Hyperion

LEARNING DOS 2.11

LEARNING DOS 2.11
LEARNING DOS 2.11
LEARNING DOS 2.11
LEARNING DOS 2.11
LEARNING DOS 2.11
LEARNING DOS 2.11
LEARNING DOS 2.11
LEARNING DOS 2.11
LEARNING DOS 2.11
LEARNING DOS 2.11
LEARNING DOS 2.11
LEARNING DOS 2.11
LEARNING DOS 2.11

Comterm

LEARNING DOS 2.11

This tutorial booklet is a beginner's guide to using DOS 2.11 and EDLIN on the Hyperion.

As well, there are chapters of interest to more experienced individuals who are interested in DOS 2.11's special features and enhancements such as tree-structured directories, redirection of standard input and output, and batch file subcommands.

Published by: Comterm Inc.
1 July 1984
Version 00
620035 Rev 00

Trademarks

Hyperion	is a trademark of Comterm Inc.
MS-DOS	is a trademark of Microsoft Corporation.
Microsoft	is a trademark of Microsoft Corporation.
MULTIPLAN	is a trademark of Microsoft Corporation.
IBM	is a trademark of International Business Machines Corporation.

Disclaimer

The information in this manual has been carefully prepared and checked for completeness and accuracy. There is however, always the possibility of omission or error. In such an event Comterm Inc. cannot assume liability for any damages resulting from the use of this manual.

TABLE OF CONTENTS

...continued

TABLE OF CONTENTS (cont)

...continued

TABLE OF CONTENTS (cont)

...continued

TABLE OF CONTENTS (cont)

INTRODUCTION

INTRODUCTION

Why Use a Tutorial?

This book, *Learning DOS 2.11*, is provided as a guide to Microsoft's Disk Operation System, DOS 2.11.

We feel that there is something for everyone in the tutorial. New users are introduced to the conventions and procedures of the operating system. Experienced users, those who have worked with DOS 1.1 or other operating systems, may be more interested in the sections on advanced features: the redirection of input/output, and batch files, for instance.

What is DOS 2.11?

DOS is an acronym for Disk Operating System. The number 2.11 indicates the version number of the system.

When DOS was first introduced, it was designed to handle diskettes storing about 320K bytes of information. This version of the operating system has changed to permit more storage on diskettes, and on hard disks as well. In addition, DOS can now control the storage of more files and handle them more efficiently. Also, with this version of DOS, new commands have been added, along with some new features for existing commands.

Microcomputers Using DOS

DOS can be used on a number of microcomputers, including the Hyperion and the Hyperion PC. A description of the computers in the Hyperion family is provided in Appendix B. You should read over Appendix B to learn about some of the features of your system, either the Hyperion or the Hyperion PC.

Preparing to Use DOS 2.11 and the Tutorial

Although you can learn a lot by simply reading this book, you should have your computer running DOS 2.11 and try each lesson. In this manner, you can try each example, and do some experimenting as well.

The step-by-step instructions in this manual have been designed so that each activity follows from the previous activity. We suggest you follow the steps exactly as described when first trying procedures with which you are not familiar. Use Lesson 1 to set up your system, and then continue on to any of Lessons 2 to 6.

Before you continue, though, you should read the *Hyperion Setup Guide* which is supplied with each Hyperion computer. As well, the *Hyperion DOS 2.11 Guide* provides a complete reference for all DOS commands.

Typographical Conventions

In this guide, we use certain conventions to represent different types of information:

a) words which are printed in amber are commands to be entered, or, if they are surrounded by a black border, are part of a representation of the screen display.

b) words which are bold-faced and italicized are key features in DOS. These are usually explained in the Glossary, at the rear of this guide.

c) certain standard names have been used for all special keys. The names displayed on the keys may vary somewhat from machines to machine. Consult the table in Appendix B if you are unsure of which keys are which.

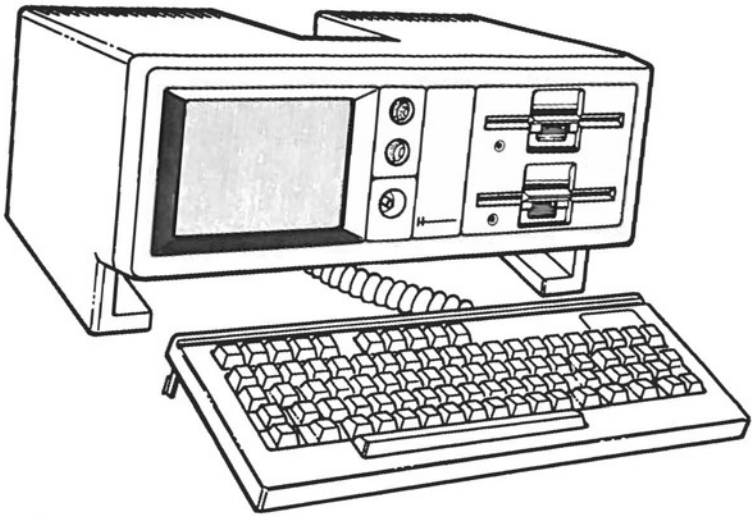


Fig. 1 — The Hyperion, plugged in and ready for use.

Lesson 1

STARTING DOS 2.11

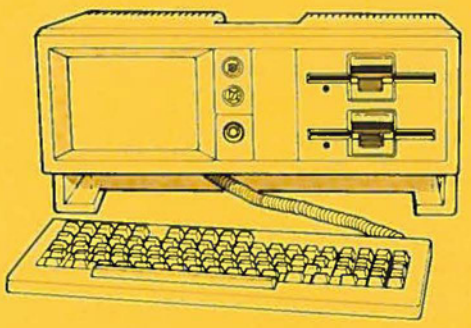


Fig. 1-1: In order to start this tutorial, you need a Hyperion, a master DOS diskette, a Supplemental Programs diskette, and one new unformatted diskette.

Lesson 1

STARTING DOS 2.11

1.1 INTRODUCTION

This lesson introduces you to the Hyperion. It tells you how to start up (or *boot*, as many computer users say) your system. Steps 1 to 7 must be followed every time you start your Hyperion. When you follow the steps described in all subsequent lessons of this tutorial, it is assumed that you have already followed steps 1 to 7.

1.2 THINGS YOU REQUIRE

In the plastic sleeve at the back of the *Hyperion DOS Guide*, there are 2 diskettes which can be used in the Hyperion: the master *DOS diskette* and the *Supplemental Programs diskette.* When the master DOS diskette is inserted in drive A and the Hyperion started by pressing the amber power button, DOS (the **D**isk **O**perating **S**ystem) is read or "loaded" into the Hyperion. This creates an "environment" in which other programs may operate.

For this tutorial you require the following:

* Your Hyperion should be set up as shown in Fig. 1-1, but not powered on (the amber power light should not be illuminated).

* If you have an expansion unit, do not use it for this tutorial. Set up and use the expansion unit only after becoming familiar with the Hyperion main unit.

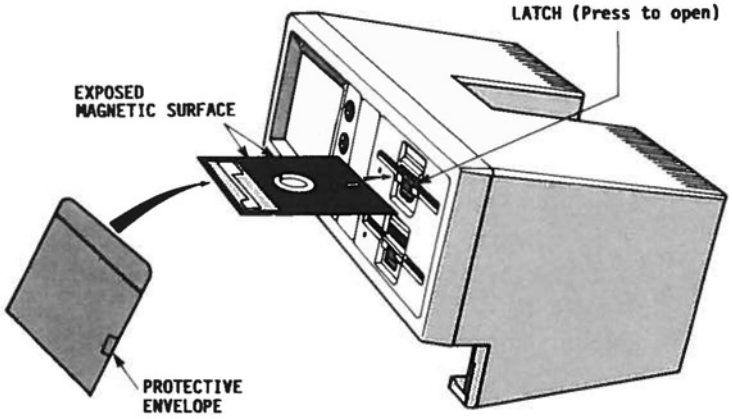


Fig 1-2: Inserting the master DOS diskette. (Your Hyperion may have only one diskette drive, drive A.)

* You should have the master DOS diskette.
* You should have the Supplemental Programs diskette.
* You should have a blank, unformatted diskette.

1.3 INSERTING THE MASTER DOS DISKETTE

The master diskettes can be read, but cannot be modified in any way, as they are *write-protected.* Master diskettes are not meant to be used regularly, but should be copied onto other diskettes, and the copies used. The master diskettes should be stored in a safe place and used to produce more copies in case your working DOS and Supplemental Programs diskettes are ever damaged.

For the first part of this tutorial, however, you need to use the 2 master diskettes. You will be shown how to make copies of them in Section 2 of the tutorial.

STEP

1) Remove the master DOS diskette from the back of the *Hyperion DOS 2.11 Guide.*
2) Slide it out of its protective envelope. DO NOT TOUCH THE EXPOSED MAGNETIC SURFACE.

...continued

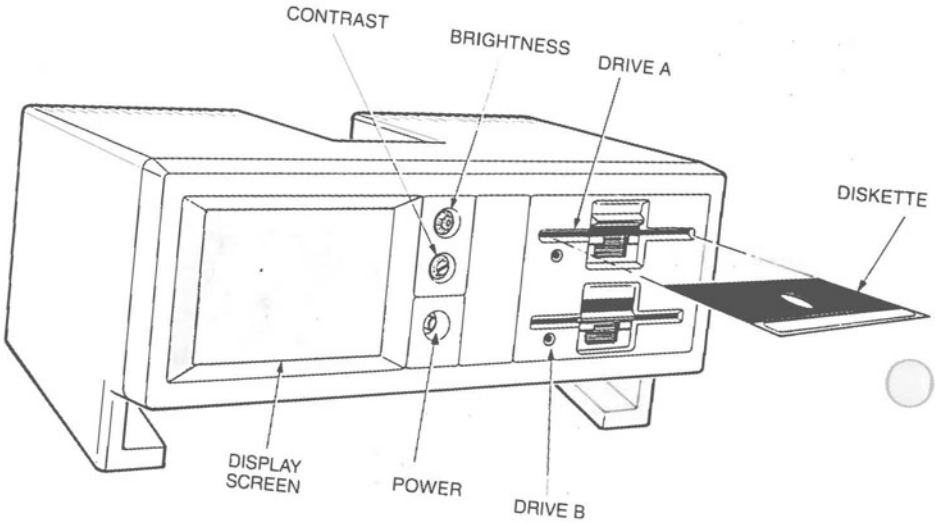


Fig. 1-3: Press the power button to turn on the Hyperion.

STEP *(cont)*

3) Insert the diskette into the uppermost diskette drive slot (drive A) on the front of the Hyperion, as follows (see Fig. 1-2):

— open the disk drive latch by pressing on the lower half of the latch,

— hold the diskette with the labels on top and towards you, and insert the diskette all the way in, until it catches,

— close the disk drive door by pressing down on the upper half of the drive latch until it clicks shut.

1.4 TURNING ON THE HYPERION

Once the master DOS diskette is inserted into drive A, you can turn on the Hyperion. DOS is then loaded and becomes the operating system controlling the Hyperion. *"Loading"* refers to the fact that the programs contained on the DOS diskette are copied (loaded) into the Hyperion's memory.

STEP

4) Press the amber power button shown in Fig. 1-3. This button lights up, to indicate that the computer is receiving power.

...continued

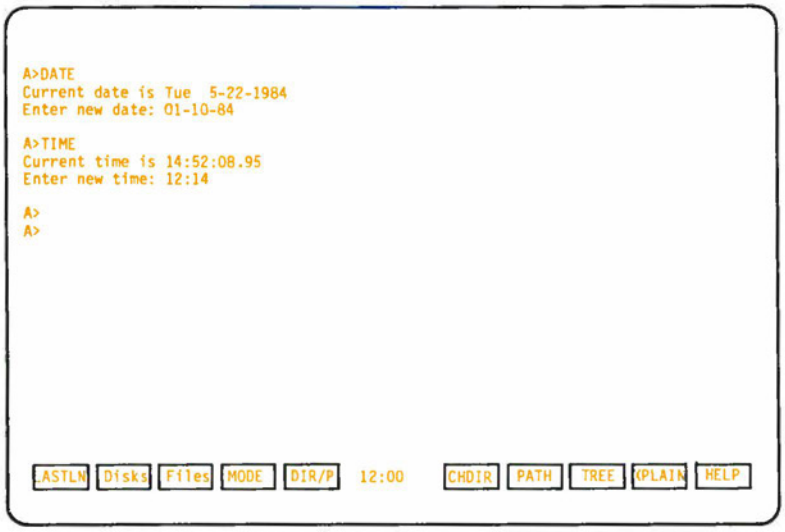


Fig. 1-4: System startup messages, and the system prompt.

The System Self-Test

Whenever the Hyperion is powered on, it goes through an extensive self-test of all internal electronics. This test takes about 30 seconds, during which time nothing is shown on the screen except a flashing cursor.

The Waiting State

As well, it is important to remember that the Hyperion screen, after three minutes of inactivity, is turned off to save power and to prolong the life of the equipment. Striking any key turns the screen back on again, displaying the identical screen. The waiting state does not affect any operation in process, and you can simply pick up where you left off. The key pressed to turn the screen back on does not appear on the screen.

1.5 SYSTEM STARTUP MESSAGES

As soon as the self-test is completed, the system begins to execute the AUTOEXEC.BAT file that is contained on the DOS diskette. This AUTOEXEC.BAT (batch) file displays the Hyperion logo (a star), and then sets the operating environment using specifications contained in a MODE subprogram. The automatic batch file then prompts you for the date and time.

If a Diskette is Missing

If there is no diskette in drive A, it looks for one in drive B. After both drives have been searched unsuccessfully several times, the message DISK FAULT is displayed. The system must be restarted, by putting a system diskette into drive A and pressing the POWER button off and then on.

If the Information Is Not There

If there is a diskette, but without the needed information, the system displays the following message:

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

You have inserted the wrong diskette, or have put it in improperly. Reinsert the master DOS diskette and press any key on the keyboard. The system looks for the needed information again, and displays the Hyperion star logo followed by the startup messages shown in Fig. 1-4. If the system does not start up, press the **Ctrl**, **Alt** and **Del** keys simultaneously (**Ctrl + Alt + Del**) to reboot the system.

