette, otherwise consult your dealer.

```
Disk read error
```

Origin	SID
Explanation	The disk file specified in an R command could not be read properly.

```
Disk write error
```

Origin	SID
Explanation	A disk write operation could not be successfully performed during a W command, probably due to a full disk.

```
Drive is already locked by another program
```

Origin	FORMAT
Explanation	Another program has requested sole access and use of the drive specified and has not released it.

```
Drive not ready
```

Origin	Any command
Explanation	The diskette is not correctly in the drive, or the drive door is still open.

```
EMM386: A MemoryMAX device driver is already installed
```

Origin	EMM386.SYS
Explanation	You can only have one MemoryMAX device driver installed.

```
EMM386: Another 80386 control program is already active
```

Origin	EMM386.SYS
Explanation	Another program is already using protected mode.

```
EMM386: Another device driver is already using high memory
```

Origin	EMM386.SYS
Explanation	You have already installed a device driver that is using high memory.

```
EMM386: Cannot control address line A20
```

Origin EMM386.SYS

Explanation The driver is unable to enable and disable address line A20.

Contact your dealer.

```
EMM386: Cannot install as EMS is already present
```

Origin EMM386.SYS

Explanation You have attempted to load EMM386.SYS when a driver supporting expanded memory is already loaded.

```
EMM386: Cannot install as XMS is already present
```

Origin EMM386.SYS

Explanation An extended memory manager, such as HIMEM.SYS, is already loaded.

```
EMM386: Cannot let Windows run in protected mode as upper
        and/or EMS memory is in use.
        If standard mode is required, use the /WINSTD
        option of EMM386
```

Origin EMM386.SYS

Explanation EMM386.SYS needs to remain in protected mode, because it has relocated TSRs or drivers to upper memory, or because operating system data structures have been relocated to upper memory (HIDOS=ON). Specify the /WINSTD option in the EMM386.SYS DEVICE line in **config.sys** to start Windows in standard mode.

```
EMM386: Conventional memory is extended into video memory
        space.
        Graphics modes that would use this space are
        disabled.
        The current program is attempting to select one
        of the disabled modes.
        If the program is allowed to continue running,
        memory in video memory space could be corrupted
        by the program.
        If the program is aborted now, the system could
        be left in an unstable condition.
        You are recommended to abort the program and then
        reboot the system.
        Press Esc to abort the program, any other key to
        continue...
```

Origin EMM386.SYS

Explanation EMM386.SYS has been installed using the /VIDEO option, and has extended conventional memory into video memory that is normally used for graphics.

To make sufficient video memory available to the current application, abort the program, use MEMMAX -V to disable the use of video memory.

```
EMM386: Invalid options or parameters specified....
```

Origin EMM386.SYS

Explanation You have incorrectly specified the DEVICE = EMM386.SYS statement in the **config.sys** file.

Retype the statement in the **config.sys** file. The error message includes information about the correct usage. (You can use the SETUP program to install EMM386.SYS device driver.)

`EMM386: Page fault error at ...`

Origin EMM386.SYS

Explanation EMM386.SYS has detected an error in the program you are running. Experiment with different configurations to find one under which the program will run. Note that the source of the error is the program, not EMM386.SYS.

`EMM386: Protection error at ...`

Origin EMM386.SYS

Explanation EMM386.SYS has detected an error in the program you are running. Experiment with different configurations to find one under which the program will run. Note that the source of the error is the program, not EMM386.SYS.

`EMM386: Requires an Intel 80386, 386sx or 486`

Origin EMM386.SYS

Explanation You cannot load the EMM386.SYS device driver on this computer. This device driver requires an Intel 386-SX, 386, i486 or compatible processor.

`EMM386: Requires INT 015h extended memory functions`

Origin EMM386.SYS

Explanation Your system is missing INT 15 functions used by EMM386.SYS. You may have added a 386 board to your computer without loading the driver that supports these functions. The driver is supplied on a diskette with the board.

```
EMM386: The driver can't install as there is no extended
        memory (above 1Mb) in this machine
```

Origin EMM386.SYS

Explanation You cannot load the EMM386.SYS device driver on a computer that does not have extended memory.

```
EMM386: The driver can't install as there is no unused
        64Kb range available for memory management
```

Origin EMM386.SYS

Explanation EMM386.SYS is unable to find sufficient free upper memory to create a LIM page frame. You may be able to create enough space by changing the addresses of network adaptors and other hardware that uses upper memory. You can also use the /LOWEMM option to free more upper memory. If you are still unable to create sufficient space for the LIM page frame, use the /FRAME=NONE option to stop EMM386.SYS attempting to create the page frame.

```
EMM386: The LIM frame address specified via /FRAME=...
        conflicts with other hardware in the machine. Try
        specifying a different value for /FRAME=xxxx, or
        omit /FRAME=xxxx to use automatic scanning
```

Origin EMM386.SYS

Explanation You have attempted to create a LIM frame in an area of memory used by other hardware. Either specify a different address for the frame via the /FRAME= option, or omit /FRAME from the **config.sys** command that loads EMM386. If you omit /FRAME, EMM386.SYS automatically scans upper memory for a free 64Kb block in which to put the LIM frame.

```
EMM386: Warning: Address line A20 already enabled
```

Origin EMM386.SYS

Explanation The driver tested address line A20 and found that the line was enabled. This may indicate that you are already running something that is using high memory. EMM386.SYS will now use the high memory.

```
EMM386: Write attempted to relocated BDOS from ...
```

Origin Several commands

Explanation An application is attempting to write to memory used by the relocated BDOS. You may solve the problem by disabling lower memory. If this has no effect, do not relocate the BDOS.

```
Environment error
```

Origin SET

Explanation DR DOS cannot find the current environment and so cannot change it.

```
Environment full
```

Origin SET

Explanation You have probably tried to add one too many variables to the environment.

Display the current variables by typing the SET command without any options, and then delete any environment variables that you do not need. See the description of SET in Chapter 10, "DR DOS commands".

```
Error accessing physical drive
```

Origin DISKCOPY, DISKCOMP

Explanation The drive you are trying to read from or write to could not be read or written to properly due to a track error on the disk. The disk may have become corrupted.

```
Error on file ...
```

Origin	PRINT
Explanation	PRINT has discovered an error in a file that it was trying to print. Attempt to recover the file to a different area of the disk.

```
Error opening or creating disk file
```

Origin	DISKCOPY
Explanation	The disk you have tried to copy to using DISKCOPY is full or is write-protected. Remove data from the disk, remove the write-protection (see Chapter 1, "Getting started with DR DOS", section 1.1) or use another disk.

```
Error reading configuration file
```

Origin	FILELINK
Explanation	Your configuration file cannot be processed. Check that the format of the file is correct.

```
Error reading filelist ...
```

Origin	Several commands
Explanation	The "filelist" file you specified cannot be processed. Check the format of your filelist. It must be in ASCII text format, use EDITOR to create it, not a word processor. Each filename in the list must begin on a new line. If you still have problems check the integrity of your disk using CHKDSK.

```
Error reading from destination disk
```

Origin	UNFORMAT
Explanation	The disk you are trying to unformat cannot be read due to a track error on the disk. The disk may have become corrupt.

```
Error reading from standard input
```

Origin SORT

Explanation Standard input could not be read.

Check your source file or device for errors.

```
Error reading from the disk
```

Origin DISKCOPY, DISKCOMP

Explanation The disk you are trying to copy or compare cannot be read due to a track error on the disk. The disk may have become corrupt.

```
Error reading from the image file
```

Origin DISKCOPY

Explanation DISCOPY cannot read the image file you are trying to copy. Try the command again. If the problem re-occurs, the image file may have become corrupt.

```
Error reading operating system files
```

Origin FORMAT

Explanation FORMAT has been unable to copy system files to the newly formatted diskette.

Check that the drive door is closed. If it is you may have a defective diskette.

```
Error reading system area
```

Origin FDISK

Explanation This message implies a hardware failure.

Contact your dealer.

```
Error reading system files
Error writing system files
```

Origin	SYS
Explanation	SYS has been unable to copy system files to the diskette. Check that the drive door is closed. If it is you may have a defective diskette.

```
Error writing to critical system area of disk!
```

Origin	UNFORMAT
Explanation	The disk you were trying to unformat, has become corrupt due to a write error on the system track.

```
Error writing to the destination disk
```

Origin	DISKCOPY
Explanation	An error has occured during the copy process. The disk may be write-protected. Remove the write-protection; see Chapter 1, "Getting started with DR DOS", section 1.1.

```
Error writing to the destination image file
```

Origin	DISKCOPY
Explanation	The disk you are copying to is full or a write error has occurred. Remove some data from the disk and try again, or use a different disk.

```
Error writing to the target disk
```

Origin	FORMAT
Explanation	There is an error on the disk in the drive specified or there may be a problem with the drive itself. Try another disk. If the problem remains, contact your dealer.

```
Existing disk format is different -- Continue (Y/N)?
```

Origin FORMAT

Explanation The disk you are attempting to format is already formatted in a different format from the one you specified.

Type Y if you wish to reformat the disk to the new format. Type N to cancel the command and return to the DR DOS command prompt.

```
Expanded Memory Manager Problem; Function = x; Status
Code = y
```

Origin SUPERPCK

Explanation The expanded memory manager returned the error indicated. The memory manager may have also issued a message when you booted your system. You may wish to reboot your system to watch for such a message.

```
Failed to change mode of printer ...
```

Origin MODE

Explanation A MODE change could not be carried out on the printer.

Check that the printer is connected properly, and is online.

```
Failed to get current path on drive...(OS error code ...)
```

Origin Several commands

Explanation This is probably an error on the disk in the drive mentioned or a problem with the disk drive itself.

Consult your dealer.

```
Failed to set default system password
```

Origin PASSWORD

Explanation If you get this error message, make a note of the situation in which it occurred and contact your dealer.

```
FC requires two filenames
```

Origin FC

Explanation You have not specified two filenames to be compared (FC must have two elements to compare one against the other).

Try again, specifying the full filename and filename extension of the two files to be compared. If you are comparing a filelist against a file, you must also specify a filename extension for the filelist.

```
File already open
```

Origin ATTRIB, PASSWORD, REPLACE, XCOPY, XDIR

Explanation The file cannot be accessed by the command you specified because the file is already open. This may mean it is in use by another program already in memory.

Close the file and try again.

```
File contains a line longer than 256 characters
```

Origin FC

Explanation You have tried to compare files that are not in plain ASCII text format (for example, the files were created in word processor format).

```
File list must be the first parameter
```

Origin Several commands

Explanation When you specify a filelist in a command line, it must be the first parameter.

Try again, specifying filelist as the first parameter.

```
Filename too long
```

Origin Several commands

Explanation DR DOS supports filenames up to eight characters with an optional filename extension up to three characters. This extension is separated from the filename by a period (.).

Retype the command with a shorter filename.

```
File not erased
```

Origin ERASE, ERA

Explanation The file may be a password-protected file or a read-only file.

Retype the command but add the correct password after the file specification. Or use the ATTRIB command to change a read-only file into a read-write file.

```
File not found
```

Origin Several commands

Explanation The command cannot find the file specified in your command line.

Check the spelling of the filespec. If the filespec was spelt correctly, use DIR or XDIR to check the contents of other directories for the location of the specified file.

```
File Sharing Conflict
```

Origin Any application

Explanation This error only occurs on network systems. The action you have requested conflicts with the actions of another user.

```
FORMAT not allowed on fixed media
```

Origin BACKUP

Explanation The command you have issued requires BACKUP to format a hard disk. This option is left out of BACKUP so that you cannot accidentally destroy the data on a hard disk.

Backup onto diskettes instead.

```
General failure reported by the drive
```

Origin FORMAT

Explanation Your disk drive has returned an error which may be due to a faulty disk or drive.

Try another disk (if you are formatting a removable disk) or contact your dealer.

```
HIDOS.SYS: A MemoryMAX device driver is already installed
```

Origin HIDOS.SYS

Explanation You can only have one MemoryMAX device driver installed.

```
HIDOS.SYS: A VDISK device is using the High Memory Area
          (segment OFFFFh)
```

Origin HIDOS.SYS

Explanation HIDOS.SYS has attempted to relocate the DR DOS kernel to high memory, but is unable to do so because a memory disk has been set up in high memory.

```
HIDOS.SYS: Conventional memory is extended into video
           memory space.
           Graphics modes that would use this space are
           disabled.
           The current program is attempting to select one
           of the disabled modes.
           Press Esc to abort the program, any other key
           to continue...
```

Origin HIDOS.SYS

Explanation HIDOS.SYS has been installed using the /VIDEO option, and has extended conventional memory into video memory that is normally used for graphics.

To make sufficient video memory available to the current application, abort the program, use MEMMAX -V to disable the use of video memory.

```
HIDOS.SYS: No High Memory Area (segment 0FFFFh) exists
```

Origin HIDOS.SYS

Explanation HIDOS.SYS has attempted to use high memory, but your computer does not support high memory.

```
HIDOS.SYS: No supported chip set present or no shadow RAM.
           You may be able to reconfigure hardware and/or
           install more RAM.
           Many machines can be reconfigured after
           Ctrl/Alt/Del.
```

Origin HIDOS.SYS

Explanation Your computer does not have a chip set supported by HIDOS, or does not have Shadow RAM. Consult your computer's documentation for information about how you can reconfigure your computer hardware.

```
HIDOS.SYS: The specified chip set is not present or there
           is no shadow RAM
```

Origin HIDOS.SYS

Explanation You have specified the wrong chip set in the HIDOS.SYS /CHIPSET option, or your computer does not have Shadow RAM. Consult your computer's documentation for information about the computer's hardware.

```
HIDOS.SYS: This computer appears to have more than one of
           the supported chip sets. Use the /CHIPSET
           option to specify the chip set
```

Origin HIDOS.SYS

Explanation HIDOS.SYS has been unable to identify the chip set in your computer. Use the /CHIPSET option to specify which chip set you have.

```
HIDOS.SYS: Warning: ROM range ... but not write protected
```

Origin HIDOS.SYS

Explanation You have used the HIDOS.SYS /ROM option, and ROM at the addresses specified in the warning, has been copied to RAM but is not write protected.

```
I/O error ...
```

Origin Any application

Explanation An Input/Output error occurred while accessing the requested device. In the case of a diskette-drive, this may mean a corrupt diskette.

`IBMBIO.COM read error`

Origin	Diskette system loader
Explanation	The programs used to startup DR DOS from diskette are corrupt or in the wrong order. Start DR DOS from a different startup diskette. If you want to use the failed startup diskette again you will need to reformat it using the /S (copy system files) switch within FORMAT.

WARNING Reformatting a diskette destroys all the files originally stored on it.

`Illegal decimal value`

Origin	FC
Explanation	You have entered an invalid decimal value after the /M*n* switch, for example /M*xy*. For more information on using this switch, see the explanation of FC in Chapter 10, "DR DOS commands".

`Illegal drive in Path Specification`

Origin	Several commands
Explanation	You specified a drive that is not assigned in your system, or you mistyped the drive specification. Retype the command with a valid drive.

`Image file is corrupt, or not a DISKCOPY image file`

Origin	DISKCOPY, DISKCOMP
Explanation	Either the image file you have specified was not created using DISKCOPY and so cannot be copied or compared using the DISKCOPY and DISKCOMP commands, or the file has become corrupt.

```
Incorrect version of operating system
```

Origin	Several commands
Explanation	The version numbers are different between the DR DOS running on your computer and the file containing the utility that you have just tried to load. The best solution is not to keep any old version of DR DOS on your disks.

```
Input redirection already active
```

Origin	Command line instructions
Explanation	You have tried to start input/output redirection when it is already happening.

```
Insert SYSTEM diskette in A: then PRESS Ctrl/Alt/Del...
```

Origin	DR DOS hard disk system loader
Explanation	This message follows any error message that arises from the hard disk system loader. The effect is to get you to restart DR DOS from your startup diskette.

```
Insufficient disk space for image file
```

Origin	DISKCOPY
Explanation	There is not enough space for the image file on the disk you are copying to. Remove some data from the disk or use a different disk.

```
Insufficient memory
```

Origin	SID
Explanation	There is not enough memory to load the file specified in an R or E command.

```
Insufficient memory available, aborted
```

Origin DISKCOMP, DISKCOPY

Explanation There is not enough conventional memory for the command to work.

Free some conventional memory, for example remove any memory-resident programs you have loaded and try the command again.

```
Insufficient memory for multidisk copy, single copy only
```

Origin DISKCOPY

Explanation You have tried to make multiple copies of a disk, but there is not enough space in memory to store the complete disk image that you want to make multiple copies of. Therefore, you can only make one copy of this disk at a time.

```
Insufficient memory to execute FORMAT/UNFORMAT
```

Origin FORMAT, UNFORMAT

Explanation There is not enough conventional memory for the command to work.

Free some conventional memory, for example remove any memory-resident programs you have loaded and try the command again.

```
Insufficient space to save UNFORMAT information without
destroying some data -- Continue (Y/N)?
```

Origin FORMAT

Explanation There is not enough space on the disk for the "safe format" information to be recorded without overwriting files already there.

If you type Y, you may not be able to recover all the data from the disk using UNFORMAT. Type N, and you are returned to the DR DOS command prompt.

```
Internal error: System tables too large
```

Origin TASKMAX

Explanation The configuration on your computer is too complex, there is not enough memory to describe it and it has exceeded the internal structures created by TaskMAX.

Consult your dealer with details of your configuration, explaining the error message you received.

```
Invalid attempt to both set ... and remove
```

Origin PASSWORD

Explanation You have attempted to set and reset an option at the same time.

Refer to the PASSWORD command in Chapter 10, "DR DOS commands" for the permitted options and then retype the command.

```
Invalid baud rate
```

Origin MODE

Explanation The baud rate specified is not one of the standard baud rates. See the description of MODE in Chapter 10, "DR DOS commands".

Retype the command with a standard baud rate.

```
Invalid break switch (use ON or OFF)
```

Origin The **config.sys** file in the root directory of your startup disk.

Explanation Use the SETUP program to set BREAK to ON or OFF, or directly modify the **config.sys** file, then restart the computer. Refer to the description of **config.sys** in Chapter 11, "Customizing your system" for details of the BREAK command.

```
Invalid category or statement in file
```

Origin TASKMAX

Explanation SETUP automatically creates the TaskMAX-specific configuration file, **taskmax.ini**, and for some reason this file has been configured incorrectly. Delete the **taskmax.ini** file (the default directory where this file is stored is **/drdos**), and use SETUP to reconfigure TaskMAX.

```
Invalid characters in label
```

Origin LABEL

Explanation: Create a different label, using only letters and numbers.

```
Invalid character in password
```

Origin PASSWORD

Explanation: Create a different password, using only letters and numbers.

```
Invalid code
```

Origin GRAFTABL

Explanation: Refer to the description of GRAFTABL in Chapter 10, "DR DOS commands" for the correct code or type the command followed by the /? or /H(elp) option.

```
Invalid combination of options ...
```

Origin REPLACE, XCOPY

Explanation Certain options conflict if they are selected together.

Refer to the command explanations in Chapter 10, "DR DOS commands" for the available options. Type a new command that avoids the conflict.

```
Invalid command in CONFIG.SYS file
```

Origin	DR DOS startup procedure
Explanation	DR DOS does not recognize one of the commands in the **config.sys** file on the root directory of your startup disk. Remove the invalid command from **config.sys**; the valid commands are described in Chapter 11, "Customizing your system". There is no need to restart the computer.

```
Invalid country code
```

Origin	The **config.sys** file in the root directory of your startup disk.
Explanation	Alter the original **config.sys** file by putting in the correct country code for the keyboard you have specified, and then restart DR DOS. See the description of **config.sys** in Chapter 11, "Customizing your system" for a list of the supported country codes.

```
Invalid data in font file filename.CPI
```

Origin	MODE PREPARE
Explanation	There is a conflict between the code page specified in the DEVICE statements and that specified in the MODE PREPARE command. Reload the font file from master disk.

```
Invalid date specified
Enter date: __
```

Origin	DATE, TOUCH
Explanation	Enter the date in the correct format for your country. See the entry for DATE in Chapter 10, "DR DOS commands" for examples of the correct format.

`Invalid destination path`

Origin REPLACE

Explanation Retype the command with the correct path for the destination of the copied files.

`Invalid drive specified`

Origin Several commands

Explanation You specified a drive that is not assigned in your system, or you mistyped the drive specification.

Retype the command with the correct drive specification.

`Invalid filename`

Origin Several commands

Explanation Retype the command with the correct filename.

`Invalid file specification`

Origin PRINT

Explanation Retype the command with the correct path and filename.

`Invalid format for target drive`

Origin FORMAT

Explanation The device driver you are using does not support the parameters you have specified.

```
Invalid hard disk system partition
```

Origin DR DOS hard disk system loader

Explanation The system loader is unable to read information in the Partition Table.

Restart DR DOS from diskette. Use the FDISK utility to recreate the partition on the hard disk.

WARNING Using FDISK will destroy all the data on that partition of your hard disk.

```
Invalid last drive character (use A...Z)
```

Origin The **config.sys** file in the root directory of your startup disk

Explanation Use the SETUP program to amend the LASTDRIVE entry in **config.sys** or modify **config.sys** directly and restart DR DOS. Refer to the description of **config.sys** in Chapter 11, "Customizing your system" for details of the LASTDRIVE command.

```
Invalid lines per inch
```

Origin MODE

Explanation You must specify either 6 or 8 lines per inch for your printer in the MODE command.

```
Invalid line width
```

Origin MODE

Explanation You must specify 80 or 132 characters per line for your printer in the MODE command.

```
Invalid number of buffers
```

Origin — The **config.sys** file in the root directory of your startup disk

Explanation — Amend the BUFFERS entry in **config.sys** using the SETUP program or modify **config.sys** directly, and then restart DR DOS. Refer to the description of **config.sys** in Chapter 11, "Customizing your system" for details of the BUFFERS command.

```
Invalid number of data bits
```

Origin — MODE

Explanation — The databits setting specified is not one of the standard settings (7 or 8). See the description of MODE in Chapter 10, "DR DOS commands".

```
Invalid number of files (use 5...255)
```

Origin — The **config.sys** file in the root directory of your startup disk

Explanation — Amend the FILES entry in **config.sys** using the SETUP program or modify **config.sys** directly, and then restart DR DOS. See Chapter 11, "Customizing your system" on **config.sys** for details of the FILES command.

```
Invalid number of parameters
```

Origin — Several commands

Explanation — You have typed too many, or conflicting, options.

Refer to the description of the command in Chapter 10, "DR DOS commands".

```
Invalid number of stop bits
```

Origin — MODE

Explanation — The stopbits setting specified is not one of the standard settings (1 or 2). See the description of MODE in Chapter 10, "DR DOS commands".

```
Invalid option - decimal number expected
```

Origin	SORT
Explanation	The /+n option of SORT stands for the on-screen column number at which you want the sort to occur. This number must be given as an ordinary number (a 'decimal'), and not as a hexadecimal.

```
Invalid option ... in parameter ...
```

Origin	ATTRIB, PASSWORD, REPLACE, XCOPY
Explanation	You have typed an option that DR DOS does not recognize. Refer to the command description in Chapter 10, "DR DOS commands" for the correct options and then retype the command.

```
Invalid parameter
```

Origin	Several commands
Explanation	You have typed an option that DR DOS does not recognize. Refer to the command description in Chapter 10, "DR DOS commands" for the correct options and then retype the command.

```
Invalid parameter - missing option
```

Origin	FIND
Explanation	You have typed an option that DR DOS does not recognize. Refer to the explanation of FIND in Chapter 10, "DR DOS commands" for the correct options and then retype the command.

```
Invalid parity setting
```

Origin	MODE
Explanation	The parity setting specified is not one of the standard parities. See the description of MODE in Chapter 10, "DR DOS commands".

```
Invalid partition table
```

Origin	DR DOS hard disk system loader
Explanation	More than one partition is marked as bootable (that is able to start DR DOS), a non-existent partition is marked as bootable, or there is a hard disk failure. Restart DR DOS from diskette. Use the FDISK utility to set the bootable partition, see Chapter 10, "DR DOS commands". Note that FDISK formats the disk, and you will lose any data on the disk.

```
Invalid password
```

Origin	Several commands
Explanation	The password you have typed is not correct.

```
Invalid path or path not found
Invalid path specified: check path
```

Origin	Several commands
Explanation	Retype the command with the correct path.

```
Invalid SHELL filename
```

Origin	The **config.sys** file in the root directory of your startup disk
Explanation	Amend the SHELL entry in **config.sys** and restart DR DOS. See also the description of **config.sys** in Chapter 11, "Customizing your system.

```
Invalid switch ... (PRINT already installed)
```

Origin	PRINT
Explanation	The option you chose cannot be set because PRINT is already installed in memory. Carry on without the option, or restart DR DOS and load PRINT again.

```
Invalid time specified
Enter time: __
```

Origin TIME, TOUCH

Explanation You entered the time in an incorrect format.

Enter the time in the format as shown in the screen prompt.

```
Label ... not found
```

Origin Batch file and **config.sys** processing

Explanation A GOTO statement in the batch or **config.sys** file refers to a label that does not exist.

Correct the relevant file.

```
Line too long in file
```

Origin FIND

Explanation You have tried to use FIND on a non-text file. FIND is only designed for text files.

```
Loading conflict
```

Origin Several commands

Explanation There is probably not enough memory for the task you have specified.

Free some memory by removing memory-resident programs that you do not need, or by reducing the amount of memory occupied by memory disks.

```
Log file path not found
```

Origin BACKUP

Explanation You have probably mistyped the path for the log file.

Try again.

```
MEMMAX: Must be run from master command processor
```

Origin MEMMAX

Explanation You cannot disable upper memory with MEMMAX from within another program or application. Run MEMMAX after exiting your current application.

```
Mono display adaptor not found
```

Origin MODE

Explanation The MONO display mode that you have specified is not supported by your display adaptor.

```
Multidisk copies to a file is not allowed, ignored
```

Origin DISKCOPY

Explanation You have specified a file as the destination for multiple copies. This is not allowed because if you make multiple copies to a file, the file contents will be continually overwritten by each new copy made.

```
No backup files present on source disk
```

Origin RESTORE

Explanation You inserted a disk that contained no backup files, or you inserted a backup diskette out of sequence.

```
No Expanded Memory Manager found; Check CONFIG.SYS files.
Check that the most recent expanded memory manager is
installed
```

Origin SUPERPCK

Explanation The expanded memory manager has not been successfully installed. See the documentation for your expanded memory board for information on installing the expanded memory hardware and software.

```
No file
```

Origin SID

Explanation The file specified in an R or E command could not be found on the disk.

```
No files to backup
```

Origin BACKUP

Explanation No files matching the conditions you specified were found. You may have mistyped the file specification.

```
No operating system
```

Origin Hard disk system loader

Explanation This message originates in software that your supplier provides with the hard disk.

Start DR DOS from the diskette. Copy DR DOS onto the hard disk and then use the FDISK utility to make the DR DOS partition "bootable" (that is, capable of loading DR DOS when the computer is started).

```
No space
```

Origin SID

Explanation There is no space in the directory for the file being written by a W command.

```
Non standard sector length - cannot write new boot sector
```

Origin SYS

Explanation You have typed Y when asked whether a new boot sector is to be written. However SYS cannot recognize the disk format, since the sector size is not 512 (or a multiple of 512 for a hard disk).

Backup any data on the disk and reformat it under DR DOS.

```
Not enough disk space
```

Origin FILELINK, REPLACE, SORT, XCOPY

Explanation The disk has too much stored on it. Erase some unwanted files from the disk and try again.

```
Not enough memory
Not enough memory for ...
Not enough memory to ...
```

Origin Several commands

Explanation There is insufficient free memory for the command to work.

Free some memory by creating a smaller memory disk, or by removing memory-resident programs, or device drivers that you do not need. Then try the command again.

```
Not ready error
```

Origin Any application

Explanation The device you are trying to access is not responding. In the case of a diskette-drive, this usually means there is no diskette in the drive, or the drive door is not shut. With hard disks, this error may mean a hardware failure.

Check your diskette is inserted correctly into the diskette-drive, and the door is shut. For hard disks, contact your dealer.

```
Operand
```

Origin SID

Explanation An attempt was made to assemble a command with an ambiguous operand. Precede the operand with the prefix “BYTE” or “WORD”.

```
Operating system error
```

Origin Several commands

Explanation If you get this error message, make a note of the situation in which it occurred, and contact your dealer.

```
Operating system load error
```

Origin DR DOS hard disk system loader

Explanation The computer cannot read the operating system's loader file.

Restart DR DOS from diskette. Use BACKUP to save the files on the partition onto diskette. Use the FDISK utility to delete and then recreate the partition. Then use RESTORE to recover your files.

```
Out of memory
```

Origin FC

Explanation The files you have tried to compare are too large to be compared.

```
Output redirection already active
```

Origin Command line instructions

Explanation You have tried to start input/output redirection with it already happening.

```
Parameter not recognized
```

Origin BACKUP, MODE, RESTORE

Explanation You have typed an invalid option.

Refer to the relevant command description in Chapter 10, "DR DOS commands" for the permitted options.

```
Path not found
```

Origin Several commands

Explanation The path you have typed is incorrect.

Retype with the correct path.

```
Path is too long
```

Origin FIND

Explanation You have tried to specify a path name that is too long.

Retype with the correct path.

```
Path was too long to backup - skipping over
```

Origin BACKUP

Explanation Your hierarchy of subdirectories is too complex.

Simplify the arrangement of subdirectories on your disk and then try BACKUP again.

```
Physical Media Error
```

Origin Several commands

Explanation DR DOS was unable to read from or write to a disk. The disk has probably been corrupted.

Copy as many files as you can from the corrupt disk to another disk and then reformat or dispose of the first disk.

```
Print queue full
```

Origin PRINT

Explanation If the printer is printing, try again later. Otherwise, delete a file that you do not need from the print queue and then try adding your file again.

```
Printer out of paper
```

Origin: Any application

Explanation: The printer you are attempting to access is out of paper.

Refill the paper tray, and try printing again.

Problem with drive *d*. Correct and then press any key

Origin: SUPERPCK

Explanation: If you receive this message for a diskette drive, check to see that there is a diskette in drive *d* and that the drive door is closed. Then, press a key and Super PC-Kwik retries the operation. If the problem reoccurs, the diskette in drive *d* may contain bad sectors. Use CHKDSK to test the diskette. If it confirms that there is a problem with the diskette, format a fresh diskette and copy the files on the defective diskette to the new diskette.

If there appears to be nothing wrong with the diskette, your hardware may be incompatible with the advanced diskette support provided by Super PC-Kwik. In this case, reload Super PC-Kwik with the /D- option.

There is also a remote chance that this error message could appear for a hard disk. If you suspect bad sectors on your hard disk, back up all your files. Then, reformat your hard disk or use CHKDSK to test the disk for bad sectors. (Reformatting detects the bad sectors and prevents them from being used again. CHKDSK detects problem sectors on your hard disk and removes them from further use.) Restore your files from the backup media. Finally, if you are sure that you have no bad sectors being used on your hard disk, specify the /H- option to instruct Super PC-Kwik to use generic support for your hard disk.

```
Program not loaded; invalid parameters. Type "SUPERPCK
/?" for information on parameters
```

Origin	SUPERPCK
Explanation	You used options that are not valid for the Super PC-Kwik program. Use /? for a list of valid options and arguments.

```
Program terminated normally
```

Origin	SID
Explanation	The program running under SID completed, or was terminated by a **Ctrl+Break**.

```
Protection level not changed file has no password
```

Origin	PASSWORD
Explanation	This message informs you that no password is attached to the file you have named in the command line.

```
Read error on file
```

Origin	FIND
Explanation	Your disk may be corrupted. Try again with a backup copy.

```
Resident portion installed
```

Origin	MODE
Explanation	A portion of the MODE.COM file remains in memory once MODE is invoked, slightly increasing the amount of space occupied by DR DOS on your system. This portion is called the *resident* portion.

```
Sector not found
```

Origin: Any application

Explanation: The requested sector on the disk is not present, or is corrupt. With diskette-drives, this usually means that the diskette in the drive has not been formatted.

```
Seek error
```

Origin: Any application

Explanation: The disk drive being accessed was physically unable to move its read/write heads to the requested location. This generally means a disk controller hardware failure.

Contact your dealer.

```
Source and destination cannot be the same file
```

Origin: COPY, REPLACE, XCOPY

Explanation: You have named the same file as both the source file and the destination file for the copying operation.

```
Source and destination drives are the same
```

Origin: BACKUP, RESTORE

Explanation: The destination drive must be different from the source drive (the drive containing the disk to be backed up or restored).

Retype the command with different source and destination drives.

```
Source drive must be specified
```

Origin BACKUP, RESTORE

Explanation BACKUP and RESTORE need both the source drive (the drive containing the disk to be backed up or restored), and the destination drive to be included in the command you type. These drives must be different drives.