Adjust Brightness and Contrast

STEP *(cont)*

5) The two circular controls above the power switch

are used to adjust the brightness  and

contrast  on the screen. Turn these

controls to produce the most comfortable display. See Figure 1-3.

...continued

1.6 SETTING THE DATE AND TIME

The system prompts you to enter the current date and time. The current date and time are entered whenever you start DOS.

Set System Date

STEP *(cont)*

6) Enter a new date in month-day-year (mm-dd-yy) format, including hyphens; for example, 01-10-84 would indicate a date of January 10, 1984. Press the **Return** key. (See Fig. 1-4.)

Set System Time

After you have entered the current date, the system prompts for the current time.

STEP

7) Enter the new time as hours:minutes (hh:mm), with the colon, and press the **Return** key (See Fig. 1-4).

...continued

System Clock

The system clock now shows the new time you have entered, displayed in the center of the bottom line (the soft key line) of the screen.

1.7 CHANGING THE SYSTEM PROMPT

STEP *(cont)*

8) Look at the last line. The A> is called the *system prompt* and is followed by a blinking underline called the *cursor*.

...continued

The prompt is a character (or characters) whose presence tells you that the computer is ready to receive commands from you. The prompt is usually a letter (i.e. A, B, C, etc.) followed by the "greater than" sign (>).

When the prompt is not displayed, it usually means that DOS is performing a command and is not "listening" at the moment. Depending on the command, though, the prompt usually returns within a minute or two.

The blinking underline after the prompt is the *cursor*. It represents "where you are" on the screen. Any characters you type appear at the cursor position, and the cursor moves to the next space.

Changing the System Prompt (cont)

The letters (A, B, etc.) of the prompt show which drive the system is "looking at" for information. To change the letter, and where the system looks for information, simply type in the letter of the desired drive:

STEP *(cont)*

9) Type in the letter **B** followed by a colon (:), and press the **Return** key.

The prompt should now read "B>". When entering a DOS command, the system now looks at the diskette in drive B for information. The drive corresponding to the prompt letter is called the *default* drive, since DOS looks there for information by default if no drive is specified. If you have a single-drive Hyperion, DOS treats the one physical drive as two *logical* drives called A and B.

STEP

10) Type in the letter **A** followed by a colon (:), and press the **Return** key.

...continued

The prompt now reads "A>" once more.

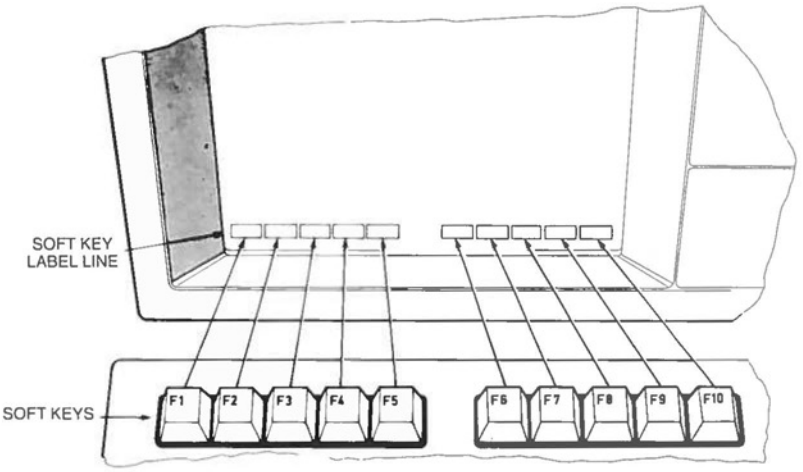


Fig. 1-5: The soft key line, labels, and soft keys.

1.8 THE SOFT KEYS

With Soft Keys Displayed

If the MODE setting for the Hyperion's soft key display is turned on, a row of 10 highlighted boxes across the bottom of the screen appears simultaneously with the system prompt. This is called the *soft key line*, and is divided into two sets of five boxes by a clock displaying the time of day.

Each highlighted box on this line is a label. The label describes the command or feature that can be accessed by striking the corresponding *soft key*. The soft keys are the ten upper left keys on the Hyperion keyboard, **F1** through **F10**.

If the Soft Key Line Is Not Displayed

When DOS is booted and MODE is not executed (if alterations have been made to the AUTOEXEC.BAT file) or if MODE has been altered to turn Hyperion characters off, the soft key line is not displayed. If this happens, you have 3 options:

a) Reset the MODE screen function by typing in the word **SETHYP** and pressing the **Return** key.

b) Set the mode key functions to emulate IBM by typing in the word **SETIBM** and pressing **Return**. MODE functions are then IBM-type, and can either be manually entered or the keys can be translated to IBM using the MODE SOFTKEYS function.

c) Enter DOS commands manually by typing in the desired command, as well as each parameter.

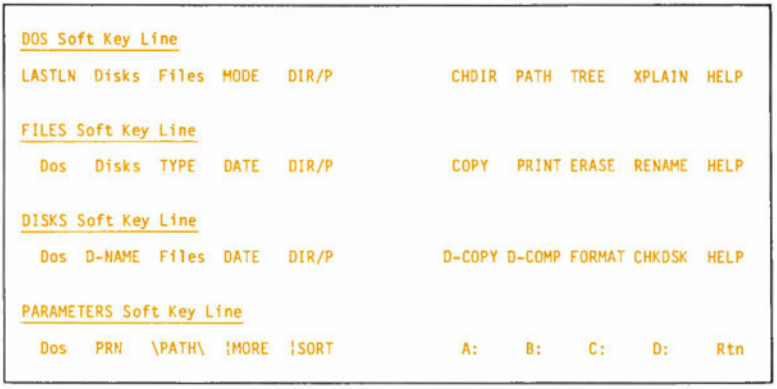


Fig. 1-6: The four soft key command lines in the DOS system.

If you change the MODE settings (options a and b), you must re-boot the system by pressing the **Ctrl**, **Alt** and **Del** keys simultaneously. The MODE command is described in Part II of the *Hyperion DOS 2.11 Guide*.

Displaying Different Soft Key Lines

The soft key labels displayed on the screen can change. As the soft key line labels change, the function of each soft key also changes. Soft keys are so named because their functions can change. The *hard* keys (the alphabetic and numeric keys for example), on the other hand, have permanent functions.

STEP *(cont)*

11) Press the soft key **F3** (labelled **Files**). Notice that labels in the soft key line have changed.

12) Press the soft key **F2** (labelled **Disks**).

13) Press the soft key **F1** (labelled **Dos**).

This returns you to the same line of labels that you started from. The soft key lines are shown in Fig. 1-6.

...continued

The labels for the soft keys used to access other soft key lines are in mixed upper and lower case letters to distinguish them from the keys used to enter DOS commands, which are only upper case.

The Soft Key Line Tree

If you do not have a soft key line displayed, any of the DOS commands found on the soft key lines can be typed in manually, followed by the appropriate parameters.

Each set of soft key labels, i.e. the soft key line, displayed across the bottom of the screen is given a name in order to enable it to be identified and referred to later on.

The computer system you are using at the moment is called the *Disk Operating System (DOS)*. The soft key line currently displayed is the main DOS soft key line and is called "DOS".

This particular system has two other soft key lines used for entering commands: the "DISKS" line, with commands mainly concerned with disk management; and the "FILES" line which has commands mainly concerned with file management.

For some of the DOS commands, a special soft key line is displayed to help you enter any parameters you need for the command. This line, the "PARAMETERS" line, disappears after you choose another softkey from the Parameters line or press Return.

STEP *(cont)*

14) Repeat steps 11 to 13.

...continued

1.9 ASKING THE SYSTEM TO EXPLAIN

EXPLAIN is a DOS command which enables you to ask the system about particular commands and some system features.

STEP *(cont)*

15) Insert the Supplemental Programs diskette into drive B (drive A if the Hyperion only has one drive).

16) Press **F9** (XPLAIN) or type in the word **EXPLAIN**.

Since the soft key label is upper case, it is a *command.*

Notice that the system types the characters **EXPLAIN** on the screen where the cursor used to be. The cursor is now just to the right of the **N**. Nothing has happened. The system is still waiting for you to do something.

17) Press the **Return** key.

...continued

The screen changes to display information about how the *EXPLAIN* command works (Fig. 1-7). The message:

Strike a key when ready . . .

appears at the bottom of the screen, when there is more than one screenful of EXPLAIN information.

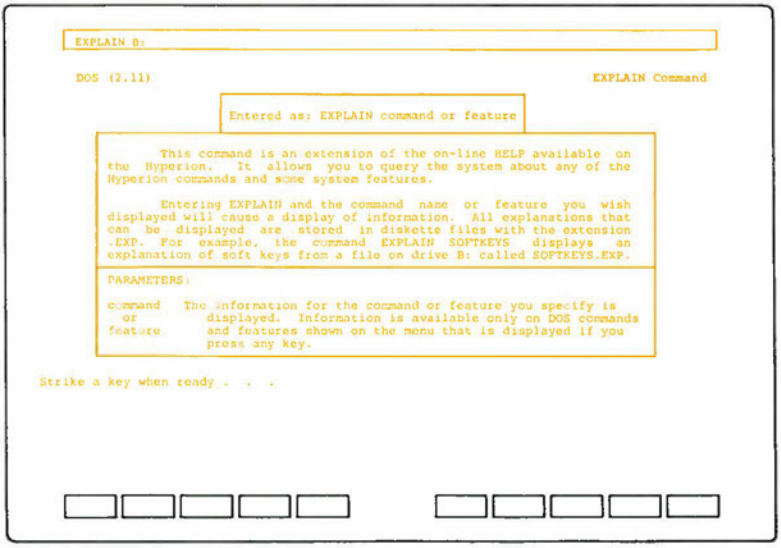


Fig. 1-7: First page of the EXPLAIN command display.

EXPLAIN (cont)

STEP *(cont)*

18) Press any character key to display the next page of information. See Figure 1-8.

...continued

The system description of the command EXPLAIN was too long to be displayed on one screen. Pressing any key advances you to the subsequent page of text which, in this case, is a list of commands and features that can be explained. The first topic, the ASSIGN command, is highlighted.

If you are using the master DOS and Supplemental Programs diskettes with no adaptations, the following message is displayed at the bottom of this second, and last, "page":

Topic choice: ASSIGN

Notice also that the soft key labels are now blank, except **F6**, **F7**, **F8** and **F9**, which contain arrows. Pressing the appropriate soft key or cursor control key moves the cursor block in the direction of the cursor or arrow to another feature.

If you do not wish to query a system feature, press the **Esc** key. This returns you to your DOS screen.

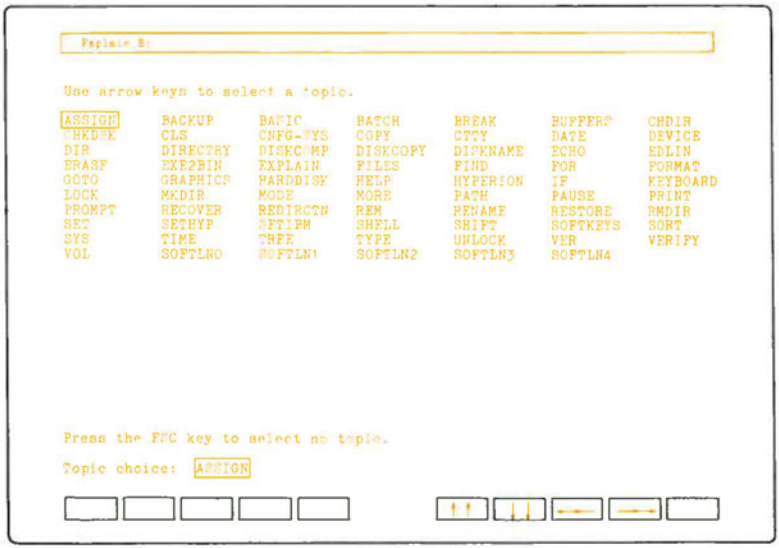


Fig. 1-8: The second page of the EXPLAIN command display.

EXPLAIN (cont)

STEP *(cont)*

19) Using the soft keys, move the cursor block to the **SOFTKEYS** feature. Note how the label in the Topic Choice field changes.

20) Press the **Return** key. The system displays information about the soft keys. Read the information displayed on the screen to learn more about soft keys.

21) Press any keyboard key to display subsequent pages of information until the system prompt is redisplayed.

The Hyperion is ready for another instruction.

22) Press the **F9** (XPLAIN) key or type **EXPLAIN**.

23) Type the word **SOFTKEYS** on your keyboard. (Typing mistakes may be corrected using the **Rub Out** key to backspace over the mistake.)

24) Press the **Return** key.

...continued

The screen immediately clears and displays information about soft keys and how they are used. This is the same screen as previously displayed when you selected the SOFTKEYS option on the menu provided by the EXPLAIN command.

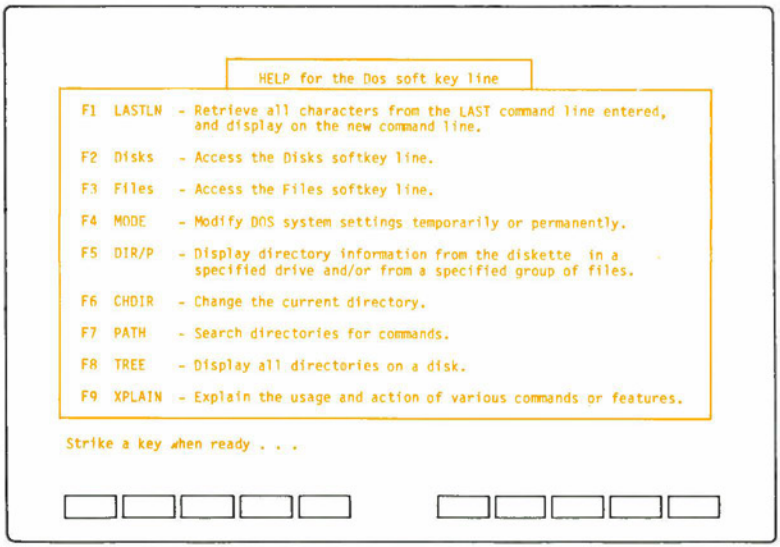


Fig. 1-9: The HELP display for the DOS soft key line.