Retype the command with the correct drives specified.

```
Source file ... is not a valid source EXE file
```

Origin EXE2BIN

Explanation The file you tried to convert is not a valid executable type file.

```
Source path not found
```

Origin BACKUP, RESTORE

Explanation You have probably mistyped the path.

Try again.

```
Specified /F option is not valid for target drive
```

Origin FORMAT

Explanation The drive type you specified using the /F switch is invalid; see Chapter 10, "DR DOS commands", for a description of the FORMAT command switches.

```
Specified drive is invalid, or non-removable media
```

Origin DISKCOPY, DISKCOMP

Explanation The drive you have specified does not exist, does not have a compatible device driver or is a fixed (non-removable) disk.

```
Specified path is too long
```

Origin Several commands

Explanation You have tried to specify a path name that is too long.

Retype with the correct path.

```
Specify source AND destination drives
```

Origin BACKUP, RESTORE

Explanation BACKUP and RESTORE need both the source drive (the drive containing the disk to be backed up or restored), and the destination drive to be included in the command you type. These drives must be different drives.

Retype the command with the correct drives specified.

```
Super PC-Kwik program already loaded. You must unload
before reloading with different parameters
```

Origin SUPERPCK

Explanation You have requested that the program be loaded when it is already loaded.

```
Switch value out of range
```

Origin PRINT

Explanation The value chosen for the option is out of the allowed range.

Refer to the description of PRINT in Chapter 10, "DR DOS commands" for the correct ranges for each option.

```
SYS is not supported on this operating system
```

Origin SYS

Explanation SYS only runs under DR DOS.

```
TaskMAX: An extended Memory Specification (XMS) driver
         such as HIDOS.SYS should be installed in
         CONFIG.SYS to use extended memory
```

Origin TASKMAX

Explanation You have tried to configure TaskMAX to use extended memory without loading an Extended Memory Specification (or XMS) driver first, for example, EMM386.SYS or HIDOS.SYS.

Use SETUP to configure your computer so that an XMS driver is loaded; see Chapter 11, "Customizing your system".

```
There is not enough free memory in your system. Your
system has nnnnnK bytes of free memory
```

Origin SUPERPCK

Explanation You have selected a cache or reserve size that exceeds the memory capacity of your system. Try a different value for the /S or /R options. If you requested the cache be loaded in conventional memory and you also have extended or expanded memory, try loading the cache in one of these alternate forms of memory using the /A or /E options.

```
Time is not valid
```

Origin BACKUP, RESTORE

Explanation You have probably mistyped the time.

Retype the command with a valid time. You may find the DR DOS TIME command helpful. Using it you can display, and if necessary, change the DR DOS current time.

```
Too few parameters
```

Origin MODE

Explanation You have not typed in enough information.

Refer to Chapter 10, "DR DOS commands" for a description of the MODE command and its various formats and parameters.

```
Too many files of type ... in directory ...
```

Origin FC

Explanation You have tried to compare more than 128 files. FC can only handle 128 files in one operation.

```
Too many files open
Too many open files
```

Origin Several commands

Explanation You need to increase the number of files that can be open at one time. Read about the FILES command in Chapter 11, "Customizing your system".

```
Too many files to sort
```

Origin XDIR

Explanation There is not enough room for XDIR to sort and then display all the files you have specified.

Sort files from a subdirectory so that XDIR is sorting fewer files.

```
Too many parameters
```

Origin Several commands

Explanation You have specified too many, or conflicting, options.

Refer to the description of the relevant command in Chapter 10, "DR DOS commands".

```
Unable to backup due to sharing conflict
```

Origin BACKUP

Explanation The file you specified is currently being used by another process.

Retry the command when the file is not being used.

```
Unable to convert. Initial IP of ... must be 0 or 100h
```

Origin EXE2BIN

Explanation For an **.exe** file to be converted to a **.com** file, its entry point must be at offset 100h within its segment. For an **.exe** file to be converted to a **.bin** file, its entry point must be at absolute offset 0 in its memory usage. If either of these cases is untrue, the conversion cannot be carried out.

```
Unable to convert. Source file ... has a stack declared
```

Origin EXE2BIN

Explanation You cannot convert a **.exe** file to either a **.com** or **.bin** file if the program has a stack statement declared.

```
Unable to convert. Source file ... requires relocation
fixups
```

Origin EXE2BIN

Explanation The destination file is to be a **.com** type, but its header contains relocation items which must be performed.

```
Unable to convert. Source file ... requires too much
memory
```

Origin EXE2BIN

Explanation The destination file is to be a **.com** type but it requires more than 64 Kbytes to run.

```
Unable to create directory
```

Origin FILELINK, MKDIR, XCOPY

Explanation Your disk is probably full, or it is physically write-protected, or corrupt. You can also get this message if you try to create a directory with the same name as an existing directory.

Use DIR to find out the amount of free space on the disk, and then delete unwanted files. Check the disk for write-protection (see Chapter 1, "Getting started with DR DOS", section 1.1). Try creating a different directory name, or create the directory on a different drive.

```
Unable to delete existing volume label
```

Origin LABEL

Explanation The diskette is probably physically write-protected.

```
Unable to find start of program code
```

Origin EXE2BIN

Explanation The source file contained a valid **.exe** header but no valid program code.

```
Unable to open codepage font file
```

Origin MODE PREPARE

Explanation The code page font file specified in the MODE command could not be found.

Make sure that you are specifying the drive, path and file name of the driver correctly.

```
Unable to open or create destination file
```

Origin EXE2BIN

Explanation The program was unable to access the specified destination file. This is probably because there is already a read-only file of the same name.

```
Unable to open source file
```

Origin EXE2BIN

Explanation The specified source file does not exist. Check the name of your source file.

```
Unable to perform refresh
```

Origin MODE REFRESH

Explanation The printer driver does not have a copy of the code page to download to the printer.

Prepare and select the code page. To prevent this error recurring, make sure that the number of additional code pages (n) defined in the DEVICE= PRINTER.SYS statement is one or greater, see Chapter 11, "Customizing your system".

```
Unable to read BACKUP file
```

Origin RESTORE

Explanation There is a problem with the diskette you have inserted. It may be the wrong diskette, or it may have been inserted out of sequence, or corrupted.

Try again with the correct diskette.

```
Unable to read from codepage font file
```

Origin MODE PREPARE

Explanation The complete contents of the font file could not be read because of a disk error that occurred during the MODE command.

Reload the font file from the master disk.

```
Unable to read relocation information
```

Origin EXE2BIN

Explanation The segment fixup information in the **.exe** header of the source file is not valid.

`Unable to restore due to sharing conflict`

Origin RESTORE

Explanation The file you specified is currently being used by another process.

Retry the command when the file is not being used.

`Unable to set volume label`

Origin LABEL

Explanation The diskette is probably physically write-protected.

Remove the write-protection (see Chapter 1, "Getting started with DR DOS", section 1.1), and try again.

`Unable to write to destination`

Origin RESTORE

Explanation The destination disk is probably physically write-protected. Remove the write-protection (see Chapter 1, "Getting started with DR DOS", section 1.1), and try again.

`Unable to write to destination file`

Origin EXE2BIN

Explanation The program was able to open, but could not write to, the destination file. This is probably due to a hardware malfunction.

`UNFORMAT information has been overwritten`

Origin UNFORMAT

Explanation You have tried to unformat a disk to which you have copied new files, consequently the system information required by UNFORMAT to restore data from the disk has been overwritten. You will not be able to unformat the disk.

`Unknown error code`

Origin	Several commands
Explanation	If you get this error message, make a note of the situation in which it occurred, and contact your dealer.

`Unknown switch`

Origin	Several commands
Explanation	You have specified an option that does not exist. Refer to the description of the command in Chapter 10, "DR DOS commands" for a list of the valid options. Then retype the command.

`Unrecognized characters ignored`

Origin	SUPERPCK
Explanation	You have typed a character that is not a recognized option for the Super PC-Kwik program. Super PC-Kwik ignores the error, and continues to load.

`Unrecognized display mode`

Origin	MODE
Explanation	You have probably mistyped the display mode. The MODE command description in Chapter 10, "DR DOS commands" contains a listing of the display options.

`Unrecognized parameter ignored: /x`

Origin	SUPERPCK
Explanation	You have selected an option that is not recognized by the Super PC-Kwik program. Super PC-Kwik ignores the error, and continues to load.

```
Use /F:360 or /F:320 before /1 on high density drives
```

Origin FORMAT

Explanation You have specified the /1 switch when trying to format a diskette in a 1.2 Mbyte drive.

The /1 switch is only valid on 320/360 Kbytes drives unless preceded by /F:320 or /F:360; see Chapter 10, "DR DOS commands", for a description of the FORMAT command switches.

```
Use /F:360 or /F:320 before /8 on high density drives
```

Origin FORMAT

Explanation You have specified the /8 switch when trying to format a diskette in a 1.2 Mbyte drive. The /8 switch is only valid on 320/360 Kbyte drives unless preceded by /F:320 or /F:360; see Chapter 10, "DR DOS commands" for a description of the FORMAT command switches.

```
Use another boot disk
```

Origin Diskette system loader

Explanation The programs used to startup DR DOS from diskette are corrupt, or in the wrong order.

Startup DR DOS from a different startup diskette. If you want to use the failed startup diskette again you will need to reformat it using the FORMAT command with the /S (copy system files) switch. The FORMAT command is explained in Chapter 10, "DR DOS commands".

WARNING Reformatting a diskette destroys all the files originally stored on it.

```
Use both /T and /N parameters to define a custom disk type
```

Origin FORMAT

Explanation You have tried to format a disk into a non-standard format without specifying both /T and /N; see Chapter 10, "DR DOS commands", for a description of the FORMAT command switches.

```
Verify error at s:o
```

Origin SID

Explanation The value placed in memory by a Fill, Set, Move, or Assemble command could not be read back correctly, indicating bad RAM or attempting to write to ROM, or non-existent memory at the indicated location.

```
WARNING: Boot sector for drive d contradicts BIOS; Boot
         information ignored
```

Origin SUPERPCK

Explanation The boot sector for the designated drive contains information that is different to that provided by the BIOS. Since you specified the /G- option, Super PC-Kwik uses the information provided by the BIOS rather than the information provided by the boot sector. If you would prefer to use the information provided by the boot sector, use the default /G+ option. Press any key to continue loading.

```
WARNING: Boot sector for drive d contradicts BIOS; Boot
         information used
```

Origin SUPERPCK

Explanation The boot sector for the designated drive contains information that is different to that provided by the BIOS. If you specified the default of /G+, Super PC-Kwik uses the information provided by the boot sector rather than the information provided by the BIOS. To use the information provided by the BIOS instead, use /G-. Press any key to continue loading.

```
WARNING: Boot sector for drive d invalid; previous
         information used
```

Origin SUPERPCK

Explanation The boot sector for the designated drive could not be used to obtain information about the drive. Super PC-Kwik uses the information provided by the BIOS instead. In some cases, it can use information from previous boot sectors on the same drive as well. Press any key to continue loading.

```
WARNING: Partition x differs from previous; no advanced
         support provided
```

Origin SUPERPCK

Explanation Your system includes a hard disk that is divided into partitions. Super PC-Kwik has discovered that the boot sector for one of the partitions contains information that differs from the information provided by the boot sector for the previous partition. This new information is used for caching the designated partition, but the advanced support feature is not used on the drive containing this partition. Press any key to continue loading.

```
WARNING: PCKWIK.SYS placed after non-disk driver
```

Origin SUPERPCK

Explanation Either your Bernoulli box is not turned on or the driver PCKWIK.SYS is not placed after the Bernoulli driver in your **config.sys** file. Press any key to continue loading.

```
WARNING: SHARE.EXE should be loaded before this program
```

Origin TASKMAX

Explanation The SHARE.EXE (file sharing) program must be loaded before you can use TaskMAX.

Use the SETUP program to alter your computer's configuration so that SHARE.EXE is loaded.

```
WARNING: The Cache is being loaded high by another program
```

Origin SUPERPCK

Explanation HILOAD is trying to put the code for the Disk Accelerator into Upper Memory Blocks. Let Super PC-Kwik load itself into high memory so there won't be any conflict, and so you minimize the amount of high memory being used. Super PC-Kwik ignores the error, and continues to load.

```
WARNING: Using boot sector info. for partition x which
         differs from previous
```

Origin SUPERPCK

Explanation Your system includes a hard disk that is divided into partitions. The boot sector for one of the partitions contains information that differs from the information provided by the boot sector for the previous partition. Super PC-Kwik uses the new boot sector information to cache the designated partition. If you would prefer the information provided by the BIOS be used instead, add the /G- option to the Super PC-Kwik command line. Press any key to continue loading.

```
Write protect error
```

Origin	Any application
Explanation	The diskette to which you are attempting to write is physically write-protected. Remove the write protection (see Chapter 1, "Getting started with DR DOS", section 1.1. if you are not sure about this), or use a different, non-write-protected diskette.

```
Wrong boot sector format for SYS
```

Origin	SYS
Explanation	You may have typed N when asked whether a new boot sector is to be written. SYS only copies files to a disk with a DR DOS boot sector.

```
Wrong version of operating system
```

Origin	BACKUP, MODE, RESTORE
Explanation	The version numbers are different between the DR DOS running on your computer and the file containing the utility that you have just tried to load. The best solution is not to keep any previous version of DR DOS on your disks.

```
XMS driver failure; cannot continue with installation
```

Origin	SUPERPCK
Explanation	Super PC-Kwik has detected an Extended Memory Specification driver in your system and has attempted to use it to enhance access of extended memory. Use HIDOS.SYS or EMM386.SYS.

A.2 Troubleshooting

This section discusses several classes of errors, their causes, and some recovery procedures under the following headings:

□ Troubleshooting checklist

□ Diskette handling

□ Corrupted files

□ When absolutely nothing happens

□ Unresolvable errors

Troubleshooting checklist

This checklist covers some of the more common error conditions.

□ Have you prepared a DOS partition on your hard disk? You cannot access your hard disk until this is done. See Chapter 10, "DR DOS commands" for a description of the disk partitioning utility, FDISK.

□ Is your hard disk unit turned on? An external hard disk unit may have its own electrical connection.

□ Are the peripheral devices connected properly to your computer and to electrical outlets?

□ If a file is set to read-only, you cannot edit the file. Use the ATTRIB command to set the file to read-write, as described in Chapter 10, "DR DOS commands".

□ Did you specify a directory when typing the command line? By default DR DOS looks for the file only in the current directory of the current drive.

□ Is the file password-protected? Refer to Chapters 4, "Protecting your information" and 10, "DR DOS commands" for information on password protection.

□ Are you running a program under the correct operating system version? Many DR DOS commands run only under the operating system with which they are shipped.

Diskette handling

Diskette handling is the source of many errors. Here is a summary of points to check through (see also Chapter 1, "Getting started with DR DOS", section 1.1, which describes how to handle diskettes.)

❑ Is the disk correctly inserted and the drive door closed?

❑ Is the disk the correct type for the drive?

Double-sided drives can read single-and double sided diskettes. Single-sided drives can access only single-sided diskettes.

Some drives are designed to access both 1.2 Mbyte diskettes and 360 Kbyte diskettes. Other drives will only access one of these types. Still others will only access 160 Kbyte and 180 Kbytes diskettes, or 3½" diskettes. The FORMAT command description in Chapter 10, "DR DOS commands" contains full details of diskette types.

❑ Is the disk physically write-protected? See Chapter 1, "Getting started with DR DOS", section 1.1 if you are unsure about write-protection.

❑ If the error is not caused by one of the above problems, the disk might be damaged. Use the COPY command (see Chapter 10, "DR DOS commands") to copy information to a new disk. Replace any files you cannot copy with the files from your backup disk. If you have not maintained a backup disk, you must recreate the missing files.

You can use the CHKDSK command to check the condition of your disks. The verification process is non-destructive. See Chapter 10, "DR DOS commands" for instructions on how to use CHKDSK.

If these directions fail to correct an error condition, you may be faced with a hardware error: contact your dealer.

Corrupted files

Files can be corrupted by all kinds of unforeseen events. Power failures are probably the most common cause.

Replace the damaged file from your backup copy. If you do not have a backup copy, you will have to recreate the file.

If files are corrupted after you run a program that you yourself have written, debug the program carefully. Your program might be overwriting other files on the disk, or corrupting the directory.

When absolutely nothing happens

First, wait. Some programs take a long time to finish. Some errors, such as disk Input/Output errors, cause DR DOS to retry the operation many times. With experience, you will know when a command is taking too long.

Unresolvable errors

If you encounter an error condition you cannot correct, contact your dealer. Be prepared to provide the following information:

- ❑ The text of any error messages you received.
- ❑ The serial number of your operating system. (This number appears on your distribution diskette label.)
- ❑ What your computer configuration is, that is the number of drives, quantity of memory, and any additional equipment such as printers or modems.
- ❑ Enough information to reproduce the error. Try to recall which programs were running when the error happened. If possible, provide a diskette with a copy of the program.

Character sets for code pages supported by DR DOS

This appendix shows the five code pages supported in DR DOS. To enter a character, hold down the **Alt** key while you type the decimal value shown in the table.

Decimal Value/ Hex Value	Code Page 437	Code Page 850	Code Page 852	Code Page 857	Code Page 860	Code Page 863	Code Page 865	Code Page 866
0/00h								
1/01h	☺	☺	☺	☺	☺	☺	☺	☺
2/02h	☻	☻	☻	☻	☻	☻	☻	☻
3/03h	♥	♥	♥	♥	♥	♥	♥	♥
4/04h	♦	♦	♦	♦	♦	♦	♦	♦
5/05h	♣	♣	♣	♣	♣	♣	♣	♣
6/06h	♠	♠	♠	♠	♠	♠	♠	♠
7/07h	•	•	•	•	•	•	•	•
8/08h	◘	◘	◘	◘	◘	◘	◘	◘
9/09h	○	○	○	○	○	○	○	○
10/0Ah	◙	◙	◙	◙	◙	◙	◙	◙
11/0Bh	♂	♂	♂	♂	♂	♂	♂	♂
12/0Ch	♀	♀	♀	♀	♀	♀	♀	♀
13/0Dh	♪	♪	♪	♪	♪	♪	♪	♪
14/0Eh	♫	♫	♫	♫	♫	♫	♫	♫
15/0Fh	☼	☼	☼	☼	☼	☼	☼	☼
16/10h	►	►	►	►	►	►	►	►
17/11h	◄	◄	◄	◄	◄	◄	◄	◄
18/12h	↕	↕	↕	↕	↕	↕	↕	↕
19/13h	‼	‼	‼	‼	‼	‼	‼	‼

Decimal Value/ Hex Value	Code Page 437	Code Page 850	Code Page 852	Code Page 857	Code Page 860	Code Page 863	Code Page 865	Code Page 866
20/14h	¶	¶	¶	¶	¶	¶	¶	¶
21/15h	§	§	§	§	§	§	§	§
22/16h	—	—	—	—	—	—	—	—
23/17h	↨	↨	↨	↨	↨	↨	↨	↨
24/18h	↑	↑	↑	↑	↑	↑	↑	↑
25/19h	↓	↓	↓	↓	↓	↓	↓	↓
26/1Ah	→	→	→	→	→	→	→	→
27/1Bh	←	←	←	←	←	←	←	←
28/1Ch	∟	∟	∟	∟	∟	∟	∟	∟
29/1Dh	↔	↔	↔	↔	↔	↔	↔	↔
30/1Eh	▲	▲	▲	▲	▲	▲	▲	▲
31/1Fh	▼	▼	**	**	▼	▼	▼	▼
32/20h								
33/21h	!	!	!	!	!	!	!	!
34/22h	"	"	"	"	"	"	"	"
35/23h	#	#	#	#	#	#	#	#
36/24h	$	$	$	$	$	$	$	$
37/25h	%	%	%	%	%	%	%	%
38/26h	&	&	&	&	&	&	&	&
39/27h	'	'	'	'	'	'	'	'

Decimal Value/ Hex Value	Code Page 437	Code Page 850	Code Page 852	Code Page 857	Code Page 860	Code Page 863	Code Page 865	Code Page 866
40 /28h	(	(	(	(	(	(	(	(
41 /29h	)	)	)	)	)	)	)	)
42 /2Ah	*	*	*	*	*	*	*	*
43 /2Bh	+	+	+	+	+	+	+	+
44 /2Ch	,	,	,	,	,	,	,	,
45 /2Dh	-	-	-	-	-	-	-	-
46 /2Eh	.	.	.	.	.	.	.	.
47 /2Fh	/	/	/	/	/	/	/	/
48 /30h	0	0	0	0	0	0	0	0
49 /31h	1	1	1	1	1	1	1	1
50 /32h	2	2	2	2	2	2	2	2
51 /33h	3	3	3	3	3	3	3	3
52 /34h	4	4	4	4	4	4	4	4
53 /35h	5	5	5	5	5	5	5	5
54 /36h	6	6	6	6	6	6	6	6
55 /37h	7	7	7	7	7	7	7	7
56 /38h	8	8	8	8	8	8	8	8
57 /39h	9	9	9	9	9	9	9	9
58 /3Ah	:	:	:	:	:	:	:	:
59 /3Bh	;	;	;	;	;	;	;	;

Decimal Value/ Hex Value	Code Page 437	Code Page 850	Code Page 852	Code Page 857	Code Page 860	Code Page 863	Code Page 865	Code Page 866
60 /3Ch	<	<	<	<	<	<	<	<
61 /3Dh	=	=	=	=	=	=	=	=
62 /3Eh	>	>	>	>	>	>	>	>
63 /3Fh	?	?	?	?	?	?	?	?
64 /40h	@	@	@	@	@	@	@	@
65 /41h	A	A	A	A	A	A	A	A
66 /42h	B	B	B	B	B	B	B	B
67 /43h	C	C	C	C	C	C	C	C
68 /44h	D	D	D	D	D	D	D	D
69 /45h	E	E	E	E	E	E	E	E
70 /46h	F	F	F	F	F	F	F	F
71 /47h	G	G	G	G	G	G	G	G
72 /48h	H	H	H	H	H	H	H	H
73 /49h	I	I	I	I	I	I	I	I
74 /4Ah	J	J	J	J	J	J	J	J
75 /4Bh	K	K	K	K	K	K	K	K
76 /4Ch	L	L	L	L	L	L	L	L
77 /4Dh	M	M	M	M	M	M	M	M
78 /4Eh	N	N	N	N	N	N	N	N
79 /4Fh	O	O	O	O	O	O	O	O

Decimal Value/ Hex Value	Code Page 437	Code Page 850	Code Page 852	Code Page 857	Code Page 860	Code Page 863	Code Page 865	Code Page 866
80 /50h	P	P	P	P	P	P	P	P
81 /51h	Q	Q	Q	Q	Q	Q	Q	Q
82 /52h	R	R	R	R	R	R	R	R
83 /53h	S	S	S	S	S	S	S	S
84 /54h	T	T	T	T	T	T	T	T
85 /55h	U	U	U	U	U	U	U	U
86 /56h	V	V	V	V	V	V	V	V
87 /57h	W	W	W	W	W	W	W	W
88 /58h	X	X	X	X	X	X	X	X
89 /59h	Y	Y	Y	Y	Y	Y	Y	Y
90 /5Ah	Z	Z	Z	Z	Z	Z	Z	Z
91 /5Bh	[	[	[	[	[	[	[	[
92 /5Ch	\	\	\	\	\	\	\	\
93 /5Dh	]	]	]	]	]	]	]	]
94 /5Eh	^	^	^	^	^	^	^	^
95 /5Fh	_	_	_	_	_	_	_	_
96 /60h	′	′	′	′	′	′	′	′
97 /61h	a	a	a	a	a	a	a	a
98 /62h	b	b	b	b	b	b	b	b
99 /63h	c	c	c	c	c	c	c	c

Decimal Value/ Hex Value	Code Page 437	Code Page 850	Code Page 852	Code Page 857	Code Page 860	Code Page 863	Code Page 865	Code Page 866
100 /64h	d	d	d	d	d	d	d	d
101 /65h	e	e	e	e	e	e	e	e
102 /66h	f	f	f	f	f	f	f	f
103 /67h	g	g	g	g	g	g	g	g
104 /68h	h	h	h	h	h	h	h	h
105 /69h	i	i	i	i	i	i	i	i
106 /6Ah	j	j	j	j	j	j	j	j
107 /6Bh	k	k	k	k	k	k	k	k
108 /6Ch	l	l	l	l	l	l	l	l
109 /6Dh	m	m	m	m	m	m	m	m
110 /6Eh	n	n	n	n	n	n	n	n
111 /6Fh	o	o	o	o	o	o	o	o
112 /70h	p	p	p	p	p	p	p	p
113 /71h	q	q	q	q	q	q	q	q
114 /72h	r	r	r	r	r	r	r	r
115 /73h	s	s	s	s	s	s	s	s
116 /74h	t	t	t	t	t	t	t	t
117 /75h	u	u	u	u	u	u	u	u
118 /76h	v	v	v	v	v	v	v	v
119 /77h	w	w	w	w	w	w	w	w

Decimal Value/ Hex Value	Code Page 437	Code Page 850	Code Page 852	Code Page 857	Code Page 860	Code Page 863	Code Page 865	Code Page 866
120 /78h	x	x	x	x	x	x	x	x
121 /79h	y	y	y	y	y	y	y	y
122 /7Ah	z	z	z	z	z	z	z	z
123 /7Bh	{	{	{	{	{	{	{	{
124 /7Ch	|	|	|	|	|	|	|	|
125 /7Dh	}	}	}	}	}	}	}	}
126 /7Eh	~	~	~	~	~	~	~	~
127 /7Fh	⌂	⌂	⌂	⌂	⌂	⌂	⌂	⌂
128 /80h	Ç	Ç	Ç	Ç	Ç	Ç	Ç	А
129 /81h	ü	ü	ü	ü	ü	ü	ü	Б
130 /82h	é	é	é	é	é	é	é	В
131 /83h	â	â	â	â	â	â	â	Г
132 /84h	ä	ä	ä	ä	ã	Â	ä	Д
133 /85h	à	à	ů	à	à	à	à	Е
134 /86h	å	å	ć	å	Á	¶	å	ж
135 /87h	ç	ç	ç	ç	ç	ç	ç	З
136 /88h	ê	ê	ł	ê	ê	ê	ê	И
137 /89h	ë	ë	ë	ë	Ê	ë	ë	Й
138 /8Ah	è	è	Ő	è	è	è	è	К
139 /8Bh	ï	ï	ő	ï	Ì	ï	ï	Л

Decimal Value/ Hex Value	Code Page 437	Code Page 850	Code Page 852	Code Page 857	Code Page 860	Code Page 863	Code Page 865	Code Page 866
140 /8Ch	î	î	î	î	Ô	î	î	М
141 /8Dh	ì	ì	Ź	ı	ì	=	ì	Н
142 /8Eh	Ä	Ä	Ä	Ä	Ã	À	Ä	О
143 /8Fh	Å	Å	Ć	Å	Â	§	Å	П
144 /90h	É	É	É	É	É	É	É	Р
145 /91h	æ	æ	Ĺ	æ	À	È	æ	С
146 /92h	Æ	Æ	ĺ	Æ	È	Ê	Æ	Т
147 /93h	ô	ô	ô	ô	ô	ô	ô	У
148 /94h	ö	ö	ö	ö	õ	Ë	ö	Ф
149 /95h	ò	ò	Ľ	ò	ò	Ï	ò	Х
150 /96h	û	û	ľ	û	Ú	û	û	Ц
151 /97h	ù	ù	Ś	ù	ù	ù	ù	Ч
152 /98h	ÿ	ÿ	ś	İ	Ì	¤	ÿ	Ш
153 /99h	Ö	Ö	Ö	Ö	Õ	Ô	Ö	Щ
154 /9Ah	Ü	Ü	Ü	Ü	Ü	Ü	Ü	Ъ
155 /9Bh	¢	ø	Ť	ø	¢	¢	ø	Ы
156 /9Ch	£	£	ť	£	£	£	£	Ь
157 /9Dh	¥	Ø	Ł	Ø	Ù	Ù	Ø	Э
158 /9Eh	Pt	×	×	Ş	Pt	Û	Pt	Ю
159 /9Fh	*f*	*f*	č	§	Ó	*f*	*f*	Я

Decimal Value/ Hex Value	Code Page 437	Code Page 850	Code Page 852	Code Page 857	Code Page 860	Code Page 863	Code Page 865	Code Page 866
160 /A0h	á	á	á	á	á	¦	á	а
161 /A1h	í	í	í	í	í	´	í	б
162 /A2h	ó	ó	ó	ó	ó	ó	ó	В
163 /A3h	ú	ú	ú	ú	ú	ú	ú	Г
164 /A4h	ñ	ñ	Ą	ñ	ñ	¨	ñ	Д
165 /A5h	Ñ	Ñ	ą	Ñ	Ñ	¸	Ñ	е
166 /A6h	ª	ª	Ž	Ğ	ª	³	ª	ж
167 /A7h	º	º	ž	ğ	º	¯	º	З
168 /A8h	¿	¿	Ę	¿	¿	Î	¿	И
169 /A9h	⌐	®	ę	®	Ò	⌐	⌐	Й
170 /AAh	¬	¬		¬	¬	¬	¬	К
171 /ABh	½	½	ź	½	½	½	½	Л
172 /ACh	¼	¼	Č	¼	¼	¼	¼	М
173 /ADh	¡	¡	§	¡	¡	¾	¡	Н
174 /AEh	«	«	«	«	«	«	«	о
175 /AFh	»	»	»	»	»	»	¤	П
176 /B0h	░	░	░	░	░	░	░	░
177 /B1h	▒	▒	▒	▒	▒	▒	▒	▒
178 /B2h	▓	▓	▓	▓	▓	▓	▓	▓
179 /B3h	│	│	│	│	│	│	│	│

Decimal Value/ Hex Value	Code Page 437	Code Page 850	Code Page 852	Code Page 857	Code Page 860	Code Page 863	Code Page 865	Code Page 866
180 /B4h	┤	┤	┤	┤	┤	┤	┤	┤
181 /B5h	╡	Á	Á	Á	╡	╡	╡	╡
182 /B6h	╢	Â	Â	Â	╢	╢	╢	╢
183 /B7h	╖	À	Ě	À	╖	╖	╖	╖
184 /B8h	╕	©	Ş	©	╕	╕	╕	╕
185 /B9h	╣	╣	╣	╣	╣	╣	╣	╣
186 /BAh	║	║	║	║	║	║	║	║
187 /BBh	╗	╗	╗	╗	╗	╗	╗	╗
188 /BCh	╝	╝	╝	╝	╝	╝	╝	╝
189 /BDh	╜	¢	Ż	¢	╜	╜	╜	╜
190 /BEh	╛	¥	ż	¥	╛	╛	╛	╛
191 /BFh	┐	┐	┐	┐	┐	┐	┐	┐
192 /C0h	└	└	└	└	└	└	└	└
193 /C1h	┴	┴	┴	┴	┴	┴	┴	┴
194 /C2h	┬	┬	┬	┬	┬	┬	┬	┬
195 /C3h	├	├	├	├	├	├	├	├
196 /C4h	─	─	─	─	─	─	─	─
197 /C5h	┼	┼	┼	┼	┼	┼	┼	┼
198 /C6h	╞	ã	Ă	ã	╞	╞	╞	╞
199 /C7h	╟	Ã	ă	Ã	╟	╟	╟	╟

Decimal Value/ Hex Value	Code Page 437	Code Page 850	Code Page 852	Code Page 857	Code Page 860	Code Page 863	Code Page 865	Code Page 866
200 /C8h	╚	╚	╚	╚	╚	╚	╚	╚
201 /C9h	╔	╔	╔	╔	╔	╔	╔	╔
202 /CAh	╩	╩	╩	╩	╩	╩	╩	╩
203 /CBh	╦	╦	╦	╦	╦	╦	╦	╦
204 /CCh	╠	╠	╠	╠	╠	╠	╠	╠
205 /CDh	═	═	═	═	═	═	═	═
206 /CEh	╬	╬	╬	╬	╬	╬	╬	╬
207 /CFh	╧	¤	¤	¤	╧	╧	╧	╧
208 /D0h	╨	û	đ	º	╨	╨	╨	╨
209 /D1h	╤	D	Đ	ª	╤	╤	╤	╤
210 /D2h	╥	Ê	Ď	Ê	╥	╥	╥	╥
211 /D3h	╙	Ë	Ë	Ë	╙	╙	╙	╙
212 /D4h	╘	È	ď	È	╘	╘	╘	╘
213 /D5h	╒	ı	Ň		╒	╒	╒	╒
214 /D6h	╓	Í	Í	Í	╓	╓	╓	╓
215 /D7h	╫	Î	Î	Î	╫	╫	╫	╫
216 /D8h	╪	Ï	ě	Ï	╪	╪	╪	╪
217 /D9h	┘	┘	┘	┘	┘	┘	┘	┘
218 /DAh	┌	┌	┌	┌	┌	┌	┌	┌
219 /DBh	█	█	█	█	█	█	█	█