1.10 ASKING FOR HELP

When the softkey labels are displayed, notice that the label corresponding to soft key F10 is always *HELP*, except during the execution of a command when a fourth soft key line (the PARAMETERS line) appears and F10 is labelled *"Return"*. This HELP feature enables you to display descriptions of all the commands in the current soft key line. The HELP screen provides different information, depending on which line it is accessed from.

In the next series of steps, you are going to access a HELP screen. This is done by pressing the HELP key, which sends DOS a command to display the screen describing the meanings of all the upper case commands on the soft key line.

STEP

25) Press **F10**.

26) Once you have read the contents of the screen (see Fig. 1-9), press any key to display the next page of the file. Press any key again to exit from Help and return to DOS.

27) Press **F3** to access another set of soft key labels (called the FILES soft key line).

28) Press **F10** again.

(Notice that the HELP display has changed. It now describes the new set of command labels.)

...continued

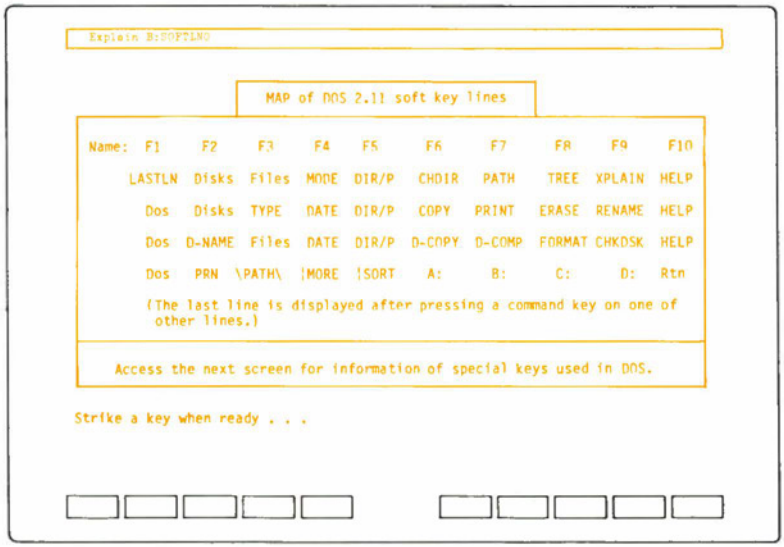


Fig. 1-10: The soft key map display, using the **Ctrl + HELP** keys.

Asking for HELP (cont)

STEP *(cont)*

29) Press any character key on the keyboard until you return to DOS.

30) Press **F1** (Dos). This returns your original soft key line, the main DOS soft key line.

When a HELP screen is displayed, pressing any character key returns you either to DOS or to the program you were executing when the HELP was requested.

Display the Soft Key Map

There is a special HELP feature called the soft key map.

The soft key map for the system can be displayed when the Supplemental Programs Diskette is in drive B and you call up the map by holding down the **Ctrl** key and pressing **F10**, the HELP soft key.

STEP

31) If the soft key display is enabled, hold down the **Ctrl** key on the left side of the keyboard, and press **F10** (HELP).

...continued

Asking for HELP (cont)

DOS executes some commands and displays a soft key map for the main operating system as shown in Fig. 1-10.

STEP *(cont)*

32) Press any key. This brings up the second screen, describing the special keys used in DOS 2.11. After you have read the contents of this screen, press any character key until you have cleared the HELP screen and returned to DOS.

HELP cannot be manually typed in, but must be accessed via the soft keys.

1.11 INTERRUPTING AND RESTARTING THE HYPERION

Many times you may find it necessary to cancel, abort, or restart a system program or function. You can do this in several ways:

a) To cancel a command that you are entering, i.e., before you press the **Return** key, press the **Esc** key. The system prompt reappears on a new line to prompt you for a new command.

STEP

33) Press the **Esc** key.

...continued

Interrupting and Restarting the Hyperion (cont)

Notice that the system puts a backslash (\) on the command line,where the cursor used to be, then redisplays the system prompt.

b) To abort a command that is currently being processed, press **Ctrl** + **Break**. This returns you to the main operating system, the Hyperion DOS.

WARNING: IF YOU ABORT CERTAIN COMMANDS, SUCH AS COPY, YOU MAY LEAVE DISKETTE FILES WITH INCONSISTENT DATA.

c) To restart the Hyperion, press **Ctrl** + **Alt** + **Del**. This procedure halts all activity within the Hyperion, clears the internal memory, and reloads DOS from the DOS diskette in drive A.

WARNING: WHEN RESTARTING THE SYSTEM, ANY ENTRIES NOT SAVED ONTO A DISKETTE WILL BE LOST.

1.12 SUMMARY OF CONCEPTS

In this section you have learned:

— about DOS,
— how to start up the Hyperion,
— how to access the soft key lines in DOS,
— to use a soft key to enter instructions (such as HELP),
— to use the soft key line and keyboard to enter commands,
— to cancel commands and perform system restart,
— how to use DOS if no soft keys are displayed.

Summary of Concepts (cont)

You have also been introduced to the following:

— **Master DOS diskette** which you must insert into the Hyperion before doing anything else.

— **System Prompt** which is the system's signal that it is ready to receive instructions.

— **A DOS command environment** without soft keys.

— **Soft Key line** which displays ten highlighted labels across the bottom of the screen.

— **Soft Key map** for the main operating system.

— **Soft Keys** to change soft key line labels and to enter instructions into the system.

— **Upper Case Command labels** to enter commands.

— **Lower Case labels** which change soft key lines to access other commands.

— **The EXPLAIN** command.

— **The Return key** which you use to tell the system that you are ready for it to begin processing your command.

— **The Rub Out key** used to backspace over mistakes.

— **The Esc key** used to cancel the entry of a command.

— **The Ctrl + Break keys** to cancel the processing of a command.

— **The Ctrl + Alt + Del** keys to restart the system.

Lesson 2

DISKETTE PROCEDURES

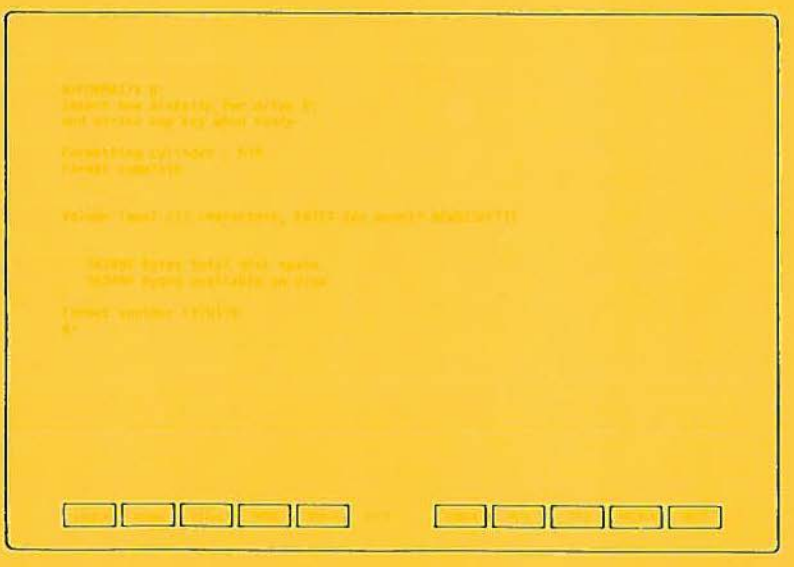


Fig. 2-1: To format a new diskette, use the FORMAT command.

Lesson 2

DISKETTE PROCEDURES

2.1 INTRODUCTION

In this lesson of the Learning DOS 2.11 tutorial, a number of disk-oriented features of DOS are described. In order to follow the steps outlined in this section, you must have booted (started) the system using steps 1 to 7 in the previous section. Make sure the system prompt displayed on your screen reads "A>". If it does not, type in **A** followed by a colon (:) and press **Return**.

You should have the master DOS diskette in drive A, and drive B (on a system with two drives) should be empty.

2.2 FORMATTING A NEW DISKETTE

When you began this tutorial, you were instructed to obtain a new unformatted diskette. This new diskette, like all new diskettes, must be formatted before it can be used with the Hyperion.

To format a new diskette, you must use the FORMAT command.

STEP

1) Press the **DISKS** soft key (F2) from either the DOS or the FILES soft key line. (If soft keys are not displayed, skip this step.) The DISKS soft key line is displayed.

...continued

Formatting a New Diskette (cont)

STEP *(cont)*

2) Press **F8**, (FORMAT) or type the word **FORMAT**. Then type in the characters **/V.**

The system displays the command **"FORMAT/V"** on the command line and waits for you to enter the letter that identifies the disk drive. Note that the command labels have changed on soft keys (see Fig. 2-1).

The **/V** is a parameter, or value that tells DOS how to use a command. In this case, **/V** instructs DOS to ask you for a volume name (or diskname) for the diskette after it has been formatted (see Steps 5 and 6).

STEP

3) Press **F7** (B:), or type **B** followed by a colon (:).

The system displays "B:" after the FORMAT command on the command line. This tells the system that the diskette you are going to format is going to be in drive B.

STEP

4) Press the **Return** key.

The system accepts the command and prompts:

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

continued

Formatting a New Diskette (cont)

STEP *(cont)*

5) Insert the a new, blank diskette into drive B. Strike any key when you're ready. (For a single-drive system, replace the master DOS diskette in drive A with the new diskette.)

   The drive now proceeds to whirr for about one minute. DOS displays:

   ```
   Formatting cylinder: 000
   ```

   The number increases as DOS formats the diskette. When finished, DOS displays:

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

(Note: "Enter" is synonymous with "Return".)

6) Type in a name for the disk: **NEWDISKETTE** and press **Return**. DOS displays:

   ```
   xxxxxx bytes total disk space
   xxxxxx bytes available on disk
   ```

   DOS is telling you how many bytes (character spaces) are available on the new diskette. DOS now displays the following:

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

7) Type in **n** and press **Return**.

...continued

Formatting a New Diskette (cont)

You are returned to the DOS system and the DISKS soft key command line is redisplayed.

STEP *(cont)*

8) Type in the command **VOL** followed by the characters **B:** and press **Return**.

...continued

DOS now displays the volume name (diskname): NEW-DISKETTE. Your new diskette is now formatted and can be used by DOS, and by other programs, to store information. It is recommended that you provide a diskname for all your diskettes to help you identify them. For more information as well as a description of the VOL and DISKNAME commands, consult the *Hyperion DOS 2.11 Guide.*

2.3 COPYING A DISKETTE

Before continuing, you should copy the files from the write-protected master DOS diskette to a new user diskette. Your master DOS diskette should be in drive A.

If your system has two drives, follow steps 9 to 13. If your system has a single drive go to step 14.

Copying a Diskette (cont)

For Double-Drive Hyperions

STEP *(cont)*

9) Enter the word **DISKCOPY** and press **Return** or, from the DISKS soft key line, press **F6** (DCOPY). DOS prompts you with:

```
Insert source diskette in drive A:
Insert target diskette in drive B:
Strike a key when ready ....
```

10) Press any key on the keyboard to start copying.

...continued

The system prompts you to insert the diskette you wish to copy from (the *source* diskette), and the diskette you wish to copy to (the *target* diskette):

The Hyperion's disk drives begin to whirr as the information from the master DOS diskette is copied and this message appears on your screen:

```
Copying 2 side(s), 9 sectors per track
```

When all the information has been copied onto the target diskette, the system prompts:

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

Copying a Diskette (cont)

STEP *(cont)*

11) When the system prompts:

Copy another (Y/N)?—

type in **N** and press **Return** to stop copying. The system prompt appears.

You now have a copy of the master DOS diskette which is not write protected and may be used to store your own files.

Your new copy should be used instead of the original master DOS diskette. The original should be stored in a safe place and used to make copies when needed.

STEP

12) Remove the master DOS diskette from drive A: and replace with your copy from drive B:.

13) Go to step 21 to continue.

...continued

For Single-Drive Hyperions

If your Hyperion has a single diskette drive, from time to time you must exchange the master DOS diskette (the "source" diskette) with the new diskette (the "target" diskette). The system prompts you each time this exchange is necessary.

STEP *(cont)*

14) From the DISKS soft key line, press **F6** (D-COPY), or enter the word **DISKCOPY.** The system prompts:

    Insert source diskette in drive A:
    and strike any key when ready

15) Replace the newly formatted diskette in drive A with the master DOS diskette.

16) Press any key to continue.

...continued

When as much information as possible has been transferred to the Hyperion's internal memory, the system prompts:

Insert target diskette in drive B:
and strike any key when ready

Note that although there may be only one disk drive the Hyperion uses a different disk drive label (B:) when writing than when reading information.

For Single-Drive Hyperions (cont)

STEP *(cont)*

17) Remove the master DOS diskette from the drive and insert the diskette you formatted in steps 1 to 8 of this lesson.

18) Press any key on the keyboard to continue copying.

...continued

When all information from the source disk has been copied to the target disk, the system prompts:

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

However it is unlikely that all the information could fit into the internal memory at one time. Therefore, *there is still more information to be copied.* In this case, the system repeats the prompt:

```
Insert source diskette in drive A:
and strike any key when ready
```

For Single-Drive Hyperions (cont)

STEP *(cont)*

19) If there is more information to be copied, repeat Steps 15 through 18 until the system indicates that the copying is complete, by prompting:

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

20) Press **N** to stop copying. The system prompt appears.

...continued

You now have a copy of the master DOS diskette, in drive A. Store the original master DOS diskette in a safe place and use it only to produce more copies when needed.