Decimal Value/ Hex Value	Code Page 437	Code Page 850	Code Page 852	Code Page 857	Code Page 860	Code Page 863	Code Page 865	Code Page 866
220 /DCh	▄	▄	▄	▄	▄	▄	▄	▄
221 /DDh	▌	¦	Ţ	¦	▌	▌	▌	▌
222 /DEh	▐	Ì	Ů	Ì	▐	▐	▐	▐
223 /DFh	▀	▀	▀	▀	▀	▀	▀	▀
224 /E0h	α	Ó	Ó	Ó	α	α	α	р
225 /E1h	ß	ß	ß	ß	ß	ß	ß	с
226 /E2h	Γ	Ô	Ô	Ô	Γ	Γ	Γ	т
227 /E3h	π	Ò	Ń	Ò	π	π	π	у
228 /E4h	Σ	õ	ń	õ	Σ	Σ	Σ	ф
229 /E5h	σ	Õ	ň	Õ	σ	σ	σ	х
230 /E6h	µ	µ	Š	µ	µ	µ	µ	ц
231 /E7h	τ	þ	š		τ	τ	τ	ч
232 /E8h	Φ	Þ	Ŕ	×	Φ	Φ	Φ	ш
233 /E9h	Θ	Ú	Ú	Ú	Θ	Θ	Θ	щ
234 /EAh	Ω	Û	ŕ	Û	Ω	Ω	Ω	ъ
235 /EBh	δ	Ù	Ű	Ù	δ	δ	δ	ы
236 /ECh	∞	y	ý	ì	∞	∞	∞	ь
237 /EDh	φ	Y	Ý	ÿ	φ	φ	φ	э
238 /EEh	ε	-	ţ	-	ε	ε	ε	ю
239 /EFh	∩	´	´	´	∩	∩	∩	я

Decimal Value/ Hex Value	Code Page 437	Code Page 850	Code Page 852	Code Page 857	Code Page 860	Code Page 863	Code Page 865	Code Page 866
240 /F0h	≡	-	-	-	≡	≡	≡	Ë
241 /F1h	±	±	˝	±	±	±	±	ë
242 /F2h	≥	‗	˛		≥	≥	≥	Є
243 /F3h	≤	¾	ˇ	¾	≤	≤	≤	є
244 /F4h	⌠	¶	˘	¶	⌠	⌠	⌠	Ї
245 /F5h	⌡	§	§	§	⌡	⌡	⌡	ї
246 /F6h	÷	÷	÷	÷	÷	÷	÷	Ў
247 /F7h	≈	¸	¸	¸	≈	≈	≈	ў
248 /F8h	°	°	°	°	°	°	°	°
249 /F9h	•	¨	¨	¨	•	•	•	•
250 /FAh	·	·	·	·	·	·	·	·
251 /FBh	√	^{1}	ű	^{1}	√	√	√	√
252 /FCh	^{n}	^{3}	Ř	^{3}	^{n}	^{n}	^{n}	№
253 /FDh	^{2}	^{2}	ř	^{2}	^{2}	^{2}	^{2}	¤
254 /FEh	■	■	■	■	■	■	■	■
255 /FFh								

Memory

C

THIS appendix describes the different types of memory in your computer.

C.1 What is memory?

Every computer must have memory to store programs and data, and to load and run programs. Memory used to hold programs and data while the computer is switched off is called *storage* memory. Memory that is used to load and run programs is called Random Access Memory (RAM). It can be thought of as the computer's workspace. The more complex a task you want to do, the more RAM memory your computer needs.

Information is stored in the computer as patterns of 1s and 0s, known as *bits*. A group of 8 bits is called a *byte*. Bytes are very important units of memory; characters such as "a", "3" or "?" can be represented in one byte of memory. One thousand and twenty-four bytes is equal to one *Kilobyte* (or Kbyte for short). Memory is also measured in *Megabytes* (Mbyte). One Megabyte is equal to 1024 Kilobytes. Modern computers contain a lot of memory, and it is therefore usually talked about in terms of Kbytes and Mbytes rather than bytes.

C.2 Types of memory

There are several types of RAM memory in personal computers:

❏ Lower.

❏ Conventional (or base).

❏ Upper.

❏ Extended.

❏ High.

❏ Expanded.

They are explained in this section.

For information on the types of memory contained in your computer, consult your computer's documentation, or your hardware dealer.

Lower and conventional memory

Conventional memory (sometimes called base memory) is the first 640Kbytes of memory in the computer, see figure C.1. It is divided into two sections:

- ❑ The area in the lowest part of base memory, where your operating system and installable device drivers are commonly loaded. This area is called lower memory.
- ❑ The area left over after your operating system is loaded. This is where your own applications and data are loaded.

Memory above the 640 Kbytes barrier is normally used for the video display adapter, and ROM BIOS in particular.

Under DOS operating systems, applications can only run in the first 640 Kbytes of memory. This limitation is sometimes called the *640 Kbytes barrier*. Other areas of memory can be used in a limited way, but only with certain types of hardware. Program designers usually prefer to write applications that can be run on a wide range of equipment. The more complicated the task to be accomplished, the larger the program will be. Your DR DOS operating system can be configured to take advantage of special memory areas in your computer, to ensure that your programs get as much conventional memory as possible with your hardware.

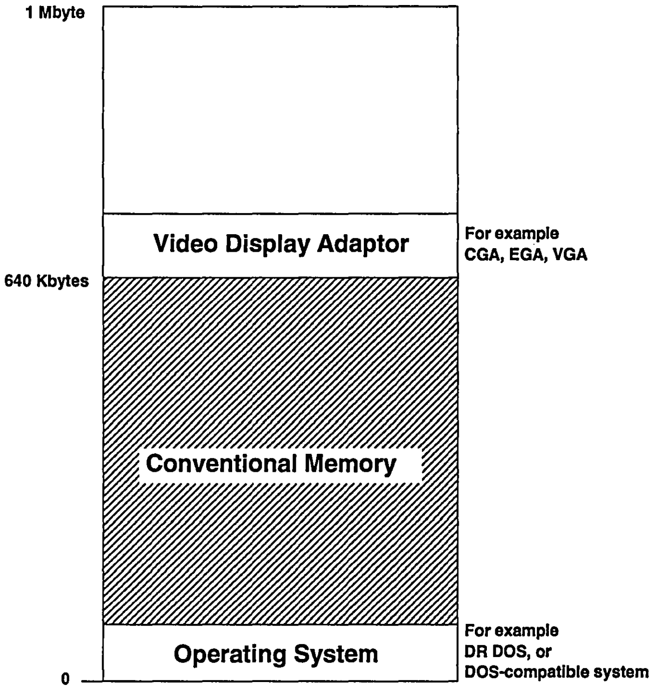


Figure C.1 8086 Memory Map

Upper memory

Any memory addressed between 640 Kbytes and 1 Mbyte is known as *upper* memory, see figure C.2. Upper memory is 384 Kbytes in size. Some sections of upper memory are reserved for use by various parts of your computer's hardware, like the video display adaptor, and also the ROMBIOS.

Your computer hardware may enable you to use one of the special DR DOS memory management drivers (EMM386.SYS and HIDOS.SYS). If so, you can take advantage of upper memory by loading installable device drivers, such as those required to run a mouse or network card, into portions of upper memory that are not being used, by for example, the video adapter, hard disk controller, or other similar hardware. These device drivers would normally be loaded in conventional memory, but when they are loaded in upper memory, a larger amount of conventional memory is available to load your applications. It is also possible to relocate the main operating system kernel into upper memory. Refer to Chapter 12, "Using MemoryMAX" for details of how you can take advantage of upper memory.

Extended memory

Intel 8086-based computers are limited to a maximum of 1 Mbyte of RAM. However Intel 80286, 386 and i486-based computers can address more than this, and the memory above 1 Mbyte is referred to as *extended* memory, see figure C.2.

Real, protected and virtual modes

Normally Intel 80286, 386 and i486-based computers run in *real* mode, which means they are simply running as fast 8086-based computers. For a computer to access extended memory it has to be put into a special mode of operation called *protected* mode. There is also a virtual 8086 mode, which combines the benefits of real and protected mode. The computer appears to an application to be running in real mode, but the features of protected mode, such as memory paging, are added or available (see also the section "Expanded memory" later in this chapter).

Certain device drivers and programs have options that allow them to gain access to extended memory. For example, VDISK.SYS has an option allowing you to set up a memory (or RAM) disk in extended memory (see Chapter 11, "Customizing your system").

High memory

High memory is the first 64 Kbytes of extended memory and begins at segment FFFF (see figure C.2). It can usually only be used by one program, or one piece of computer code at a time and is only available on Intel 80286, 386 and i486-based computers with more than 1 Mbyte of memory installed. It is sometimes referred to as the *High Memory Area* (or *HMA*). DR DOS can take advantage of high memory by moving the operating system kernel into it, so increasing the amount of memory available for your applications.

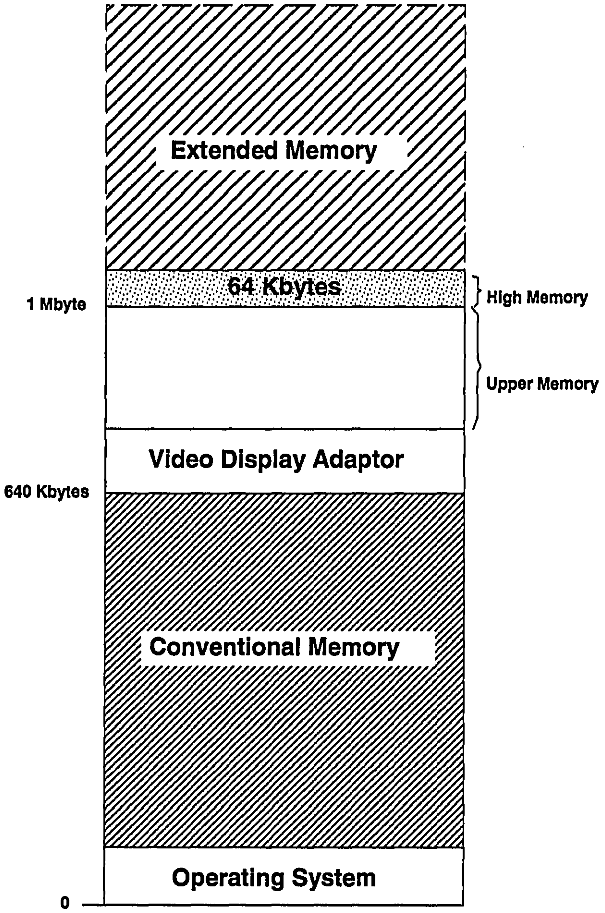


Figure C.2 Memory Map Showing Upper, High and Extended Memory

XMS (Extended Memory Specification)

A protocol called XMS (Extended Memory Specification) developed by Lotus, Intel and Microsoft, provides a standard interface to upper, high and extended memory. Up to 15 Mbytes of extended memory can be accessed on 80286-based computers, and up to 1000 Mbytes of extended memory on Intel 386, and i486-based computers.

Expanded memory

Another way of combatting the 640 Kbyte conventional memory barrier is to use a type of memory called *expanded* or *LIM* memory. The LIM EMS (Expanded Memory Specification) standard was jointly devised by Lotus, Intel, and Microsoft. It describes ways in which a program may access up to 32 Mbytes of memory outside conventional memory. Programs that use expanded memory include spreadsheets, databases, and other programs that need access to large amounts of data.

In the early days, the only way to use EMS memory was to fit a special memory expansion card inside your computer. Now, using virtual 8086 mode on Intel 386 and i486-based computers, these special memory cards can be emulated. Expanded memory is sometimes called *paged* memory, because it is divided up into 16 Kbyte memory pages.

Page frame

These memory pages are accessed through a block of four 16 Kbyte pages in upper memory, known as a page frame. Thus a 64 Kbyte area of upper memory is reserved for the page frame if access to expanded memory is required.

EMM driver

A device driver called the Expanded Memory Manager, or EMM, coordinates an application's access to expanded memory. The EMM driver arranges things so that the physical pages in the page frame always give access to the correct logical pages in expanded memory. So when data is read from or written to a physical page in the page frame, EMM ensures that the data is read from or written to the associated page of memory on the expanded memory card.

Expanded memory enables you to run programs that manipulate large amounts of data, and that would otherwise not run efficiently in conventional memory. However, it requires the programs to have been designed specifically to access it.

DR DOS has two device drivers that can provide expanded memory: EMM386.SYS and EMMXMA.SYS. EMM386.SYS is a special device driver which allows computers to emulate LIM expanded memory in extended memory (see Chapters 11, "Customizing your system" and 12, "Using MemoryMAX"). EMMXMA.SYS emulates LIM memory by using IBM XMA memory cards.

C.3 Checking on your computer's memory usage

To find out what memory you have on your computer, and how it is configured, use the MEM command. The MEM command gives a report on memory usage. It has various options you can use in any combination to produce different types of reports.

D

The SID debugging tool

D.1 About SID

SID is a powerful debugger. It allows you to test and debug programs interactively.

Most users will not need to use SID, but a description is included in this guide for completeness.

Starting SID

To start SID, type SID at the command prompt for example:

```
C>SID
```

You see the SID copyright message and prompt. SID is ready to accept your commands. You can now use the E command to load a program for a debugging session. SID tells you where in memory the program is loaded, for example:

```
#eprogram.exe

  Start      End
1000:0000 1000:2000

#
```

Alternatively, you can type SID at the command prompt followed by the name of the program you want to debug, for example:

```
C>SID program.exe
```

Quitting SID

To quit SID type **Q** or **Ctrl+C** at the # prompt. This returns control to the operating system.

SID command conventions

A command line in SID can be up to 64 characters in length, and must be terminated with a carriage return. SID does not process the command line until you press **Enter**.

The first character, or in some cases the first two characters, of each command line determines the command action. The command character can be followed by one or more arguments. Arguments must be separated from each other by commas or spaces. No spaces are allowed between the command character and the first argument.

Note that if the first character of a SID command line is a semicolon (;), the entire line is treated as a comment and ignored.

Specifying a 20-bit address

Most SID commands require one or more addresses as operands. Because the 8086 processor can address up to 1 Mbyte of memory, addresses must be 20-bit values. Enter a 20-bit address as follows:

```
ssss:oooo
```

where *ssss* represents an optional 16-bit segment number and *oooo* is a 16-bit offset. SID combines these values to produce a 20-bit effective address:

```
ssss0
+oooo
---
eeeee
```

The segment value, *ssss*, is optional. If you omit the segment value, a default value appropriate to the command being executed, is used.

Hexadecimal numbers

SID normally accepts and displays values in hexadecimal. The valid hexadecimal digits are the decimal digits 0 through 9 and the hexadecimal digits A, B, C, D, E, and F, corresponding to the decimal values 10 through 15.

Decimal numbers

To enter decimal numbers, precede the number with the # symbol. The number that follows must comprise one or more decimal digits (0 through 9).

Character values

SID accepts ASCII characters, enclosed in single quotation marks, as literal values in expressions. Characters remain as typed within the paired quotation marks that is, no case translation occurs.

Operators in expressions

Numbers and strings can be combined into expressions using + and - operators. The entire sequence must be written without embedded blanks.

D.2 SID commands

SID commands give you control of program execution and allow you to display and modify system memory and the CPU state. This section lists the SID commands in alphabetical order. The commands are summarized in Table D-1.

Table D-1 Summary of SID Commands

Command	Explanation
?	Displays available commands.
??	Displays detailed command list.
A	Assembles 8086 mnemonics directly into memory.
B	Block compares memory and displays discrepancies found.
D	Displays memory in byte or word format.
E	Loads a file into memory so that the next G, T or U command can begin program execution.
F	Fills an area of memory with a byte or word constant.
G	Transfers control to the program being tested and optionally sets one or two breakpoints.
H	Provides hexadecimal arithmetic functions.
I*string*	Sets up command line arguments.
L	Lists the contents of memory in assembly language.
M	Copies a block of data values from one area of memory to another.
P	Sets, clears and lists current breakpoints.
Q	Exits from the SID program.
QI, QO	Allows access to any of the 65,536 input/output ports. QI reads data from a port and QO writes data to a port.
QR, QW	Enables you to read and write data from absolute locations on a given disk drive.

Table D-1 Summary of SID Commands (continued)

Command	Explanation
R	Reads a file into a contiguous block of memory.
S	Changes the contents of bytes or words of memory.
SR	Searches for a given ASCII "string" or pattern of bytes and lists the addresses where the pattern occurs.
T	Executes one or more instructions.
U	This is the same as T, but without register display.
V	Displays the values of the last file loaded using the E or R commands.
W	Writes the contents of a contiguous block of memory to disk file.
X	Examines or modifies the CPU registers and flags.
Z	Displays registers of an attached 8087 maths coprocessor.