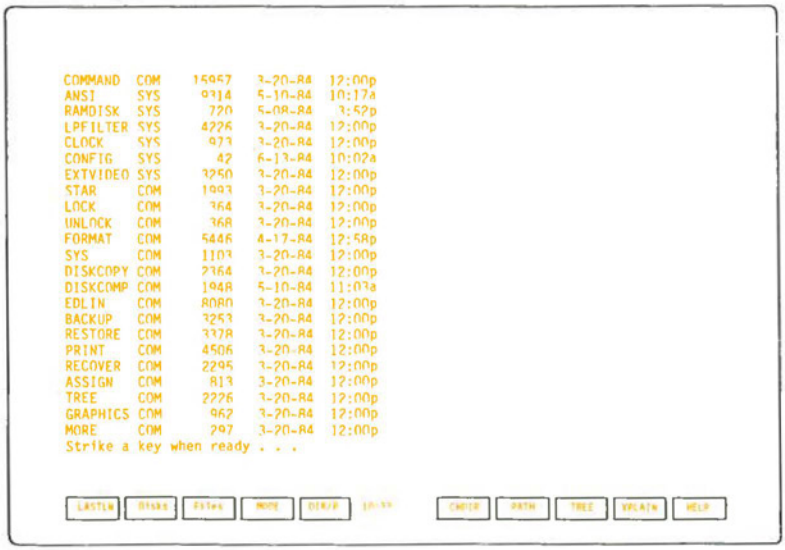


Fig. 2-2: The list of files on the DOS diskette as produced by the DIR command.

2.4 LISTING ALL THE FILES ON YOUR DISKETTE

It is very easy to list the names of files stored on any diskette. The DIR (directory) command has been provided for this purpose.

STEP *(cont)*

21) Press **F5** (DIR/P), or type in **DIR/P.** Then press the **Return** key.

A complete list of all files (as shown in Fig. 2-2) stored on the diskette in drive A is produced. The list is preceded by the volume (diskette) name, if it has one, and by the name of the directory being listed. The /P parameter on the command line tells the Hyperion to pause after each screenful. Press any key to continue the listing.

After the complete directory has been displayed, the system prompt reappears.

STEP

22) Press the **F5** soft key (DIR/P) or type **DIR/P.** Then type in the letter **A** followed by a colon (:) and press **Return.**

...continued

This has the same results as Step 21. The A: is redundant here since drive A is the default drive (you can tell this since the prompt reads "A>"). For more information on how DOS treats the default drive, see Section 1, page 13.

2.5 LISTING ONLY CERTAIN FILES ON YOUR DISKETTE

As you can see, there are many files on the master DOS diskette in drive A. You may even have trouble picking out a specific name among so many others. Fortunately, you can make the DIR command specific enough to produce a much smaller list.

STEP *(cont)*

23) Press **F1** (DOS), then **F1** (LASTLN).

Your last command line "DIR/P A: " is automatically retyped. If softkeys are not displayed, the last command line "DIR/P A:" must be typed in again.

24) Type **D*.COM** and press the **Return** key.

...continued

You have entered the command: "DIR/P D*.COM". The list of files produced this time is much shorter. This is because there are relatively few files on the DOS diskette that begin with the letter "D" and end with ".COM".

The asterisk (*) is a "wildcard" character that allows specification of a group of files instead of a single file. The command "DIR/P A:D*.COM" could be read as "list all files on the diskette in drive A that have filenames beginning with D and the filename extension .COM".

2.6 ENTERING COMMANDS

So far we have been using the FORMAT, COPY, and DIR commands. These are only three of the many commands DOS offers. DOS commands are instructions to the computer to perform certain actions, or start a particular procedure or program.

Entering Commands (cont)

You can add special values, or *parameters* to commands to tell DOS how you want the command to work. Parameters let you include or exclude certain command features, or alter the scope of a command's action.

Editing the Command Line

In most of the steps above, you entered a line of text: the command line. This line began with a command word, and sometimes included parameters.

With a long command line, it is possible that you can make an error while entering the command. DOS allows you to edit the command line, so that you can correct mistakes or change parameters.

You can have the last command you entered redisplayed on the command line by pressing F1 (LASTLN) on the DOS soft key line. LASTLN does not include the "press Return" action, so you can edit the command line, adding or changing parameters, or correcting typing mistakes. You can use the cursor left (←) and cursor right (→) keys as well as the **Rub Out**, **Ins** (for inserting) and **Del** (for deleting) keys.

If you enter a command with typing mistakes, DOS tells you:

```
Bad command or filename
```

Should this message appear, use LASTLN to redisplay the command, and make the necessary corrections.

2.7 SORTING A LIST OF FILES ON A DISKETTE

STEP *(cont)*

25) Press **F5** (DIR/P) or type in **DIR**

26) Press the **Rub Out** key 3 times to erase the "/P" on the command line if you pressed the **F5** soft key instead of typing in DIR.

27) Press **F5** (¦SORT) or type in **¦SORT**. Then press the **Return** key.

...continued

You have entered the command "DIR ¦SORT". The list of files produced by the DIR command is sorted in alphabetical order by the SORT command.

The "¦" symbol between the "DIR " and "SORT" commands is called a "pipe." DOS uses pipes to send the results of one command to another command for further processing.

2.8 COPYING A FILE

It is often useful to make copies of a file. Often, the copies are made onto another diskette for backup or archival purposes.

Copying a File (cont)

STEP *(cont)*

28) Press **F3** (Files) to access the FILES soft key line. (If the soft key line is not displayed, skip this step).

29) Press the **F6** soft key (COPY) or type the word **COPY**.

The soft key line labels, if any, change.

30) Press **F7**, now labelled "B:", or type **B** followed by a colon (**:**).

31) Type the filename **DIR.EXP**.

32) Press the spacebar once. *You must leave at least one space between the two filenames.*

33) Press **F7**, still labelled "B:", or type **B** followed by a colon (**:**).

34) Enter the file name **DIR.NEW**.

Your command line should now read:

```
A > COPY B:DIR.EXP B:DIR.NEW
```

35) Check that the Supplemental Programs diskette is in drive B, and then press the **Return** key.

...continued

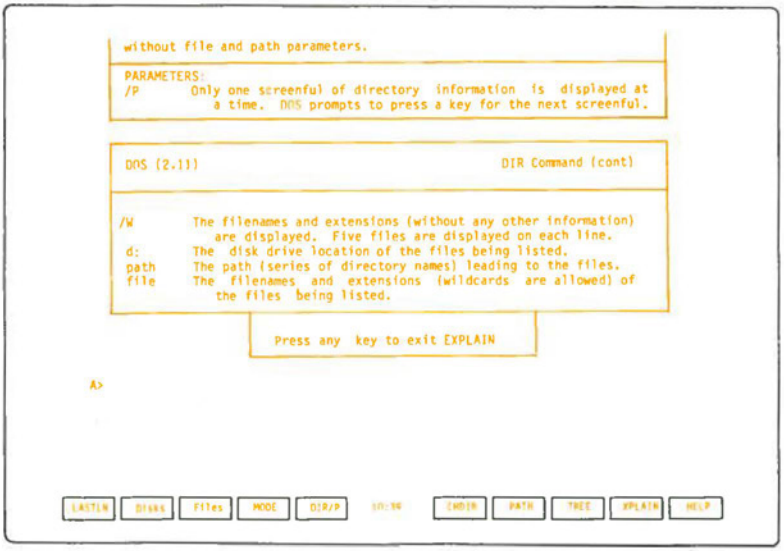


Fig. 2-3: Your document on the screen.

Copying a File (cont)

After a few seconds of disk action, the following messages is displayed, and the system prompt **A>** reappears:

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

A copy of the file called DIR.EXP is created under the filename of DIR.NEW. Both files are stored on the diskette in drive B. You can check that the COPY command was successful by using the DIR/P command.

STEP *(cont)*

36) Press **F5** (DIR/P), or type the word **DIR/P** followed by a blank space.

37) Type in **B:DIR*.*** and press **Return**.

2.9 DISPLAYING A FILE ON THE SCREEN

You can examine a file by using an editing package such as EDLIN or the Hyperion Text Editor. However, DOS makes it possible simply to view the contents of any diskette file on the Hyperion screen. You cannot edit such a display, but if you just want to look at a file, using DOS is faster than using an editing package.

STEP

38) If soft keys are displayed, press **F3** (Files).

...continued

Displaying a File on the Screen (cont)

The soft key line changes to display a set of labels that offer several DOS file management commands.

STEP *(cont)*

39) Press the **TYPE** soft key (F3), or type in **TYPE**. The command is displayed and the cursor waits for you to enter a parameter.

40) Type **B:DIR.NEW**.

Did you notice that the contents of the file DIR.NEW were displayed on the screen so quickly that you couldn't read it all? You can stop the display by pressing the **Ctrl** and **Num Lock** keys at the same time.

STEP

41) Press the **F1** (Dos) to return to the DOS soft key line, then press **F1** (LASTLN) and finally press **Return**. Alternatively, type in **TYPE B:DIR.NEW** again and press **Return**.

The file DIR.NEW is displayed.

42) To stop the display, press the **Ctrl** and **Num Lock** keys at the same time. The display should stop. When you have finished reading the display, press any character key to start the display scrolling again. You can "freeze" any scrolling display by pressing **Ctrl** and **Num Lock**, and continue the scrolling by pressing any character key.

...continued

Displaying a File on the Screen (cont)

As an alternative to freezing the scrolling display to read it, you can have DOS display the contents of a file one screenful at a time.

STEP

43) Type in the word **MORE** followed by a less than sign (<) and **B:** (the drive where the file is located), and the name of the file, **DIR.NEW**.

DOS displays a screenful of text. When you are ready for the next screenful:

44) Press any key on the keyboard.

Then press **Return**.

...continued

The display of DIR.NEW pauses after each screenful and DOS displays the prompt **--MORE --** at the bottom of the screen. The next screenful is displayed after you press a key. When the end of the file is reached, pressing any key redisplays the system prompt **A>**.

2.10 RENAMING A FILE

STEP *(cont)*

45) Make sure you are on the FILES soft key line (press the Files soft key to get there if you are not) and then press **F9** (RENAME), or type the word **RENAME** followed by a blank space.

46) Type in **B:DIR.EXP** to identify the file to be renamed and press the spacebar. There must be at least one space between the filenames.

47) Type in **DNEW.DIR** to assign this new name to the file, and then press **Return**.

As soon as the system prompt **A>** is redisplayed, look at the names of your files by entering the DIR/P command again.

STEP

48) Press **F5** (DIR/P), or type the word **DIR/P** followed by a blank space.

49) Press **F7** (B:), or type an **B** followed by a colon (:).

50) Type **D*.*** and press **Return**.

...continued

2.11 ERASING A FILE

Your two files, both copies of DIR.EXP, have been created only to show how the COPY and RENAME commands work. You should now erase these files, since they are unnecessary.

STEP

51) Press the **ERASE** soft key (F8).

52) Type **B:DIR.NEW** to identify the file to be erased. Then press the **Return** key.

53) Repeat steps 51 and 52 for the file DNEW.DIR.

If you enter the command "DIR/P B:" again, these two files are not listed. They have been erased from the diskette.

2.12 SUMMARY OF CONCEPTS

In this lesson, you have learned how to use some of the Disk Operating System (DOS) disk and file management commands to:

— format a new diskette,
— copy the contents of one diskette onto another,
— list files on a diskette,
— sort a list of files
— copy a file
— display the contents of a file on screen,
— rename a file,
— erase a file.

Summary of Concepts (cont)

You have also been introduced to the following features:

— **The Asterisk (*) Wildcard** to replace characters when searching for files on a diskette.

— **Filename.** Each file stored on a diskette must have a name.

— **The LASTLN Instruction** which is used to recall the last command line entered.

— **The FORMAT, DISKCOPY, DIR/P, SORT, COPY, TYPE, MORE, RENAME** and **ERASE** commands.

You have so far used only some of the DOS commands that are available. More DOS commands are introduced in Lessons 4, 5, and 6. All the DOS commands are described in detail in Part II of the *Hyperion DOS 2.11 Guide*. A synopsis of each command, including the command format; can be viewed by using the EXPLAIN soft key followed by the command name.

Lesson 3

USING EDLIN

Lesson 3

USING EDLIN

3.1 INTRODUCTION

Much of the work done on the Hyperion involves the entering and storing of information. This information is stored in files, each with a specific name, the file name. A file can contain information for programs, memos, letters, reports, etc. More information on files, including how to name them and organize them, is found in the next lesson of this tutorial.

The EDLIN single-line text editor has been provided to allow you to create files. An optional software package, Hyperion Text Editor, is also available. Hyperion Text Editor is a full screen editor, displaying text one screen (23 lines) at a time, allowing you to perform most word processing tasks in an easy-to-understand, straightforward manner. Both Hyperion Text Editor and EDLIN use the the ASCII character set, with each character being represented by a one-byte character code.

Before you start this lesson, you should have powered up your Hyperion as outlined in Lesson 1 (steps 1 to 7). Your DOS system prompt should read "**A>**". If it does not, type in the letter **A** followed by a colon (:) and press **Return**. The copy of your master DOS diskette made in Lesson 2 should be in drive A.

3.2 ACCESSING EDLIN AND CREATING A NEW FILE

This lesson shows you how to access EDLIN and create a file. The file you are about to create can later be stored on disk(ette), or not. Where the file is stored depends on the *drive specification* that you are instructed to enter before the filename at the end of this section.

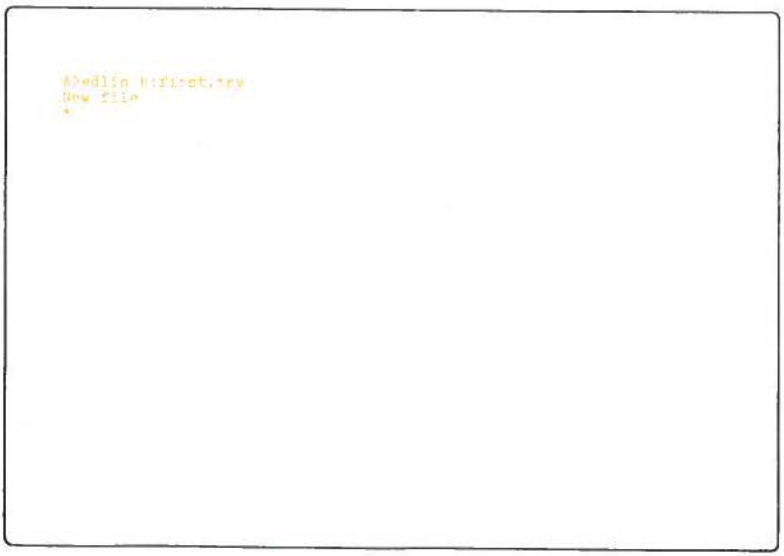


Fig. 3-1: The EDLIN screen format.