? & ?? (Help)

The ? command prints a list of available SID commands. The format is as follows:

```
?
```

The ?? command prints a detailed command list that, in addition to the SID commands, includes the available command options. The format is as follows:

```
??
```

A (ASSEMBLE)

The A command assembles 8086 mnemonics directly into memory. It takes the form:

```
As
```

where *s* is the 20-bit address at which assembly starts.

SID responds to the A command by displaying the address of the memory location where assembly begins. At this point, you enter assembly language statements. When you enter each statement, SID converts it to binary, places the value(s) in memory, and displays the address of the next available memory location. This process continues until you enter a blank line or a line containing only a period. SID responds to invalid statements by displaying a question mark, ?, and redisplaying the current assembly address.

Note

You can use an expression wherever a numeric value is valid in an assembly language statement. Under the A command, references to registers refer to the names of the registers, not to the contents of the registers as in other SID commands. Under the A command, there is no way to reference the contents of a register in an expression.

Examples

Assemble at 4000:213.

```
#A4000:213
```

Set AX register to decimal 128.

```
4000:0213 MOV ax,#128
```

Push AX register on stack.

```
4000:0216 PUSH ax
```

B (BLOCK COMPARE)

The B command compares two blocks of memory and displays any discrepancies at the screen. The B command takes the form:

```
Bs1,f1,s2
```

where *s1* is the 20-bit address of the start of the first block; *fl* is the offset of the final byte of the first block, and *s2* is the 20-bit address of the start of the second block.

If the segment is not specified in *s2,* the same value is used that was used for *s1.*

Any differences in the two blocks are displayed on the screen in the form:

```
    a1 b1 a2 b2
```

where *a1* and the *a2* are the addresses in the blocks; *b1* and *b2* are the values at the indicated addresses. If no differences are displayed, the blocks are identical.

Example

Compare 200H bytes starting at 40:0 with the block starting at 60:0.

```
#B40:0,1ff,60:0
```

D (DISPLAY)

The D command displays the contents of memory as 8-bit or 16-bit hexadecimal values, and in ASCII. The command takes the following forms, where *s* is the 20-bit address where the display starts, and *f* is the 16-bit offset within the segment specified in *s* where the display finishes.

(a) `D` Displays memory from the current display address for 12 display lines.

(b) `Ds` Response to (b) is similar to form (a) but the display address is first set to be the 20-bit address *s*.

(c) `Ds,f` Displays the memory block between locations *s* and *f*.

(d) `DW`

(e) `DWs`

(f) `DWs,f` Analogous to the first three forms, except that the contents of memory are displayed as 16-bit values, rather than 8-bit values, as shown below:

```
ssss:oooo wwww wwww . . . wwww cccc . . . cc
```

Memory is displayed on one or more display lines. Each display line shows the values of up to 16 memory locations. For the first three forms, the display line appears as follows:

```
ssss:oooo bb bb . . . bb cc . . . c
```

Where *ssss* is the segment being displayed, and *oooo* is the offset within segment *ssss*. The *bb* 's represent the contents of the memory locations in hexadecimal, and the *c* 's represent the contents of memory in ASCII. A period represents any non-graphic ASCII character.

During a long display, you can abort the D command by typing **Ctrl+Break** at the console.

Example

Display memory bytes from offset F00H through F23H in the current data segment.

```
#Df00,f23
```

E (LOAD PROGRAM)

The E command loads a file into memory so that the next G, T, or U command can begin program execution. The E command takes the forms:

(a) `E filename`
(b) `E`

Form (a) loads the file by the name *filename*. The file is assumed to be either a **.com** or **.exe** file. If no file type is specified, **.exe** is assumed. To load a **.com** file, you must specify the file type with the filename. The contents of the CS, DS, ES, SS, SP, and IP registers are altered according to the type of file loaded. When the load is complete, SID displays the start and end address of the memory block where the file was loaded. Use the V command to redisplay this information later.

Form (b) releases all memory being used by SID for programs.

When loading a program file with the E command, SID releases any blocks of memory used by previous E or R commands, or by programs executed under SID. Only one file can be loaded for execution at a time.

SID issues an error message if a file does not exist or cannot be successfully loaded in the available memory.

Example

Load file **test.exe**:

```
#Etest
```

F (FILL)

The F command fills an area of memory with a byte or word constant. The following are possible forms:

(a) `Fs,f,b`
(b) `FWs,f,w`

The *s* is a 20-bit starting address of the block to be filled, and *f* is a 16-bit offset of the final byte of the block within the segment specified in *s*.

In response to form (a), SID stores the 8-bit value *b* in locations *s* through *f*. In form (b), the 16-bit value *w* is stored in locations *s* through *f* in standard form, low 8-bits first followed by high 8-bits.

If *s* is greater than *f* or the value *b* is greater than 255, SID responds with a question mark. SID issues an error message if the value stored in memory cannot be read back successfully, indicating faulty or non-existent RAM at the location indicated.

Example

Fill memory from 100H through 13FH with 0.

```
#F100,13f,0
```

G (GO)

The G command transfers control to the program being tested, and optionally sets one or two breakpoints. The following are possible forms:

(a) `G`
(b) `G,b1`
(c) `G,b1,b2`
(d) `Gs`
(e) `Gs,b1`
(f) `Gs,b1,b2`
(g) `-G` (with forms a through f)

where *s* is a 20-bit address where program execution is to start, and *b1* and *b2* are 20-bit addresses of breakpoints. If no segment value is supplied for any of the addresses, the value defaults to the contents of the CS register.

In forms (a), (b), and (c), no starting address is specified, so SID derives the 20-bit address from the CS and IP registers. Form (a) transfers control to your program without setting any breakpoints. Forms (b) and (c) set one and two breakpoints respectively before passing control to your program. Forms (d), (e), and (f) are analogous to (a), (b), and (c), except that the CS and IP registers are first set to *s*. The forms in (g) are analogous to forms (a) through (f), except that intermediate pass point displays are suppressed.

Once control has been transferred to the program under test, it executes in real time until it encounters a breakpoint. SID then regains control, clears the breakpoints set by the G command, and indicates the address at which execution of the program under test was interrupted, as shown:

```
    *ssss:oooo
```

The *ssss* corresponds to the CS, *oooo* corresponds to the IP where the break occurred. When a breakpoint returns control to SID, the instruction at the breakpoint address has not yet been executed.

Example

Begin program execution at address given by CS and IP registers with no breakpoints set.

#G

H (HEXADECIMAL MATH)

The H command provides several useful arithmetic functions. The possible forms are as follows:

(a) `Ha,b`
(b) `Ha`

Form (a) computes the sum *ssss*, difference *dddd*, product *pppppppp*, and quotient *qqqq* with the remainder *rrrr* of two 16-bit values. The results are displayed in hexadecimal as below:

```
+ ssss  - dddd  * pppppppp  / qqqq  (rrrr)
```

Underflow and overflow are ignored in addition and subtraction.

Form (b) displays the value of the expression *a* in hexadecimal, decimal, ASCII (if the value has a graphic ASCII equivalent), forms as shown:

```
hhhh  #ddddd  'c'
```

You can abort the display by pressing any key at the console.

Example

Show sum, difference, product, and quotient of 5C28H and 80H.

```
#H5c28,80
```

I (INPUT COMMAND TAIL)

The I command prepares a file control block and command tail buffer in SID's program segment prefix, and copies this information into the program segment prefix of the last file loaded with the E command. The command takes the following form:

```
Icommand_tail
```

The *command_tail* is a character string that usually contains one or more filenames. The first filename is parsed into the default file control block at 005CH. The optional second filename is parsed into the second part of the default file control block beginning at 006CH. The characters in *command_tail* are also copied into the default command buffer at 0080H. The length of *command_tail* is stored at 0080H, followed by the character string terminated with a binary zero.

If a file has been loaded with the E command, SID copies the file control block and command buffer from the program segment prefix of SID to the program segment prefix of the program loaded. The location of the program segment prefix of a program loaded with the E command is the value displayed for DS on completion of the program load.

Examples

Set up a file control block at 5CH for the file **file1.exe** and put the string "file1.exe" in the buffer at 80H (in the program segment prefix of the last file loaded with an E command).

```
#Ifile1.exe
```

Set up file control blocks at 5CH and 6CH for the files **a:file1** and **b:file2**, and copy the string following the "i" into the buffer at 80H.

```
#Ia:file1 b:file2 c:file3 $px
```

L (LIST)

The L command lists the contents of memory in assembly language. The following are possible forms:

(a) `L`
(b) `Ls`
(c) `Ls,f`

The *s* is a 20-bit address where the list starts, and *f* is a 16- bit offset within the segment specified in *s* where the list finishes.

Each disassembled instruction is in the form:

```
ssss:oooo prefixes opcode operands
```

prefixes are segment override, lock and repeat prefixes; *opcode* is the mnemonic for the instruction; the *operands* field contains 0, 1, or 2 operands, as required by the instruction.

Form (a) lists twelve disassembled instructions from the current list address. Form (b) sets the list address to *s* and then lists twelve instructions. Form (c) lists disassembled code from *s* through *f*.

The list address is always set to the next unlisted location in preparation for the next L command. When SID regains control from a program being tested (see G, T, and U commands), the list address is set to the current value of the CS and IP registers.

You can abort long displays by typing **Ctrl+Break** during the list process or by entering **Ctrl+S** to halt the display temporarily. **Ctrl+Q** will restart the display.

If the memory location being disassembled is not a valid 8086 instruction, SID displays it in the form:

```
    ??=  nn
```

The *nn* is the hexadecimal value of the contents of the memory location.

Example

Disassemble instructions from 243CH through 244EH.

```
#L243c,244e
```

M (MOVE)

The M command moves a block of data values from one area of memory to another. The M command takes the form:

```
Ms,f,d
```

The *s* is the 20-bit starting address of the block to be moved, *f* is the offset of the final byte to be moved within the segment described by *s*, and *d* is the 20-bit address of the first byte of the area to receive the data. If the segment is not specified in *d*, the same value is used as was used for *s*.

Note

If *d* is between *s* and *f*, part of the block being moved is overwritten before it is moved, because data is transferred starting from location *s*.

Example

Move 10 bytes from 20:2400 to 30:100.

```
#M20:2400,+9,30:100
```

P (PASS POINT)

The P command sets, clears, and displays pass points. The following are the possible forms:

(a) `Pa,n`
(b) `Pa`
(c) `-Pa`
(d) `-P`
(e) `P`

A pass point is a permanent breakpoint that remains in effect until it is explicitly removed, as opposed to breakpoints set with the G command, which must be reentered with each G command. Pass points have associated pass counts ranging from 1 to 0FFFFH. The pass count indicates how many times the instruction at the pass point executes before the control returns to the console. Up to sixteen pass points can be set at a time.

An important distinction between breakpoints and pass points is that when execution stops at a breakpoint, the instruction at the breakpoint has not yet been executed. When execution stops at a pass point whose pass count has reached 1, the instruction at the pass point has been executed. This makes it simple to proceed from a pass point with a G command without immediately encountering the same pass point.

Forms (a) and (b) are used to set pass points. Form (a) sets a pass point at address *a* with a pass count of *n*, where *a* is the 20-bit address of the pass point, and *n* is the pass count, in the range 1 to 0FFFFH. If a pass point is already active at *a*, the pass count is simply changed to *n*. SID responds with a question mark if there are already 16 active pass points.

Form (b) sets a pass point at *a* with a pass count of 1. If a pass point is already active at *a*, the pass count is simply changed to 1. SID responds with a question mark if there are already 16 active pass points. Forms (c) and (d) are used to clear pass points. Form (c) clears the pass point at location *a*. SID responds with a question mark if there is no pass point set at *a*. Form (d) clears all the pass points. Form (e) displays all the active pass points in the form:

```
nnnn  ssss:oooo
```

where *nnnn* is the current pass count for the pass point, *ssss:oooo* is the segment and offset of the pass point location.

When a pass point is encountered, SID displays the pass point information in the form:

```
nnnn  PASS  ssss:oooo
```

Next, SID displays the CPU state before the instruction at the pass point is executed. SID then executes the instruction at the pass point. If the pass count is greater than 1, SID decrements the pass count and transfers control back to the program under test. When the pass count reaches 1, SID displays the break address (that of the next instruction to be executed) in the following form:

```
*ssss:oooo
```

Once the pass count reaches 1, it remains at 1 until the pass point is cleared or the pass count is changed with another P command.

Use pass points with the G, T, and U commands. When you use the G or U command, you can suppress the intermediate pass point display with the -G or -U forms (see the G and U commands). In this case, only the final pass points (when the pass count = 1) are displayed. You can interrupt the G or U command before its normal termination by pressing **Ctrl+Break** at the console. SID aborts the G or U command when the next pass point is encountered and prompts for the next command. You can also use pass points with breakpoints set with the G command.

If a pass point and a breakpoint are set at the same address, the breakpoint is encountered first. Otherwise, pass points behave in the usual manner, decrementing the pass count until it reaches 1, and then returning control to SID.

Normally, the segment registers are not displayed at pass points. You can use the S/-S command to enable/disable the segment register display (see the S command).

Examples

Display active pass points.

```
#P
```

Set pass point at address 1000:1234

```
#P1000:1234
```

Q (QUIT)

The Q command exits from SID.

QI, QO, QR, QW (QUERY I/O)

The QI and QO commands allow access to any of the 65,536 input/output ports. The QI command reads data from a port; the QO command writes data to a port. The QI command takes the following forms:

```
QIn
```

```
QIWn
```

where *n* is the 16-bit port number. In the first case, SID displays the 8-bit value read from port *n*. In the second case, SID displays a 16-bit value from port *n*.

The QO command takes the following forms:

```
QOn,v
```

```
QOWn,v
```

where *n* is the 16-bit port number, and *v* is the value to output. In the first case, the 8-bit value *v* is written to port *n*. If *v* is greater than 255, SID responds with a question mark. In the second case, the 16-bit value *v* is written to port *n*.

Examples

Display the 16-bit value of input port 20H.

```
#QIW20
```

Display the 8-bit value of input port 1024.

```
#QI#1024
```

Set the 16-bit output port number 20H to 0FF7EH.

```
#QOW20,FF7E
```

Set the 8-bit output port number 1025 to 2.

```
#QO#1025,2
```

The QR and QW commands enable you to read and write data from absolute locations on a given disk drive. The two commands have identical parameters:

```
QRs,drive,sector,count
```

```
QWs,drive,sector,count
```

where *s* is the segment:offset in memory into/from which data is to be read/written, *drive* is the drive number to use, for example 0=A:, 1=B:, *sector* is the starting sector number, and *count* is the number of sectors to process.

Examples

To read 2 sectors from logical sector 15 of drive A: into memory starting at 9000:0000:

```
QR9000:0,0,15,2
```

To write the same two sectors back out to drive B:

```
QW0,1,15,2
```

Notice that the full segment part of the address need not be given if it has not changed since the last command.

This command must be used with the upmost care, as vital information such as directories or file allocation tables can be damaged by incorrect use.

Note

A 32-bit sector number can be used which takes the form *nnnn:nnnn*, that is 0003:ffff for sector 3ffff.

R (READ)

The R command reads a file into a contiguous block of memory. The R command takes the forms:

(a) `R filename`
(b) `R filename,s`

where *filename* is the name and type of the file to be read, and *s* is the location to which the file is read. Form (a) lets SID determine the memory location into which the file is read. Form (b) causes SID to read the file into the memory segment beginning at *s*. This address can have the standard form (*ssss:oooo*) as in the following example:

```
#Rcommand.com,1000:0
```

The low-order four bits of *s* are assumed to be zero, so SID reads files on a paragraph boundary.

SID reads the file into memory and displays the start and end addresses of the block of memory the file occupies. A V command can redisplay this information later. The default display pointer (for further D commands) is set to the start of the block occupied by the file.

The R command does not free memory used by another R or E command. Thus, many files can be read into memory without overlapping.

SID issues an error message if the file does not exist or there is not enough memory to load the file.

Examples

Read file **sid.com** into memory.

```
#Rsid.com
```

Read file **test** into memory.

```
#Rtest
```

Read file **test** into memory starting at location 1000:0.

```
#Rtest,1000:0
```

S (SET)

The S command changes the contents of bytes or words of memory. The following are possible forms:

(a) `Ss`
(b) `SWs`
(c) `S`
(d) `-S`

where *s* is the 20-bit address where the change occurs.