Accessing EDLIN and Creating a new File (cont)

The drive specification is a letter, followed by a greater than sign (>). This letter can be: A, for drive A; B, for drive B. When used on systems with one disk drive, DOS considers drive B to be in the same location as drive A. DOS usually tells you when it is considering the single drive as drive A or as drive B.

STEP

1) Type in **EDLIN FIRST.TRY**. Your command line should read:

   A>EDLIN FIRST.TRY

   "FIRST.TRY" is the name of the new file you are creating.

2) Press the **Return** key.

...continued

The screen changes to display the statements shown in Fig. 3-1. The flashing cursor at the right of the asterisk is the EDLIN system prompt. It indicates the system is ready for you to enter a command.

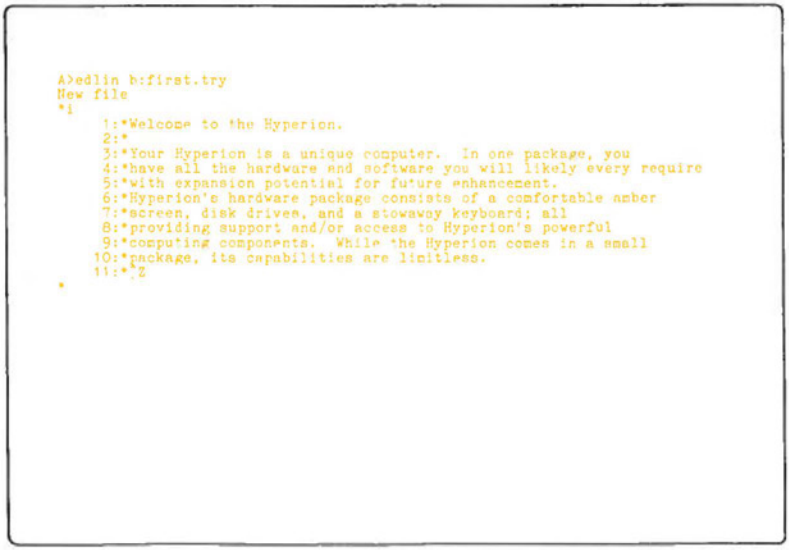


Fig. 3-2: Text, as it looks entered onto the Hyperion screen using EDLIN.

3.3 ENTERING TEXT

The Insert Mode

STEP *(cont)*

3) Type in the letter **I**. Press the **Return** key.

Entering I (the Insert command) puts you in the *insert mode*, allowing you to insert characters in a line of the file. Notice that the characters "1:*" is displayed, with the cursor just to the right of the asterisk. The number identifies the current line of the file. A particular line can be accessed by typing in the line number.

STEP

4) Type the text shown in Fig. 3-2. Press the **Return** key each time you come to the end of a line.

...continued

Notice that each time you press the **Return** key a new line number appears and the cursor waits to the right of the asterisk prompt.

Do not worry about mistakes. These can be edited out using EDLIN features described on the following pages. At any rate, pressing the **Rub Out** key will delete over any mistake.

Using "**i**" on a file that has already been created inserts newly-entered lines at the beginning of the file.

The EDLIN Command Line

STEP *(cont)*

5) Once you have typed the last line of text, and pressed the **Return** key, hold the **Ctrl** key down and press **Z**. Then press the **Return** key.

OR

Press **F6**, followed by the **Return** key.

The cursor advances to a new line, moves to the extreme left of the page, prints an asterisk, and puts the flashing cursor immediately to the right of the asterisk.

...continued

You are now no longer in the insert mode, and EDLIN is prompting you to enter a new command. This is the EDLIN command mode.

Pressing **Ctrl** + **Z** exits from the insert mode and moves you back into the EDLIN command mode. Pressing **Ctrl** + **C** at any time while on a particular line cancels the entry for that line and returns you to the EDLIN command mode.

3.4 DISPLAYING TEXT

Using EDLIN you can display 23 lines of a text file, a single line of text, several lines of text together, or the contents of a file on a line-by-line basis.

Displaying the Whole File

You may display 23 lines of a text file using the L (for LIST) command.

STEP *(cont)*

6) Type the letter **L** and press the **Return** key.

   23 lines of the specified file are displayed, with the line number of each new line, and the cursor indicating that a new command can be entered.

Displaying a Line

You may display a single line by first entering the line number then pressing the return key.

STEP

7) Type the number **3** and press the **Return** key.

   The third line of text is displayed.

8) Press the **Return** key.

   This returns you to the EDLIN command mode.

...continued

You may display a single line by first entering the number of any line that exists in your file.

Displaying a Line (cont)

STEP *(cont)*

9) Type any number and press the **Return** key.

10) Press the **Return** key.

11) Type another number, and press the **Return** key.

    Another line of text is displayed.

12) Press the **Return** key. This returns you to the EDLIN command mode.

Displaying a Group of Lines

STEP

13) Type the command **3,6 L** and press the **Return** key.

    Lines 3 to 6 are displayed.

14) Type other line numbers, and press the **Return** key.

    Another group of lines is displayed. An asterisk before the first word indicates that this is the "current line".

...continued

Displaying Lines Line-By-Line

STEP *(cont)*

15) Press the **Return** key.

Notice that the next consecutive line number is displayed, with an asterisk indicating the current line.

The first time you pressed the **Return** key, the cursor returned to the EDLIN command mode. The second time you pressed the return key, the next line of the file is displayed. Notice the asterisk has moved to this next line.

16) Press the **Return** key six times.

Notice that, for every second time you press **Return**, a new line of the file is displayed. In this way you can step through the file displaying its contents line-by-line.

17) Press the **Return** key.

This returns you to the EDLIN command mode.

...continued

3.5 REPLACING TEXT

If you wish to change the contents of a file (edit), you can do it in several ways. One way is to replace characters that you know are there with other characters.

STEP *(cont)*

18) Using the letter "r" to indicate replace, and the letter "o" to indicate the letter to be replaced, enter **1,3 ro** but don't press the **Return** key.

19) Hold down the **Ctrl** key and press the letter **Z**, or press soft key **F6**. (Using the **F6** soft key enters the **Ctrl** + **Z** combination.)

The system displays this combination as "^Z".

20) Type **a** and press the **Return** key. "a" indicates the letter that replaces "o".

...continued

The first three lines of the file are displayed as many times as there were changes made to each line. Notice that every incidence of "o" has been replaced with "a".

The command you have just entered has replaced all the o's with a's in the first three lines of the file. If you make a mistake when entering commands, you can use the **Rub Out** key to backspace over the mistake(s).

The interline commands available for text editing apply to entering commands, also. For example, if you want to repeat the command you just entered, press **F3**, and the last command (not the last edit line) is copied.

Replacing Text (cont)

STEP *(cont)*

21) Type the command **1,3 ra(Ctrl + Z)o**, your command line should read:

```
*1,3ra^Zo
```

Be careful not to put a space after the "r". If you do, the system searches for that space as well as the letters which follow it.

STEP

22) Press the **Return** key, the first three lines are redisplayed.

...continued

This time, the command has caused all occurrences of "*a*" to be replaced with "*o*". The text is still incorrect, but another part of the tutorial shows you how to correct this.

3.6 SEARCHING FOR TEXT

To search for a character string, use the letter "s".

STEP *(cont)*

23) Type the command **1,3 sHyp**

...continued

The system searches the first three lines of the file for the character string *"Hyp"*, and displays the first line in which it is found.

3.7 EDITING A LINE OF TEXT

The next thing to learn is how to edit a line of the file.

STEP *(cont)*

24) Display the first six lines of the file by typing the command **1,6 L** and pressing the **Return** key.

25) You are going to correct line 3. Type **3** and press the **Return** key.

Line 3 is displayed.

...continued

Moving through a Line of Text

There are three function keys that are used to correct a line, once the line is displayed by following step 25:

— the **F1** soft key or the *arrow* key (→) on the cursor/number keypad displays the current line character-by-character;

— the **F2** soft key displays the current line up to a certain character;

— the **F3** soft key displays the rest of the current line.

Moving Through a Line of Text (cont)

STEP *(cont)*

26) Press **F1** or the **arrow** key (→) five times.

The first five characters in line 3 are displayed on the lower or edit line. The upper line is the template line.

27) Press **F2** and type the letter **t**.

The edit line is displayed up to, but not including, the character "t".

28) Press **F2** and type the letter **k**.

The edit line is displayed up to, but not including, the character "k".

29) Press **F3**.

The rest of the line is displayed.

...continued

Storing the Changed Line

When editing a line, you must display the whole line before returning to the EDLIN command mode. *Only that part of the line that has been displayed is stored by the system.* Any portions that have not been displayed are lost.

Storing the Changed Line (cont)

STEP *(cont)*

30) Press the **Return** key to return to the command line.

31) Type the number **3** and press the **Return** key.

Line 3 is redisplayed.

32) Press **F2** and type the letter **q**.

The line is redisplayed up to the character "q".

33) Press the **Return** key.

The cursor returns to the EDLIN command mode.

34) Type the number **3** and press the **Return** key.

Line 3 is redisplayed. However, note that the line has been truncated. All text to the right of the original "q", including the "q", has been erased. Only that part of the line that was originally displayed, has been stored.

35) Press **F3** to display the existing line.

36) Using the keyboard keys, retype the text that was erased, correcting "pockoge" as you proceed. Press the **Return** key.

...continued

STEP *(cont)*

37) Type the number **3** and press the **Return** key.

Notice that the line 3 now reads as it did before.

Overwriting

One of the standard editing features of most systems is the ability to overwrite existing text. This means that entering a character from the keyboard overwrites (i.e., erase and replace) the current character in the file.

STEP

38) Press **F2** and enter **n**. Repeat this step.

39) Press the **Rub Out** key or the cursor pad **arrow key** (←) from the right side of the word three times, to remove the characters. Type the character string **a u**.

40) Press **F3** to display the rest of the line.

41) Press the **Return** key to store the changes, then enter 3 and press **Return** to redisplay the line.

Notice that the new line contains the corrected character string *"a u"* which overwrote the original string *"o u"*.

...continued

Inserting and Deleting Characters

When inserting characters between existing characters on a line, you must first insert the appropriate number of blank spaces. Pressing the **INS** key allows you to insert spaces into a line. Pressing the **INS** key again turns off this function.

STEP *(cont)*

42) Press **F2** and type the character **q**.

43) Press the **INS** key.

Notice the cursor does not move.

44) Type the word **imi**, and press **INS**.

45) Press **F3**.

Remember, you *must* display the rest of the line before returning to EDLIN, or the rest of the line is erased.

46) Press the **Return** key; then type **3** and press the **Return** key to redisplay the line.

Notice that the line displayed now reads "uni-imique", instead of "unique".

47) Press **F2** and type **i**.

48) Press **F1** until "**imi**" is displayed.

...continued

Inserting and Deleting Characters (cont)

STEP *(cont)*

49) Move the cursor (←) back over the letters imi, and press **DEL** 3 times to delete them.

50) Press **F3** to display the rest of the line.

51) Press the **Return** key to store the change.

52) Type **3** and press the **Return** key to redisplay the line.

...continued

You have deleted the three characters "*imi*" from the line.

3.8 STORING THE FILE

To store the contents of a file, you must first be in the EDLIN command mode:

STEP *(cont)*

53) Press the **Return** key.

54) To end the editing session, type the character **e** (for end) and press the **Return** key.

This command stores the file FIRST.TRY in memory, onto drive A. If you do not want to store the file permanently, delete it using the ERASE command from the floppy diskette.

3.9 SUMMARY OF CONCEPTS

In this lesson you have learned how to:

— access the EDLIN text editor,
— enter text into the system,
— display lines of text from the file being edited,
— search for character strings within the file,
— replace character strings in a file,
— edit a line of text,
— store the information and exit from EDLIN.

You have been introduced to the following EDLIN features:

— ***Using the F1 Key or the ARROW Key*** (→) from the cursor-number keypad to display a character in a specific line of a file.

...continued

Summary of Concepts (cont)

— **Using the F2 Key** plus a character.

— **Using the F3 Key** to display the rest of a line in a file.

— **Using the F6 Key** to enter the **Ctrl** + **Z** key combination.

— **Using the INS Key** to insert spaces into a line.

— **Using the DEL Key** to delete characters on a line.

— **The EDLIN Command Mode.**

— **The EDLIN Asterisk Prompt.**

— **The Numbering of lines in an EDLIN File.**

— **THE S (SEARCH) Command.**

— **THE R (REPLACE) Command.**

— **THE L (LIST) Command.**

These are only a few of the editing commands available in EDLIN. Part III of the *Hyperion DOS 2.11 Guide* describes the EDLIN system in detail, including all the EDLIN commands.

Lesson 4

ORGANIZING FILES

Lesson 4

ORGANIZING FILES

4.1 INTRODUCTION

In the previous lesson, you created a file using EDLIN. Keeping track of files is critically important, since all the information for your system is stored in files. Not only does DOS need its system files to execute commands, but most DOS commands involve the use of or manipulation of files. This lesson discusses the DOS commands and features used to help organize files.

In order to follow the steps in this lesson, you must have a blank diskette that has already been formatted using DOS. Your Hyperion must have been started as outlined in steps 1 to 7 in Lesson 1. Your system prompt should read "A>". If it does not, type in the letter **A** followed by a colon (:) and press **Return**.

4.2 NAMING FILES

Every file must have a unique name, since DOS uses the file's name to find it. (Two or more files can have the same name however, if they are on different disks or in different directories.) The name given to a file cannot be longer than eight characters, and can only use certain characters. The legal characters which can be used as part of a filename are:

A - Z (both upper and lower case)
0 - 9
$ # & ! % ( ) - — ¶ ' "

Any other character is illegal, especially the period (.), comma (,) backslash (\) greater than sign (>), less than sign (<), vertical bar (|) and space (blank). These symbols have special meanings to DOS.

The following are all good filenames:

TESTFILE JANUARY LETTERS MYFILE

In addition to a filename, a file may have an optional *filename extension*. The extension must be preceded by a period (.), and cannot be more than three characters long. You can use filename extensions to add more information to the name of a file, and help organize your files. Certain files can be grouped together by giving them the same extension. For example consider the following files containing monthly correspondence:

LETTERS.JAN LETTERS.FEB LETTERS.MAR

DOS reserves certain filename extensions for its own use. You should avoid using these extensions, except to match the meaning DOS gives them. Some of these special extensions are:

.BAK — backup file
.BAS — BASIC file
.BAT — batch file
.COM — command file
.EXE — executable file
.EXP — Explain text file
.LPT — printer control file

A filename should have meaning, so that the name reminds you of the contents of the file. Although it is possible to use **$$A4Q&.6** as a filename, it is difficult to remember what you put in that file.

The following filenames are much easier to understand, and give a clue to the file's contents.

BOB4.LET PAID.JAN DUE.FEB
TEST.BAT GAMES.BAS

4.3 SEARCHING FOR AND USING FILES

Once you have named your file, you can have DOS search for it so you can use information in the file. All you have to do is tell DOS the file's *full* name (filename and extension). If you do not provide the correct name, spelt exactly as you named it, however, DOS cannot find the file.

Wildcards

Sometimes you may want to have DOS perform the same task on a group of similiar files with similiar names. DOS provides a short-cut so you do not have to re-enter the command for each file.

This is done by using two special characters as *wildcards* in filenames, to represent either many characters or any one character. These two wildcards are:

* represents any group of characters

? represents a single character

The wildcards allow you to address and categorize files without repeated command entry.

When the question mark (?) is used in a filename or extension, DOS looks for all files that match all the other characters in the filename, but have *any character* in the position occupied by the question mark.

When the asterisk (*) is used, DOS looks for all files that match the other characters in the filename and from *0 to 8 characters* starting in the position where the asterisk occurs. When the asterisk is in the filename, characters following the asterisk in the filename are ignored. When the asterisk is in the extension, any characters following the asterisk are ignored.

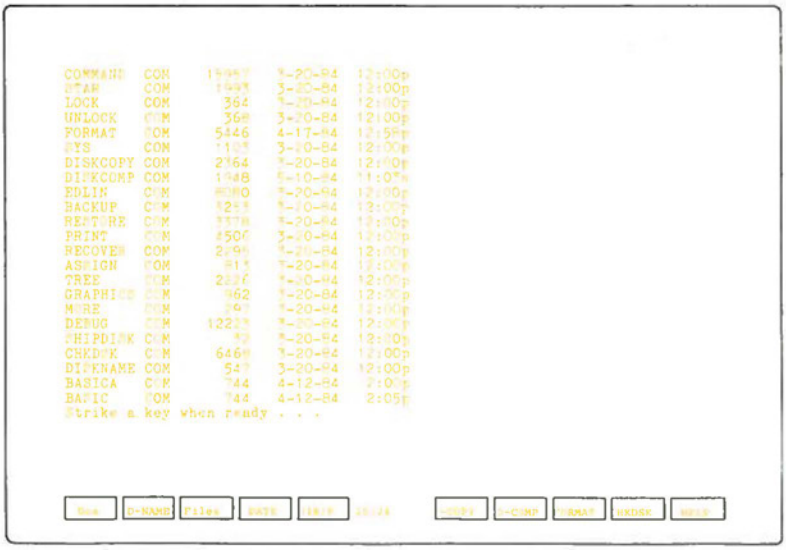


Fig 4-1: List of files produced by DIR/P *.COM

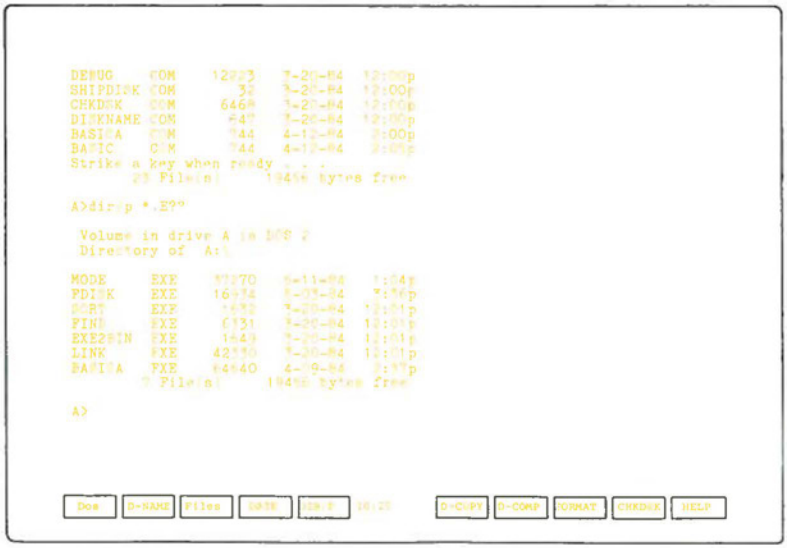


Fig 4-2: List of files produced by DIR/P *.E??

Searching For and Using Files (cont)

For example:

***.COM** means "all file[s] with the extension COM".

LET?? means "all file[s] whose name begins with LET and contains two additional characters.

The DIR command provides a good illustration of what the wildcards can do.

STEP

1) Press the soft key **F5** (DIR/P) or type in the command **DIR/P**.

2) Type in *.* and press **Return**.

   This command has the same effect as entering DIR/P without any parameters: all files on the diskette in A are displayed.

3) Press the soft key **F5** (DIR/P) or type in the command **DIR/P**.

4) Type in * **.COM** and press **Return**.

   This command displays all files with the extension .COM (the command files on the DOS diskette). See Fig. 4-1.

5) Press the soft key **F5** (DIR/P) or type in in the command **DIR/P**.

6) Type in ***.E??** and press **Return**.

   This displays files with extensions starting with E (e.g. EXE files). See Fig. 4-2.

...continued

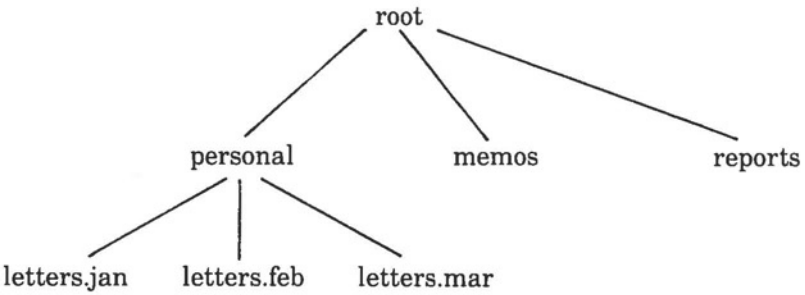


Fig. 4-3: Tree-Structured Directories. When the hierarchical organization of directories is outlined on a chart, the "branches" resemble an upside-down tree.

4.4 TREE-STRUCTURED DIRECTORIES

The old version of DOS had only one directory, with a limited number of files. DOS 2.11, however, can handle many more files, and has a different method of organizing and finding the files.

If DOS had to search a 10 megabyte hard disk in its entirety for one file, directory searches would take a long time. To avoid this, DOS sets up a *"root"* directory, from which you can create other subdirectories to hold different categories of files. Within each subdirectory you can continue to create more subdirectories.

Organizing your Files into Directories

Many users, especially those with a hard disk storage system, have hundreds and even thousands of different files to keep track of. You can organize your files and keep track of them by putting related groups of files into the same directory. The directories can be organized into a logical hierarchy, with a group of related subdirectories located in a *"parent"* directory.

For example, you may want to keep your private letters separated from your business memos and reports. A possible directory hierarchy is shown in Fig. 4-3.

Fig. 4-4: A listing of the files and directories produced by entering DIR/P B:\ (or DIR/P B: when the current directory is the root directory).

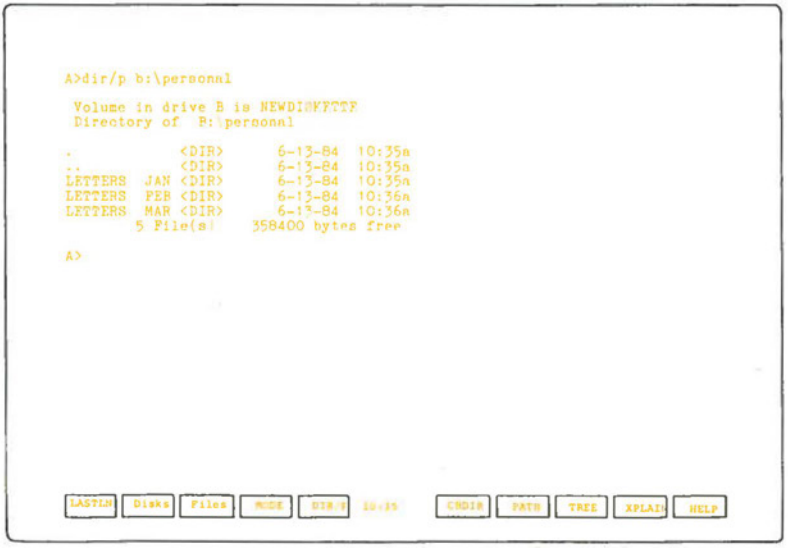


Fig. 4-5: A listing of the files and directories produced by entering DIR/P B:\PERSONAL.

4.5 CREATING SUBDIRECTORIES WITH MKDIR

DOS provides a special command for you to use to create subdirectories. This command is MKDIR (MaKe DIRectory). Let's create the subdirectory PERSONAL on drive B.

STEP*(cont)*

7) Type in the command **MKDIR B:\PERSONAL** and press **Return**.

MKDIR creates a subdirectory at the end of the chain of subdirectories you specify. To create a subdirectory in PERSONAL, you must specify both the disk drive (B) as well as the "parent" directory containing the new subdirectory. A subdirectory is referred to as a *"child"* of the parent directory.

Notice that we refer to "subdirectories" generally as "directories" , and only as "subdirectories "when we want to emphasize the fact that they are contained in another directory.

To create the subdirectories LETTERS.JAN, LETTERS.FEB and LETTERS.MAR in the directory PERSONAL:

STEP

8) Type in the command **MKDIR** followed by **B:\PERSONAL\LETTERS.JAN** and press **Return**.

9) Type in the command **MKDIR** followed by **B:\PERSONAL\LETTERS.FEB** and press **Return**.

10) Type in the command **MKDIR** followed by **B:\PERSONAL\LETTERS.MAR** and press **Return**.

...continued

Creating Subdirectories with MKDIR (cont)

You have created subdirectories LETTERS.JAN, LETTERS.FEB and LETTERS.MAR in the subdirectory PERSONAL. If you had tried to create any of these subdirectories without specifying the parent directory (PERSONAL), DOS would have sent you an error message **"Unable to create directory"**.

The names of the directories must follow the same conventions as for filenames and can use the same characters. The directory names can be up to eight characters long, and they can have directory extensions of up to three characters.

As with file names, you cannot have two subdirectories or a file and a subdirectory with the same name and extension in the parent directory. In addition, the directory name cannot match the name of a file in its parent directory.

STEP *(cont)*

11) Press the soft key **F5** (DIR/P) or type in the command **DIR/P**. Then type in **B:** and press **Return**.

    The screen display should look like Fig. 4-4.

12) Press the soft key **F5** (DIR/P) or type in the command **DIR/P**.

13) Type in the characters **B:\PERSONAL** and press **Return**.

...continued

The new display should look like that shown in Fig. 4-5. The notation <DIR> after an entry indicates that the named object is a subdirectory. Note the "." and ".." entries. These special objects are part of every directory you create. The "." entry tells DOS that this is a directory, and the ".." entry tells DOS where the parent (the next highest level) is.

4.6 CHANGING THE CURRENT DIRECTORY WITH CHDIR

When you entered DIR/P B:\ and DIR/P B:\PERSONAL, you were asking DOS for information from specific directories. If you let DOS "make the decision for you", the results are:

STEP *(cont)*

14) Type in the letter **B** followed by a colon (**:**) then press **Return**. The prompt reads "B>".

15) Press the **F5** soft key (DIR/P) or type in the command **DIR/P**. Then press **Return**.

This asks DOS for a listing of the contents of drive B. The results should be just like that shown for the command DIR/P B:\ (the listing of the root directory) in Fig. 4-4.

This is because the root directory is at present the *current directory*. The current directory is the only place DOS looks for files, unless you give it a path which shows other places to look.

If you use subdirectories extensively, and switch frequently from one directory to another, you may want DOS to prompt you with the path to the directory where you are currently located. You can do this with the PROMPT command.

STEP

16) Type in the word **PROMPT** followed by the characters **P_NG** and press **Return**.

...continued

The prompt displayed by DOS now changes to:

```
B:
B>
```

Changing the Current Directory with CHDIR (cont)

The top line tells you which directory you are in: the root directory (represented by the "\") of drive B. The next line is your normal DOS prompt, consisting of the drivespec (**B**) and a greater-than sign (>).

When you entered the command in Step 16, the **$P** tells DOS to display the current directory, the **$_** (underscore) tells DOS to go to the start of the next line (like the carriage return on a typewriter), the **$N** tells DOS to display the drivespec and the **$G** tells DOS to display the greater-than sign. You can customize the prompt to tell you the time of day or date or display any message you want. The PROMPT command is described fully in the *Hyperion DOS 2.11 Guide*.

STEP *(cont)*

17) Press the soft key F6 on the DOS soft key line **(CHDIR)** or type in the command **CHDIR**. Then type in \ **PERSONAL** and press **Return**.

    This makes PERSONAL the current directory. The prompt changes to:

```
B:\PERSONAL
B>
```

18) Type in **MKDIR LETTERS.APR** and press **Return**.