SID displays the memory address and its current contents on the following line. In response to form (a), the display is

```
    ssss:oooo bb
```

and in response to form (b), the display is

```
    ssss:oooo wwww
```

where *bb* and *wwww* are the contents of memory in byte and word formats, respectively.

In response to one of these displays, you can choose to alter the memory location or to leave it unchanged. If you enter a valid expression, the contents of the byte (or word) in memory is replaced with the value of the expression. If you do not enter a value, the contents of memory are unaffected and the contents of the next address are displayed. In either case, SID continues to display successive memory addresses and values until you enter a period on a line by itself, or until SID detects an invalid expression.

In response to form (a), you can enter a string of ASCII characters, beginning with a quotation mark and ending with a carriage return. The characters between the quotation mark and the carriage return are placed in memory starting at the address displayed. No case conversion takes place. The next address displayed is the address following the character string. SID issues an error message if the value stored in memory cannot be read back successfully, indicating faulty or non-existent RAM at the location indicated.

The forms (c) and (d) control the display of the segment registers when the CPU state is displayed with the trace command and at pass points. Form (c) turns on the segment register display, and form (d) turns it off. It is often convenient to turn off the segment register display while debugging to allow the CPU state display to fit on one line.

Examples

Set byte to 0.

```
1000:1234  55  0
```

Set 3 bytes to 'a', 'b', 'c'.

```
1000:1235  55  ''abc
```

Set byte to decimal 75.

```
1000:1238  55  #75
```

Terminate the set command.

```
1000:1239  55  .
```

Enable segment register display in CPU state display.

```
#s
```

Disable segment register display in CPU state display.

```
#-s
```

SR (SEARCH)

The SR command searches a block of memory for a given pattern of numeric or ASCII values, and lists the addresses where the pattern occurs. The SR command takes the form:

```
SRs,f,pattern
```

where *s* is the 20-bit starting address of the block to be searched, *f* is the offset of the final address of the block, and *pattern* is a list of one or more hexadecimal values and/or ASCII strings. ASCII strings are enclosed in double or single quotation marks and can be any length.

For each occurrence of *pattern*, SID displays the 20-bit address of the first byte of the pattern, in the form:

```
    ssss:oooo
```

If no addresses are listed, the pattern was not found.

Example

Search the memory block between 0:0 and 0:0FFFFH for the 3-character pattern that starts with "x", ends with "y", and has a decimal 27 in the middle.

```
#SR0:0,ffff,"x",#27,"y"
```

T (TRACE)

The T command traces program execution for 1 to 0FFFFH program steps, displaying the CPU state before each step. The T command takes the following forms:

(a) `T`
(b) `Tn`
(c) `TW`
(d) `TWn`

The *n* is the number of program steps to execute before returning control to the console. If *n* is omitted, a single program step is executed. A program step is generally a single instruction, with the following exceptions:

- ❑ If a DR DOS interrupt instruction is traced, the entire function is treated as one program step, and executes in real time. This is because SID itself makes DR DOS calls, and DR DOS is not reentrant.
- ❑ The 8086 has a feature that disables interrupts (including the Trace Interrupt) for one instruction after a MOV or POP loads a segment register. For this reason, if the traced instruction is a MOV or POP whose destination is a segment register, the CPU executes the next instruction immediately. This prevents an interrupt occurring when the stack is undefined, for example between the two instructions in a sequence such as:

  ```
  MOV SS, STACKSEGMENT
  MOV SP, STACKOFFSET
  ```

  A sequence of such MOV or POP instructions plus one instruction after the sequence is considered a one program step.
- ❑ If any of the TW forms are used and the traced instruction is a CALL, CALLF or INT, the entire called subroutine or interrupt handler (and any subroutines called therein) is treated as a one program step and executes in real time.

Before each program step is executed, SID displays the CPU state, the disassembled instruction to be executed, and the contents of the memory location(s) referenced by the instruction, if appropriate. (See the X command for a detailed description of the CPU state display.)

The segment registers are not normally displayed with the T command, which allows the entire CPU state to be displayed on one line. To enable the segment register display, use the S command (see the S command). With the segment register display enabled, the display of the CPU state is identical to that of the X command.

In all the forms, control transfers to the program under test at the address indicated by the CS and IP registers. If *n* is not specified, as in form (a), one program step is executed. Otherwise, SID executes *n* program steps and displays the CPU state before each step, as in form (b). You can abort a long trace before *n* steps have been executed by typing **Ctrl+Break** at the console.

When *n* steps have been executed, SID displays the address of the next instruction to be executed in the form:

```
*ssss:oooo
```

Forms (c) and (d) are analogous to forms (a) and (b), except in the way subroutine calls are treated. In the TW forms, the entire subroutine called from the program level being traced is treated as a single program step, and executes in real time. This allows tracing at a high level of the program, ignoring subroutines that have already been debugged, or for other reasons are not currently of interest.

When a single instruction is being traced, interrupts are disabled for the duration of the instruction. This prevents SID from tracing through interrupt handlers when debugging on systems in which interrupts occur frequently.

After a T command, the list address used in the L command is set to the address of the next instruction to be executed, and the default segment values are set to the CS and DS register values.

Examples

Trace one program step.

```
#T
```

Trace 65535 steps.

```
#Tffff
```

U (UNTRACE)

The U command is similar to the T command except that the CPU state is displayed only before the first instruction is executed, rather than before every step. The following are possible forms:

(a) `U`
(b) `Un`
(c) `UW`
(d) `UWn`
(e) `-U` (with forms a through d)

where *n* is the number of instructions to execute before returning control to the console. You can abort the U command before *n* steps have been executed by typing **Ctrl+Break** at the console.

Form (e) differs from the analogous T command in that SID disables the display of intermediate pass points (while the pass count is greater than 1). In this case, only when the pass count reaches 1 is the pass information displayed (see the P command).

Examples

Trace without display 200H steps.

```
#U200
```

Trace without display 200H steps, suppressing the intermediate pass point display.

```
#-U200
```

V (VALUE)

The V command displays the start and end addresses of the block of memory where the last file was loaded with the E or R commands. The V command takes the form:

```
V
```

SID responds to the V command with a question mark if neither the R nor E commands have been used.

W (WRITE)

The W command writes the contents of a contiguous block of memory to disk. The following are the possible forms:

(a) `W filename`
(b) `W filename,s,f`

where *filename* is the filename and file type of the disk file to receive the data, and *s* and *f* are the 20-bit first and last addresses of the block to be written. If the segment is not specified in *f*, SID uses the same value that was used for *s*.

With form (a), SID assumes the *s* and *f* values from the last file read with an R command. If no file was read with an R command, SID responds with a question mark. This form is useful for writing out files after patches have been installed, assuming the overall length of the file is unchanged.

With form (b), where *s* and *f* are specified as 20-bit addresses, the low four bits of *s* are assumed to be 0. Thus the block being written must always start on a paragraph boundary.

If a file with the name specified in the W command already exists, SID deletes it before writing a new file.

Examples

Write to the file **test.exe** the contents of the memory block read into by the most recent R command.

```
#Wtest.exe
```

Write the contents of the memory block 40:0 through 40:3FFF to the file **test.exe** on drive B:.

```
#Wb:test.exe,40:0,3fff
```

X (EXAMINE CPU STATE)

The X command allows you to examine and alter the CPU state of the program under test. The X command takes the forms:

(a) `X`
(b) `Xr`
(c) `Xf`

where *r* is the name of a CPU register and *f* is the abbreviation of one CPU flag. Form (a) displays the CPU state in the format:

```
                AX    BX    CX        . . .      SS     ES   IP
---------  xxxx  xxxx  xxxx          . . xxxx xxxx xxxx
instruction                 memory_value
```

The nine hyphens at the beginning of the line indicate the state of the nine CPU flags. Each position can be either a hyphen, indicating that the corresponding flag is not set (0), or a 1-character abbreviation of the flag name, indicating that the flag is set (1). Table D-2 shows the abbreviations of the flag names.

Table D-2 Flag Name Abbreviations

Character	Flag name
O	Overflow
D	Direction
I	Interrupt Enable
T	Trap
S	Sign
Z	Zero
A	Auxiliary Carry
P	Parity
C	Carry

The *instruction* is the disassembled instruction at the next location to be executed, which is indicated by the CS and IP registers.

If an instruction references memory, the contents of the referenced location(s) are displayed in the *memory_value* field, preceded by an equal sign. Either a byte, word, or double word value is shown, depending on the instruction. In addition to displaying the machine state, the first form changes the values of the default segments back to the CS and DS register values, and the default offset for the L command to the IP register value.

Form (b) allows you to alter the registers in the CPU state of the program being tested. The *r* following the X is the name of one 16-bit CPU register. SID responds by displaying the name of the register followed by its current value. If you type a carriage return, the value of the register is not changed. If you type a valid expression, the contents of the register are changed to the value of the expression. In either case, the next register is then displayed. This process continues until you enter a period or an invalid expression, or the last register is displayed.

Form (c) allows you to alter a flag in the CPU state of the program being tested. SID responds by displaying the name of the flag followed by its current state. If you type a carriage return, the state of the flag is not changed. If you type a valid value, the state of the flag is changed to that value. Only one flag can be examined or altered with each X*f* command. Set or reset flags by entering a value of 1 or 0.

Examples

Change registers starting with BP.

```
#Xbp
```

Change BP to hex 2B64.

```
BP=1000 2b64
```

Change SI to decimal 12345.

```
SI=2000  #12345
```

Z (Display the 8087 coprocessor registers)

This command displays the registers of an attached 8087 maths coprocessor if one is present.

D.3 Default segment values

SID has an internal mechanism that keeps track of the current segment value, making segment specification an optional part of a SID command. SID divides the command set into two types of command according to the segment a command defaults to if you do not specify a segment value in the command line.

The first type of command, pertaining to the code segment, includes the A (Assemble), L (List), P (Pass Points) and W (Write) commands. These commands use the internal type-1 segment value if no segment value is specified in the command.

When invoked, SID sets the type-1 segment value to the first available free segment.

❑ When an E command loads a file, SID sets the type-1 segment value to the value of the CS register.

❑ When an R command reads a file, SID sets the type-1 segment value to the base segment where the file was read.

❑ When an X command changes the value of the CS register, SID changes the type-1 segment value to the new value of the CS register.

❑ When SID regains control from a user program after a G, T, or U command, it sets the type-1 segment value to the value of the CS register.

❑ When an A or L command explicitly specifies a segment value, SID sets the type-1 segment value to the segment value specified.

The second type of command, pertaining to the data segment, includes the B (Block Compare), D (Display), F (Fill), M (Move), S (Set), and SR (Search) commands. These commands use the internal type-2 segment value if no segment value is specified in the command.

When invoked, SID sets the type-2 segment value to the first available free segment.

- ❑ When an E command loads a file, SID sets the type-2 segment value to the value of the DS register.
- ❑ When an R command reads a file, SID sets the type-2 segment value to the base segment where the file was read.
- ❑ When an X command changes the value of the DS register, SID changes the type-2 segment value to the new value of the DS register.
- ❑ When SID regains control from a user program after a G, T, or U command, it sets the type-2 segment value to the value of the DS register.
- ❑ When a B, D, F, M, S, or SR command explicitly specifies a segment value, SID sets the type-2 segment value to the segment value specified.
- ❑ When evaluating programs that have identical values in the CS and DS registers, all SID commands default to the same segment value unless explicitly overridden.

Table D-3 summarizes SID's default segment values.

Note

The G (Go) command does not fall into either group because it defaults to the CS register.

Table D-3 Summary of SID Default Segment Values

Command	Type-1	Type-2
A	*	
B		*
D		*
E	@	@
F		*
G	@	@
H		
I		
L	*	
M		*
P	%	
QI		
QO		
R	*	@
S		*
SR		*
T	@	@
U	@	@
V		
W	*	
X	@	@

Key to Table D-3

*	Use this segment default if none specified; change default if specified explicitly.
@	Change this segment default.
%	Use this segment default if none specified.

D.4 Assembly language syntax (A and L commands)

In general, the syntax of the assembly language statements in the A and L commands is standard 8086 assembly language. Several minor exceptions appear here.

❑ Up to three prefixes (LOCK, repeat, segment override) can appear in one statement, but they all must precede the opcode of the statement. Alternatively, a prefix can be entered on a line by itself.

❑ The distinction between byte and word string instructions is made as follows:

byte	word
LODSB	LODSW
STOSB	STOSW
SCASB	SCASW
MOVSB	MOVSW
CMPSB	CMPSW

❑ The mnemonics for near and far control transfer instructions are as follows:

short	normal	far
JMPS	JMP	JMPF
	CALL	CALLF
	RET	RETF

❑ If the operand of a CALLF or JMPF instruction is a 20-bit absolute address, it is entered in the form:

```
ssss:oooo
```

where *ssss* is the segment and *oooo* is the offset of the address.

❑ Operands that can refer to either a byte or word are ambiguous and must be preceded either by the prefix "BYTE" or "WORD". These prefixes can be abbreviated to "BY" and "WO". For example:

```
INC BYTE [BP]
NOT WORD [1234]
```

Failure to supply a prefix when needed results in an error message.

- □ Operands that address memory directly are enclosed in square brackets to distinguish them from immediate values. For example:

  ADD AX,5; add 5 to register AX

  ADD AX,[5]; add the contents of location 5 to AX

- □ The following are the forms of register indirect memory operands:

  [*pointer_register*]

  [*index_register*]

  [*pointer_register* + *index_register*]

  The *pointer_registers* are BX and BP, and the *index_registers* are SI and DI. Any of these forms can be preceded by a numeric offset. For example:

  ADD BX, [BP+SI]

  ADD BX, 3[BP+SI]

  ADD BX, 1D47[BP+SI]

E

Using Microsoft Windows

This appendix contains notes about using Microsoft Windows with DR DOS.

E.1 Windows and DR DOS

Windows and RAM

If you experience problems when Windows finds Random Access Memory (RAM) in upper memory on your system, you can use the DR DOS MEMMAX command to disable these areas of memory before you load Windows.

Refer to the MEMMAX command description in Chapter 10, "DR DOS commands" for more information.

Windows and EMM386.SYS

EMM386.SYS works with Windows 3.0 in Real, Standard, and Enhanced modes. Note, however, that the Windows SETUP program automatically changes your **config.sys** file to include the HIMEM.SYS and SMARTDRV.SYS device drivers. Since the DR DOS EMM386.SYS (and HIDOS.SYS) memory managers and Super PC-Kwik disk cache replace these, you should remove the changes to **config.sys** during Windows SETUP or manually edit **config.sys** with the DR DOS Editor after loading Windows.

Windows in Standard mode with EMM386.SYS

To use Windows in Standard mode with EMM386.SYS use the new EMM386.SYS switch /WINSTD. Refer to the description of EMM386.SYS in Chapter 11, "Customizing your system" for more information about /WINSTD.

Windows and Super PC-Kwik

Load the PCKWIN.SYS driver to run Windows in Standard or Enhanced mode with Super PC-Kwik. You can load PCKWIN.SYS during INSTALL or SETUP (when you are prompted about using Windows) or by editing the **config.sys** file yourself, using the Editor.

Refer to the section about using Windows with Super PC-Kwik in Chapter 13, "Optimizing your disk" for further information.

Windows and TaskMAX

You can run Windows as a task under DR DOS TaskMAX, but you cannot switch tasks while you are running Windows; exit Windows to start switching tasks again. You can use the Windows task switcher while running under TaskMAX, but do not open applications already open in TaskMAX.

Windows and SuperStor

Exit Windows before using the SuperStor file compression program to create a compressed disk. Also, remember that Windows Permanent Swap Files cannot be stored on a SuperStor compressed disk.

F

Using Novell NetWare workstation shells

THIS appendix contains technical notes about using Novell NetWare with DR DOS, and TaskMAX in particular.

F.1 Using Novell NetWare with DR DOS and TaskMAX

The NetWare workstation shell is Novell's basic DOS shell. You may require a different shell for a specific application. Contact your local authorized Novell reseller if you believe another shell is needed.

Note the following when using Novell NetWare with DR DOS and TaskMAX:

❑ You must update your **NET3.COM, EMSNET3.EXE**, or **XMSNET3.EXE** with the **NETX.COM, EMSNETX.EXE**, or **XMSNETX.EXE**. For example, if you are using **NET3.COM**, replace your **NET3.COM** file with the **NETX.COM** file found in the **NETWARE** subdirectory on the DR DOS release diskettes.

❑ **IPX.OBJ** v3.02 is included on the release diskettes. You should update your **IPX.COM** file to this version. In order to do so, you will need to run the SHGEN or WSGEN NetWare utility. Refer to your NetWare documentation for more information about using these utilities.