19) Press the soft key **F5** (DIR/P) or type in the command **DIR/P.** Then press **Return**.

...continued

Since the current directory is now PERSONAL, a directory listing displays only the files and subdirectories found in the PERSONAL directory.

Changing the Current Directory with CHDIR (cont)

Notice that in step 19 you did not have to provide a complete list of the directories from the root directory (B:\) to the directory you created (LETTERS.APR) as you did in steps 8 to 10. The list of directories is called a *path*. It provides a "pathway" from where DOS is currently looking (the current directory) to where you want DOS to go. Since the current directory is now PERSONAL, instead of the root directory, you do not need to include PERSONAL in the path.

STEP *(cont)*

20) Press the soft key F6 on the DOS soft key line (CHDIR) or type in the command **CHDIR**. Then type in **\PERSONAL\LETTERS.JAN** and press **Return**.

This makes PERSONAL\LETTERS.JAN the current directory. You do not have to enter PERSONAL as it is already the current directory and is taken for granted by DOS.

21) Press the soft key **F5** (DIR/P) or type in the command **DIR/P**. Then press **Return**.

Now, change the default drive to A: by entering **A:**.

22) Press the soft key F6 on the DOS soft key line (CHDIR) or type in the and **CHDIR**. Then type in **B:** and press **Return**.

DOS reports:

```
B:\PERSONAL\LETTERS.JAN
```

This is the current directory on drive B:.

...continued

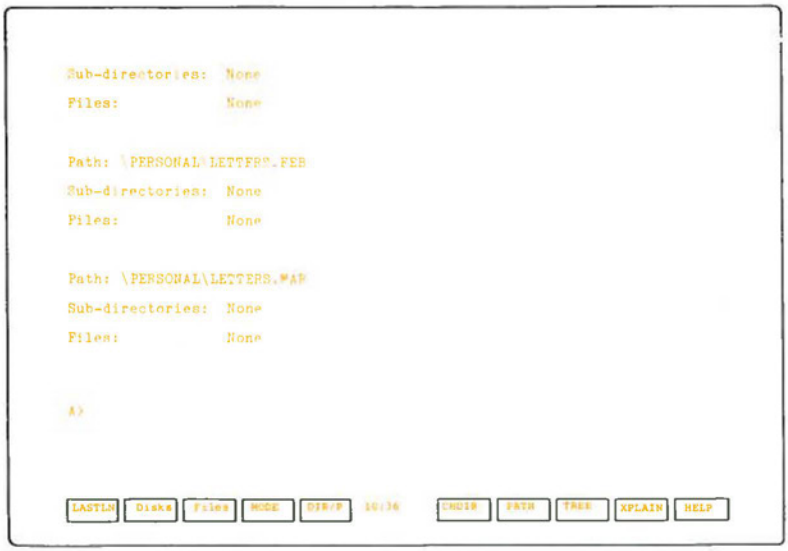


Fig. 4-6: The listing of directories (and the subdirectories they contain) generated using the TREE command with the /F parameter.

4.7 USING TREE TO LIST DIRECTORIES AND FILES

The TREE command causes DOS to display all the directories on a disk. This lets you see how the directories are organized, since all the *"child"* directories are listed for each directory.

STEP

23) Press the soft key **F8** (TREE) on the DOS soft key line, or type in the word **TREE**. Enter the drivespec **B:**, then press **Return**.

DOS now lists each directory and the subdirectories it contains on the screen.

You may also want to see which files are contained in the directories:

STEP

24) Press the soft key **F8** (TREE) on the DOS soft key line, or type in the word **TREE**. Then type in **/F**, the drivespec **B:** and press **Return**.

...continued

DOS now lists each directory, along with the subdirectories and files contained in it. The display should resemble that in Fig. 4-6.

4.8 USING THE PATH COMMAND

You already know that DOS searches the current directory if you don't specify a path before the command. You can use the PATH command to specify an alternate search path, which tells DOS where to look for a command, if it doesn't find it in the current directory.

STEP *(cont)*

25) Press the soft key **F7** (PATH) on the DOS soft key line, or type in the word **PATH**. Then type in **B:\PERSONAL\LETTERS.JAN** and press **Return**.

...continued

DOS now looks in the root directory for any commands which it doesn't find in the current directory.

You can specify as many alternate paths as you wish, by separating the paths by semicolons (;). For example

```
PATH  B:;B:\LETTERS;B:\REPORTS;A:\
```

(Don't use this example, as we have already entered a PATH command in step 25.)

Using this path, if DOS does not find the command in the current directory, it looks in the root directory of the diskette in drive B, then the subdirectories *letters* and *reports*, and finally in the root directory of the diskette in drive A.

Using the PATH Command (cont)

To determine what the current path is

STEP *(cont)*

26) Press the soft key **F7** (PATH) on the DOS soft key line, or type in the word **PATH**. Then press **Return**.

DOS replies with:

```
PATHB:\PERSONAL\LETTERS.JAN
```

PATH is used only for DOS to locate commands, programs and batch files when you enter their names. You cannot use it with a command to look for data files.

4.9 USING RMDIR TO REMOVE DIRECTORIES

Just as you can erase files from a diskette, you can also remove subdirectories. There are only two restrictions: the subdirectory must be empty, and must not be the current directory.

STEP

27) Type in the command **RMDIR** followed by **B:\PERSONAL\LETTERS.MAR**

...continued

This causes the directory PERSONAL\LETTERS.MAR to be removed from the diskette in drive B. You can use the DIR command to check that the directory was removed.

Using RMDIR to Remove Directories (cont)

If you had files in PERSONAL\LETTERS.MAR, and tried to RMDIR it, DOS would report an error to you.

Before you quit this section, you probably want to change the prompt back to the standard DOS prompt.

STEP *(cont)*

28) Type in the word **PROMPT** and press **Return**.
The standard DOS prompt is restored.

4.10 SUMMARY OF CONCEPTS

In this section you have learned how to name your files and organize them. We discussed:

— filenames and extensions
— permitted and restricted characters for filenames
— reserved extension names
— the ? and * wildcards

We also discussed tree-structured directories, how to use them, and the special DOS commands used to set up, and manage a hierarchy of directories. These commands are:

CHDIR — Change the current directory

MKDIR — Create a subdirectory

PATH — Specify a search path for commands

RMDIR — Delete a subdirectory

TREE — Display all directory paths

Lesson 5

REDIRECTING INPUT AND OUTPUT

Lesson 5

REDIRECTING INPUT AND OUTPUT

5.1 INTRODUCTION

So far in this tutorial booklet, we have been using DOS commands to perform fairly straightforward actions. These have usually been tasks that might be termed as *"system housekeeping"*, for example, providing lists of files on a diskette (DIR) erasing or copying files (ERASE and COPY) and formatting a diskette so you can use it (FORMAT).

All these commands have a direct, one-step action. In addition, they are all designed to be entered into the system from only one place (the keyboard) and they usually send any output to only one place (the screen).

Even though these commands are very useful, there are times when users familiar with the system would want them to be more flexible. For example, the DIR command provides a list of files on a disk, but if you want to work with those files you must key in their names manually.

As well, there are times when you might want to use a number of commands in a sequence, and use the results from one for the next. For example, to find certain text strings in a file, and then sort the lines in alphabetical order.

Also, there are some commands (the filter commands: MORE, FIND, SORT) that are really designed to take their input from somewhere other than the keyboard (although you can provide the input from the keyboard if you want).

This version of DOS provides a special feature, called the *redirection of standard input and output* to permit the kind of actions mentioned above.

This means that the input can be taken from somewhere other than the standard source (the keyboard), and the output can be sent somewhere other than the standard target (the screen). DOS calls the standard source the *standard input device* and the standard target the *standard output device.*

To follow the steps outlined in this lesson of the tutorial, the DOS prompt should read "**A>**". If the prompt is not of the form X>, where X is either A, B or C, type in the word **PROMPT** and press **Return**. This happens if you fail to perform the final step (step 28) outlined in Lesson 4.

If the prompt shows some other letter (B or C), type in the letter **A** followed by a colon (:) and press **Return**. Finally, be sure you have a blank, formatted diskette in drive B.

5.2 REDIRECTING OUTPUT

Normally DOS sends the output from a command directly to the screen, since this is where you can most conveniently see it. This has the drawback that you cannot save the output for later use. However, you can save the output to a file using the *standard redirection of output* feature.

An easy example is the redirection of output from the DIR command into a file, instead of sending it to the screen. The greater than symbol (>) is used to redirect the input.

STEP

1) Type in **DIR A: > B:DIRLIST** and press **Return**.

...continued

The top diskette drive starts whirring. When it stops, the DOS prompt reappears. Instead of being displayed on the screen, the output (the directory of A) has been sent to B:DIRLIST.

Redirecting Output (cont)

STEP *(cont)*

2) Type in **DIR B:DIR*.*.**

The system reports one file: DIRLIST. If you notice the time it was created, it is the current time. This file contains the listing generated by the DIR command.

STEP

3) Type in **TYPE B:DIRLIST.**

...continued

The contents of DIRLIST are displayed. In this case, it looks just like a directory listing.

The key in the command entered in Step 1 was the greater than symbol (>). This indicates that the output from a command is to be sent to the location specified after the symbol instead of the screen (the standard output device).

When you redirect the output into a file, DOS creates a file under the name you entered, and places the data there. If a file already exists under that name, the contents of the file are erased, and replaced by the incoming data.

Sometimes you may want to add the information to an existing file without destroying the information already in the file. To do so, two greater than symbols (>>) are used.

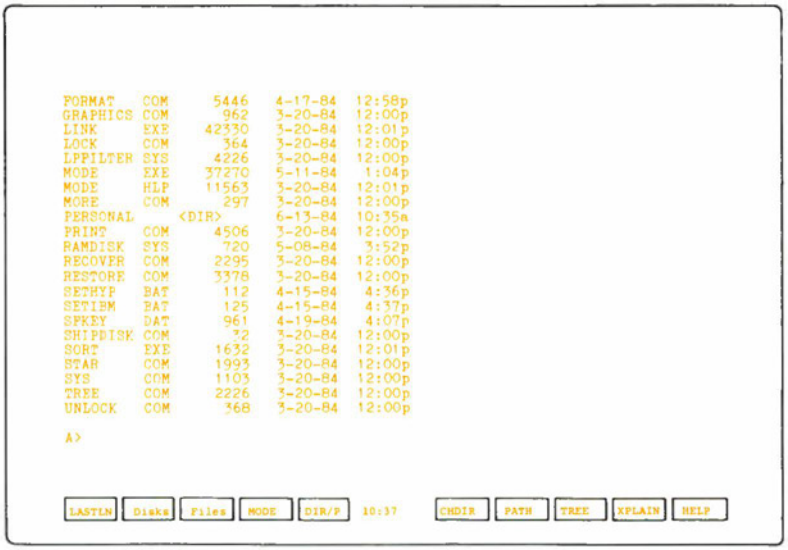


Fig. 5-1: The list of files on the diskette in drive A, sorted in alphabetical order. The list is produced by sorting the contents of a file (using SORT) containing a listing of the files on the diskette.

***Redirecting Output** (cont)*

STEP *(cont)*

4) Type in **DIR B: >> B:DIRLIST.**

5) When the DOS prompt reappears, type in **TYPE B:DIRLIST.**

The list of files on the diskette in drive A has been appended to the end of the file containing the list of files in drive B.

5.3 REDIRECTING INPUT

In the same way that DOS normally sends any output to the standard output device (screen) it normally takes any input from only one place: the keyboard. The drawback in this is that you cannot take the input from an existing file. The *standard redirection of input* feature overcomes this constraint.

In this example, we'll use the DOS command SORT, which sorts the lines in a file in alphabetical order. The less than symbol (<) is used to redirect the input.

STEP

6) Type in **SORT <B:DIRLIST.**

...continued

This command tells SORT to use B:DIRLIST as its standard input, and display the sorted lines on the screen (Fig. 5-1).

Redirecting Input (cont)

Not only can you redirect standard input for a DOS command or redirect the standard output, but you can also do both at the same time for the same command. This lets you handle whole files at a time, without any re-entry of the information.

STEP

7) Type in **SORT <B:DIRLIST > B:SORTLIST.**

   The input is taken from DIRLIST and the output is redirected into SORTLIST.

8) Type in **TYPE B:SORTLIST.**

The display is the same as that obtained by Step 6 (see Fig. 5-1).

5.4 FILTER COMMANDS

There are certain commands that almost always use the standard redirection of input and output feature. These commands alter the information received from the standard input device and send it to the standard output device.

These processes are called *filtering*, since they "filter" data in a similiar manner that a filter works on the liquid passing through it. The DOS commands involved are called *filter commands*.

Three filter commands are provided with DOS:

FIND Finds a given character string. As used previously, FIND locates the string, and displays the line in which it was found.

Filter Commands (cont)

SORT Arranges lines of text in ascending alphabetical order. In the example above, SORT rearranged the lines of text in DIRLIST in alphabetical order.

MORE Takes the input and displays it one screenful at a time, prompting for a keystroke before displaying the next screenful. You have already been introduced to the MORE command in Section 2.

5.5 PIPING DATA FROM ONE COMMAND TO ANOTHER

So far we have taken input from a source other than the standard input and sent output from a command to a destination other than the standard output. If fact we have done both these actions simultaneously. The next step is to take the output from one command and use it as the input for a *different command.* This passing of information from one command to another is called *piping.*

DOS creates a temporary file into which it sends the output from the first command. The contents of this file are then used as the input for the next command. This activity cannot be seen by the user (it is said to be *"transparent"* to the user). However, you can find the pipe file under certain circumstances.

STEP *(cont)*

9) Type in **DIR | SORT**

All the files on drive A are sorted in alphabetical order and displayed.

...continued

Piping Data from one Command to Another (cont)

DOS executed the DIR command, but instead of sending the output to the screen, placed it in the files **%PIPE1.$$$** and **%PIPE2.$$$**. DOS then executed the SORT command, and used the two new files as its input. The SORT command sent its output to the standard output device (the screen).