To check your IPX version (IPX allows the NetWare shell to communicate with a NetWare file server), type the following at the command line where you start NetWare:

```
IPX I
```

❑ If you are using the NetWare utilities MAP, LOGIN, and SESSION from more than one command prompt via the task switcher, the resulting drive mappings may not work as expected. You should map drives globally with these utilities before loading TaskMAX.

❑ The **RPLFIX.COM** program is required to allow workstations to remote program load properly with DOS. See the following section in this appendix.

❑ If network applications which make calls to IPX or SPX (sometimes called peer-to-peer applications) are run under TaskMAX, read the **TBMI2.DOC** file in the **NETWARE** directory of your DR DOS release diskettes.

Installing RPLFIX.COM

RPLFIX.COM allows workstations to remote program load (RPL) properly with DR DOS. You must run **RPLFIX.COM** after the boot image file (usually **NET$DOS.SYS**) has been created with the DOSGEN utility. **RPLFIX.COM** will modify the boot image file. RPLFIX needs to be run against the boot image file once; the utility will detect attempts to run RPLFIX against a boot image file already modified, and will make no further modifications.

Follow the instructions for DOSGEN (setting up remote reset workstations) in the Novell reference manuals. After you perform all the steps as outlined in the manual to create the boot image file, run **RPLFIX.COM**.

The remote workstation may "hang" during the reset process if you are resetting using DR DOS and the boot image file has not been modified by RPLFIX.

You can find RPLFIX in the **NETWARE** directory. After you have located **RPLFIX.COM**, map a drive to the LOGIN subdirectory on the file server you want to remote reset from. The DOSGEN procedure instructed you to create a boot image file in that subdirectory (usually named **NET$DOS.SYS**). If the drive you mapped to SYS:LOGIN was drive F:, then you would enter the following:

```
RPLFIX F:NET$DOS.SYS
```

Note that if you renamed your boot image file, then you must use the new filename with RPLFIX. The command format is:

```
RPLFIX [d:]filename
```

where *d:* is the drive letter where the image file is located and *filename* is the name of the file created with the DOSGEN utility.

Glossary

Ambiguous filename

A filename that contains either of the DR DOS wildcard characters ? or *. When you replace characters in a filename with these wildcard characters, you create an ambiguous filename and can easily reference more than one DR DOS file in a single command.

Application

A piece of software used to carry out a particular task, like a word processor or graphics package.

ASCII

The American Standard Code for Information Interchange is a standard code for representing numbers, letters and symbols. An ASCII text file is a file that can be intelligibly displayed on the video screen or printed on paper.

Autoexec.bat file

This is a configuration file which DR DOS executes every time it is started. The easiest way to change the DR DOS configuration parameters in the **autoexec.bat** file is to use the SETUP program.

Backup

A copy of a file, directory or disk made for safe keeping.

Batch file

A file containing a sequence of commands. When you type the name of the file at the command line, each command in the file is carried out in turn, just as if you had typed the commands individually. The filename becomes in effect a "command" in itself. Batch filenames can be up to eight characters in length, but must always have the extension **.bat**.

Baud rate

The baud rate is the number of bits transmitted per second in serial communications.

Bit

A bit is the smallest unit of memory in a computer. Bits are electronic switches. They are given values to indicate whether they are switched on or off, 0 is off and 1 is on. Eight bits are equal to one byte, or character. See also *Byte.*

Beurnoulli Drives

A cartridge drive system supplied by Iomega Corporation. The Bernoulli cartridges are high-capacity removable media.

Buffer

An area of memory that temporarily stores data during information transfer.

Built-in command

See *Internal* command.

Byte

A unit of memory or disk storage usually used to represent one character. See also *Bit.*

Cluster

A group of contiguous sectors on a disk; a sector is the smallest unit of allocation for files on the disk.

Code page

A code page is a table stored in your computer which defines the particular character set required by a specific national language.

Code page switching

This is a mechanism DR DOS uses to allow you to use the range of characters needed in a variety of national languages. Using it you can switch between the different character sets needed by differing languages.

Command

A short program in DR DOS designed for a specific task.

Command line

The command followed by any other relevant information given at one time to the computer, to enable it to carry out your instructions. The command line ends when you press the **Enter** key.

Communications port

A socket to which another computer may be attached usually via a cable, thus allowing the computers to communicate with each other.

Config.sys file

A configuration file which DR DOS executes every time it is started. It contains various commands used to set up your system for items like keyboard and country information. The easiest way to change the DR DOS configuration parameters in the **config.sys** file is to use the SETUP program.

Conventional memory

Memory below 640 Kbytes, sometimes called base memory. It is generally where the operating system and applications are loaded. It is usually referred to as RAM (Random Access Memory). See also Appendix C, "Memory".

Current disk drive

The drive on which DR DOS first looks for files. The current drive letter is usually displayed in the system prompt.

Cursor

A blinking line or block on the screen that indicates where your next keystroke will have an effect.

Databits

Refers to the number of bits used to define a character on serial data transmission, usually 7 or 8. When connecting two computers together via the communication ports, it is important to set the same number at each end.

Default

A pre-set value used by a computer until it is specifically changed by a user.

Device

A device is a piece of hardware attached to a computer (usually a peripheral), for example a printer or mouse.

Device driver

Device drivers are programs that are loaded alongside the operating system to operate hardware devices which are not recognized as a default part of the system. Device drivers usually have the extension **.sys**.

Directory

A place on disk where filenames are stored so that DR DOS can find the files when needed. Each disk can contain many directories.

Disk

Magnetic medium used for storing information.

Disk Accelerator

See *Disk cache.*

Disk cache

A disk cache or cache is an area of RAM memory used to store data accessed frequently from disk. If your applications want to access a disk for data, it can become time-consuming. To speed up the process of reading and writing data to and from disk, you can set up a cache. Whenever data is read or written from disk, a check is made to see if a copy is in the cache first. If it is in the cache, the copy is used. Caches are very useful for increasing the performance of applications that have to access a lot a data from disk, like databases and spreadsheets.

Diskette

A disk that can be removed from the drive (also known as floppy disks).

Disk operating system

A collection of programs that manages computer resources and other programs on a computer.

Drive

The device that holds and reads a hard disk or diskette.

EDITOR

A DR DOS program that allows you to create and edit ASCII text files on a computer.

EMM

Expanded Memory Manager; see also Appendix C, "Memory".

EMS

Expanded Memory Specification; see also Appendix C, "Memory".

Enter key

The key that tells DR DOS to carry out a command. Also referred to as **Return** or ↵.

Environment

A source of information within DR DOS used by all applications, often called the global environment. It is where the PATH, PROMPT and SET variables are stored.

Executable

This means "programs that can be run by the computer". Executable code is a series of instructions that can be carried out by the computer. Executable programs usually have the extension **.exe** or **.com**.

Expanded memory

Expanded memory is often known as LIM memory. The LIM EMS standard was jointly devised by Lotus, Intel and Microsoft. EMS is the Expanded Memory Specification. It describes ways in which a program may access up to 32 Mbytes of memory outside conventional memory. Programs that use expanded memory include spreadsheets, databases, and other programs that need access to large amounts of data. For more details see Appendix C, "Memory".

Extended memory

Memory above 1 Mbyte on 80286, 386 and i486-based computers; see also Appendix C, "Memory".

External command

Commands that are used less often than other commands, and so are not automatically loaded into memory. To save space they are stored on disk instead. They have the file extension **.com** or **.exe**. When you enter an external command, DR DOS retrieves the "command file" from disk and runs it. The memory used by the command is freed when it has finished its work. See also *Internal command.*

File

A collection of related instructions or data stored on disk.

File Allocation Table (FAT)

The File Allocation Table is a kind of index which exists at the start of every disk, and is used by the operating system to locate entries for files.

FILELINK

A DR DOS utility used to transfer files between two computers connected by their serial communication ports.

Filelist

A list of filenames in a file. The file containing the filelist is identified by an @ character in front of its filename. Some commands that usually operate on only one file per command line can, by using a filelist, be made to operate on several files. See Chapter 3, "Managing you information".

Filename and filename extension

The name assigned to a file. A filename can include a primary filename of 1 through 8 characters and optional filename extension of 1 through 3 characters. A period (.) separates the filename from the filename extension.

Filespec

Short for file specification. The combination of drive letter, path, and filename that identifies a file uniquely to DR DOS.

Fixed disk

See *Hard disk.*

Floppy disk

See *Diskette.*

Format

The process by which DR DOS prepares a disk for use by writing electronic markers onto the disk, so that DR DOS can store and retrieve files. FDISK is used to prepare partitionable disks for use. FORMAT is used to format disks and diskettes. When you format a disk, except in special circumstances (see the description of FORMAT in Chapter 10, "DR DOS commands") all the information previously stored on it is erased, so take care when using this command.

Hard disk

A disk for storing information that is not removable from the computer. They usually have large storage capacity and provide fast access to data.

Hexadecimal

Hexadecimal numbers are numbers to base 16. Hexadecimal uses the numbers 0-9 and A, B, C, D, E, F. The letters are equivalent to the numbers 10, 11, 12, 13, 14, 15 in decimal. Hexadecimal numbers are used as a shorthand for binary.

High memory

The first 64 Kbytes of extended memory; see also Appendix C, "Memory".

Image file

A file used to store an *image* of a diskette; that is; an exact copy of the diskette's contents.

Input

Information going into the computer, usually from you typing at the keyboard or from a program reading from disk.

I/O

Abbreviation for input/output.

Internal command

A DR DOS command that usually resides in memory. These commands respond quickly because they do not have to be loaded from disk.

Kilobyte (Kbyte or KB)

1024 bytes, denoted as 1Kbyte, 1 KB or 1 K. 32 Kilobytes equal 32 Kbytes. 1024 Kbytes equal 1 Megabyte (or Mbyte), over one million bytes.

Label

A marker within a batch or **config.sys** file. It is used to indicate the specific point that execution jumps to, following a GOTO statement.

Lend

The process wherein the Super PC-Kwik cache monitors requests from other application programs for expanded or extended memory, and gives up some of the memory allocated to its cache buffers when the other programs request memory. When the other applications are finished, Super PC-Kwik returns the memory to its cache buffer allocation.

LIM

See *Expanded memory.*

Logical drive

The operating system's internal representation of a drive. It may refer to an actual disk device, or to a group of directories specified using the SUBST command.

Lower memory

The lowest part of conventional memory, where the operating system and installable device drivers are commonly loaded. See also Appendix C, "Memory".

Megabyte (Mbyte or MB)

A unit of storage in a computer denoting 1024 Kbytes. It is usually denoted as 1 Mbyte or 1 MB.

Memory

Memory is where the computer stores data and programs. It is measured in bytes. One byte is equal to a group of eight bits. A bit is the smallest unit of memory in a computer. Each bit has the value 0 or 1. Memory is thought of in bytes because one byte of memory can be used to represent a character such as "a", "?", or "4". One thousand and twenty-four bytes is equal to one *Kilobyte* or *(Kb)* for short. As computers contain a lot of memory, memory is usually talked about in terms of Kilobytes rather than bytes. See also Appendix C, "Memory".

Memory disk

A portion of computer memory that is made to act like a very fast disk. You store your files on it in the same way as on a physical disk. Information stored on a memory disk is lost when the computer is switched off or when DR DOS is restarted. Memory disks are therefore best used for temporary storage of expendable data. They are also referred to as *RAM* disks or Virtual disks.

Modem

*Mod*ulator/*Dem*odulator. A device which converts a digital signal from a computer into an analogue signal which can then be transmitted down telephone lines and decoded by a modem at the receiving end.

Online

A computer or peripheral is “online” when it is switched on and ready to accept information.

Operating system

See *Disk operating system.*

Output

Data that the computer sends to the console, disk or some other device.

Parallel

The carrying out of several tasks at the same time. For example in parallel communications, 8 bits of a byte would be transmitted together (see also *Serial*).

Parity

A means of checking that data has been transferred correctly, either between computers across a communications link, or between components inside a single computer, such as a disk and memory.

Partition

A partition is a section of a hard disk created by the FDISK disk preparation utility.

Password

A series of characters needed to gain access to a computer, file or directory.

Path

The description of the route through the directory hierarchy to a subdirectory or file. A drive letter can also be included at the beginning of the path.

PC

An abbreviation for personal computer.

Peripheral

An external device connected to the computer. Peripherals are generally used for input and output. Examples include disks, modems, and printers.

Piping

The process where the output from one program is made to be the input for another.

Program

A series of specially coded instructions that perform a specific task when executed by a computer.

Prompt

The display on the screen that shows you DR DOS is ready to receive a command. The standard DR DOS prompt can be changed by using the PROMPT command.

RAM

Random Access Memory is a type of memory that is used to load and run programs from. It can be thought of as the computer's workspace. The more complex a task you want to do, the more RAM memory your computer needs. It is memory that can be read to, as well as written from, in any order. See also *ROM* and *Shadow RAM*.

RAMdisk

See *Memory disk*.

Read-only (RO)

An attribute that can be assigned to a file or directory. When switched on, the read-only attribute allows you to read from the file but not to write any changes to it.

Real time

A system where execution depends on critical timing criteria. For example a system may be required to respond to some event within a given time.

RETURN

See **Enter** key.

ROM

Read-Only Memory. This is a type of memory which you can only read or copy; you cannot alter it's contents. It is normally used to contain the computers diagnostic programs, and a very basic input/output system (or BIOS).

Root directory

The main directory that DR DOS creates on disk when the disk is formatted.

Scrolling

Scrolling means to move text up or down on a screen so that you can read it all.

Serial

The carrying out of tasks one after the other. For example in serial communications, each byte is transferred one bit at a time (see also *Parallel*).

Session

A session at your computer is the time between switching on and when you next switch off.

SETUP

A DR DOS menu-driven program which allows you to change the standard DR DOS configuration.

Shadow RAM

RAM in upper memory into which data and code can be copied from ROM. The RAM is given the same addresses as the ROM, so that in effect shadows the ROM. The term Shadow RAM is used to describe any RAM in upper memory that can potentially be used for shadowing ROM.

SID debugging tool

A powerful debugger which allows you to test and debug programs interactively; see Appendix D, "The SID debugging tool".

Spooler

A print spooler provides a queueing system for files you want to print. This way files can be printed while your computer performs other tasks.

Stopbits

A term used in serial data transfer to distinguish where one character starts and another stops, communication programs normally add one, or sometimes two, stopbits on the end of each character. Like databits, it is important that the number of stopbits set up agrees at both ends of the line.

Strings

A sequence of characters, such as "hello" or "arc132".

Subdirectory

Any directory that is not the root directory on a disk can be called a subdirectory.

System prompt

The display on the screen that shows you the operating system is ready to receive a command. The standard system prompt can be changed by using the PROMPT command.

Timeout

DR DOS sets a time limit for response to a command, so it can detect if something is taking too long to respond (for example a printer may have inadvertently been switched off), and return an error message.

TSR (Terminate-and-Stay-Resident) program

A program that leaves some of its code in the computer's memory when you exit the program and return to the DR DOS command prompt. Therefore it does not have to be reloaded from disk every time it is executed.

Upper memory

Memory between 640 Kbytes and 1 Mbyte (384 Kbytes) reserved for use by the video display adapter, add-on adaptors, and system software; see also Appendix C, "Memory".

Upper Memory Blocks (UMBs)

Memory located above the normal DOS limit of 640Kbytes and below the beginning of extended memory at 1024Kbytes (1 Mbyte) that is addressable by 16-bit processors. Parts of this memory are reserved for DOS and BIOS functions.

Utility

A DR DOS program that enables you to perform certain operations such as copying, erasing, or editing files.

Variable

In batch files, a variable is a character that is substituted for a filename when the batch file is run.

Video memory

A special area of memory used by the hardware which operates the computer's display or monitor.

Virtual disk

See *Memory disk.*

Volume label

The name given to a disk for identification. It is set by the LABEL command. You can display the volume label on a disk using the VOL command.

Wildcard characters

Special characters that match certain specified items. In DR DOS there are two wildcard characters, ? and *. The ? can be substituted for any single character in a filename and the * can be substituted for one or more characters in the filename or filename extension, or both.

Wildspec

The same as *filespec* except that you can use wildcard characters in the command syntax to specify groups of files.

Write-protected

A write-protected disk, directory or file can be read but not altered or erased. Disks are write-protected physically, usually by the absence or presence of a notch on the diskette casing. Files are write-protected via the ATTRIB command (setting the read-only attribute), and files and directories can be write-protected via the PASSWORD command.

XMS

Extended Memory Specification, which defines a protocol that controls access to high, upper, and extended memory on Intel 80286, 386, and i486-based personal computers.

Index