At the beginning of the directory listing, you may have noticed two files **%PIPE1.$$$** and **%PIPE2.$$$**. These are temporary pipe files, created by DOS to hold input/output so it could be redirected to another command.

All temporary pipe files are of the form **%PIPEx.$$$**, where x is the temporary file number. These files are erased after the input/output has been sent to its required destination. If you ask DOS for a directory listing for the files on drive A now, using just the DIR/P command, the %PIPE1.$$$ and %PIPE2.$$$ files are no longer listed.

You can link more than one filter command with pipes:

STEP *(cont)*

10) Type **TYPE B:DIRLIST ¦ SORT/+23 ¦ FIND "COM" ¦ MORE** and press Return.

...continued

Piping Data from one Command to Another (cont)

The file B:DIRLIST is sent to FIND, which selects the lines containing the letters COM. Those lines are then used with the SORT command, which arranges them in numerical order by date (the /+23 tells DOS to SORT on the twenty-third column, which is the year element of the date the files were last modified.) That sorted file is then used as input for MORE, which displays the lines in it one screenful at a time.

5.6 SUMMARY OF CONCEPTS

In this lesson, we have discussed the following concepts:

— the redirection of input

— the redirection of output

— the piping of the output data from one command to be used as input for another command

In addition, three special commands, called filter commands, have been discussed:

— the **FIND** command

— the **MORE** command

— the **SORT** command

Lesson 6

USING BATCH FILES

Lesson 6

USING BATCH FILES

6.1 INTRODUCTION

You've now spent some time learning to use DOS. This last lesson is designed to show you how to make DOS work for itself. Batch files are a convenient and powerful DOS feature which allows you to put a series of DOS commands together in one file, and execute them by entering only a filename.

There are two main advantages gained from using batch files: time is saved, as you don't have to wait to enter the command in a procedure; and commonly-used series of commands can always be correctly entered without typing errors.

Before you start this section, you should be familiar with the concepts described in the previous sections of this tutorial: DOS commands, parameters, and commands lines; paths and directories; as well as the commands used in previous lessons: FORMAT, COPY, DIR, etc.

Again, as with other lessons, you should have already started up your system and have a copy of the master DOS diskette in drive A, and the Supplemental Programs diskette in drive B. Your DOS prompt should read "**A>**".

6.2 CREATING A BATCH FILE

A batch file is made up of DOS commands: one command per line. There is no limit on the length of a batch file.

The name you assign to the batch file must be a legal filename: not longer than 8 characters and without any reserved characters. You must use the filename extension **.BAT**, as this is how DOS identifies batch files.

Fig. 6-1: Your batch file on the screen (just before you press Return to execute the batch file, as described in Step 4).

Furthermore, you can't use filenames which exist for command and executable files, i.e. COPY, DATE, FORMAT, etc. If you used one of these names, DOS would not know if you meant the command or the batch file. Unpredictable system behaviour may result.

Although we recommend that you use EDLIN or a text editor for creating complex batch files, or editing them to correct mistakes or to modify their actions, we are going to use a short cut in creating batch files here.

When you are creating short, simple batch files, you can enter the file directly on the screen using the COPY command. The format for this use of COPY is:

```
COPY CON d:filename.BAT
```

where *CON* stands for the CONsole (the screen and keyboard you are working at); *d:* specifies the drive where the file is to be stored; and *filename.BAT* is the name of the batch file.

STEP

1) Type in **COPY CON DIRALL.BAT** and press **Return.**

   Notice that we don't need to specify a drive since we are going to store the file on drive A, which is the current drive.

2) When the cursor moves to the line under the command you have just entered, type in the following:

   ```
   DIR A:
   DIR B:
   ```

   (press **Return** after each line, including the last one).

...continued

Creating a Batch File (cont)

STEP *(cont)*

3) Press **Ctrl** + **Z**, then **Return**.

4) When the A> prompt reappears, type in **DIRALL** and press **Return** (see Figure 6-1).

(Note that when executing a batch file, only the filename has to be entered. DOS assumes the .BAT extension.)

DOS now displays the first command in the batch file and executes it. After the list of all the files on the diskette in drive A have been displayed, the next command in the batch file is displayed and executed. This continues until all the commands in the batch file have been displayed and executed.

6.3 USING PARAMETERS IN BATCH FILES

Batch files can do more than just group DOS commands together for convenient execution. They can also be used to perform the same commands on different files, disk drives, etc. Like DOS commands, batch files can be used with parameters entered along with the batch filename on the command line. These parameters can then be used to modify the commands in the batch.

This is done by using *parameter variables* in the batch file. These parameters are represented by a percent sign (%) followed by a one-digit number. DOS matches the parameter variable number to the parameters listed on the command line following the name of batch file.

Using Parameters in Batch Files (cont)

This next example shows how you can use parameter variables in a batch file.

STEP *(cont)*

5) Type in **COPY CON DIRFIND.BAT** and press **Return.**

6) When the cursor moves to the line under the command you have just entered, type in the following:

```
DIR %1: |FIND %2
DIR %3: |FIND %4
```

(press **Return** after each line, including the last one).

7) Press **Ctrl** + **Z**, then **Return.**

8) When the A> prompt reappears, type in **DIRFIND A "COM" B "EXP"** and press **Return.**

...continued

DOS displays the first command in the batch file as DIR A: |FIND "COM" and executes it. Notice how the %1 and %2 have been replaced with the first two parameters listed on the command line after DIRFIND.

After the list of files containing "COM" has been displayed, DOS displays and executes the next command, replacing %3 and %4 with the next two parameters on the command line after DIRFIND.

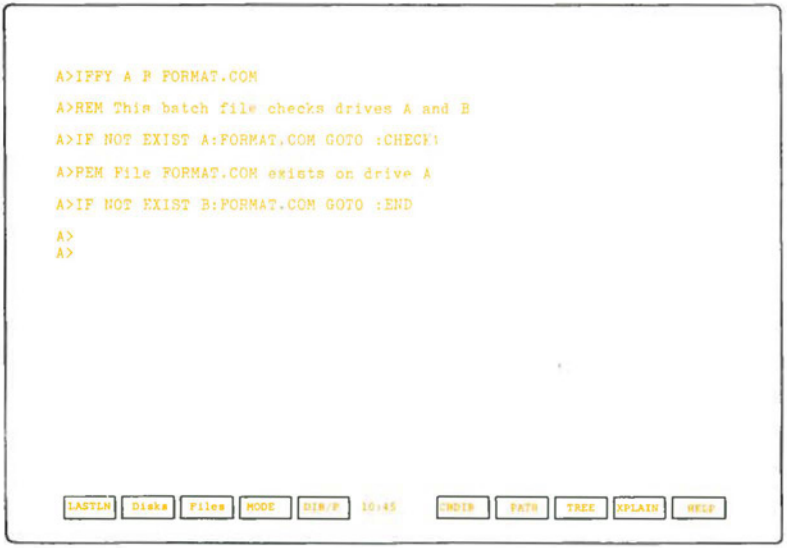


Fig. 6-2: The display produced by the batch file IFFY.BAT, demonstrating the GOTO, IF and REM subcommands.

6.4 BATCH FILE SUBCOMMANDS

To provide extra flexibility, this version of DOS has a number of special *subcommands* that you can use.They are called subcommands, since unlike ordinary DOS commands, they are designed to be used only in batch files.

In order to demonstrate the batch file subcommands they have been divided into four arbitrary groups: GOTO, IF and REM; ECHO and PAUSE; FOR; and SHIFT.

GOTO, IF, and REM:

To start, let's use the GOTO, IF and REM subcommands: Notice how the GOTO subcommand is executed only if the IF condition is true (i.e. the file does not exist). REM lets you insert remarks at any point in the batch file without affecting any of the DOS commands in the file.

STEP *(cont)*

9) Type in **COPY CON IFFY.BAT** and press **Return.**

10) When the cursor moves to the line under the command you have just entered, type in the following:

```
REM This batch file checks drives %1 and
  %2
IF NOT EXIST %1:%3 GOTO :CHECK1
REM File %3 exists on drive %1
:CHECK1
IF NOT EXIST %2:%3 GOTO :END
REM File %3 exists on drive %2
:END
```

(press **Return** after each line, including the last one).

STEP *(cont)*

11) Press **Ctrl** + **Z**, then **Return.**

12) When the A prompt reappears, type in **IFFY A B FORMAT.COM** and press **Return.**

...continued

The batch command files (and their output) should resemble Fig. 6-2.

The REM subcommand is one of the most straightforward. Use it to insert comments into the batch file. The comments are displayed when the REM subcommand is executed (unless the ECHO subcommand, described later in this section, is set to OFF). The comment following REM can be up to 123 characters long.

The GOTO subcommand transfers control to a line elsewhere in the program. When a GOTO subcommand is executed, DOS begins executing commands starting with the line after the line with the target label. The label (:START in the example below) must be the same in both the target line and the line with the GOTO statement. Maximum label length is 8 characters.

The IF subcommand lets you execute a DOS command (or subcommand) only under the conditions you specify. The most common conditions you can specify are whether or not a file exists, and whether or not two text strings are identical. These are the four possibilities:

IF EXIST file1	Statement is true if file exists;
IF NOT EXIST file1	Statement is true if the file does *not* exist;
IF string1 = = string2	Statement is true if string1 is identical to string2;
IF NOT string1 = = string2	Statement is true if string1 is *not* identical to string2.

(Another condition, ERRORLEVEL, can be used by certain programs. It is described in the *Hyperion DOS 2.11 Guide.*)

ECHO and PAUSE:

Now let's use ECHO and PAUSE. Although this file is designed to search for certain files on specified drives entered as parameters after the batch filename, the REM, ECHO, and PAUSE subcommands demonstrate the effects of ECHO. The PAUSE subcommand also halts processing so you can stop the continually repeating loop resulting from the GOTO subcommand. pause

STEP *(cont)*

13) Type in **COPY CON SUBCOM.BAT** and press **Return.**

14) When the cursor moves to the line under the command you have just entered, type in the following:

```
:START
REM This batch file looks for %2 on drive
  %1
ECHO OFF
REM This remark won't be displayed
ECHO But captions following ECHO are
  displayed
DIR %1:%2
PAUSE Do you want to stop? Press Ctrl + C
  to do so
ECHO
ECHO The PAUSE caption wasn't displayed
  with ECHO off
ECHO ON
PAUSE You have a second chance to stop.
  Press Ctrl + C
GOTO :START
```

(press **Return** after each line, including the last one).

15) Press **Ctrl** + **Z**, then **Return.**

16) When the A> prompt reappears, type in **SUBCOM A FORMAT.COM** and press **Return.**

...continued

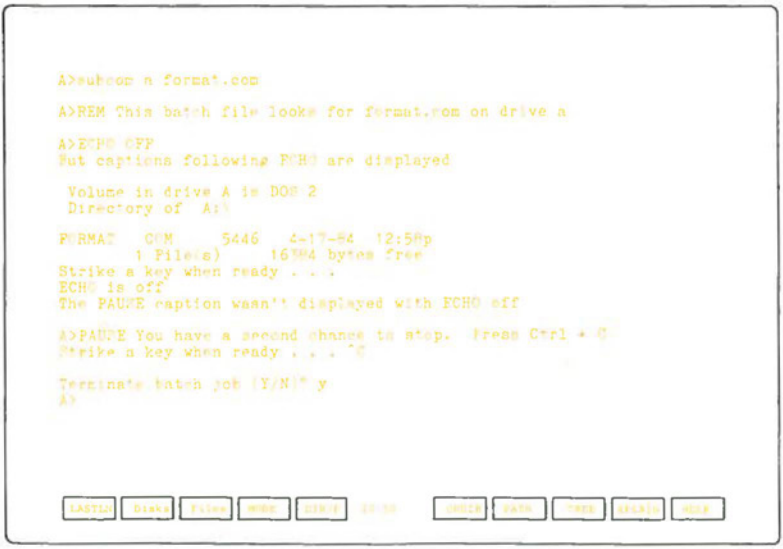


Fig. 6-3: The display produced by the batch file SUBCOM.BAT demonstrating the use of the ECHO and PAUSE subcommands.

The system executes the batch file, producing a display similiar to that shown in Fig. 6-3. Notice that DOS goes to the :START label at the start of the batch file and repeats the list of commands every time it reaches the GOTO command. This continuous repetition of a series of commands is called a *"loop"*.

To stop this loop, wait until DOS displays the PAUSE command and prompts **Strike a key when ready...**

STEP *(cont)*

17) Press the **Ctrl** and **C** keys at the same time.

DOS prompts you:

Terminate batch job (Y/N)?

18) Type in **Y** and press **Return.**

...continued

ECHO controls the display of the DOS commands (and remarks) when a batch file is being executed. Unless otherwise specified, ECHO is set to ON, and all commands and remarks are displayed. If ECHO is set to OFF, only captions following the subcommand ECHO are displayed, along with any normal output from the DOS commands being executed. If ECHO is used without any parameters (ON, OFF or a message) DOS tells you what the current state of ECHO is (ON or OFF).

The PAUSE subcommand is designed to stop the execution of DOS commands so you can perform manual actions, like switching diskettes, or halting the system. When DOS encounters a PAUSE subcommand, it halts processing and displays the message **"Strike a key when ready..."**. If you want to continue, **press any key**, if you want to stop, press the **Ctrl** and **C** keys together (**Ctrl + C**).

The FOR subcommand

Here is an example that shows how the FOR subcommand can be used. In this batch file, DOS is instructed to perform the DIR command on both drives A and B. The line containing the FOR subcommand thus replaces two lines:

STEP *(cont)*

19) Type in **COPY CON FORE.BAT** and press **Return.**

20) When the cursor moves to the line under the command you have just entered, type in the following:

```
:START
FOR %%V IN (A B) DO DIR %%V:%1
REM Do you want to go on? Press Ctrl + C
  to stop.
PAUSE Before pressing a key to go on,
  swap disks
GOTO :START
```

(press **Return** after each line, including the last one).

21) Press **Ctrl** + **Z**, then **Return.**

22) When the A> prompt reappears, type in **FORE FORMAT.COM** and press **Return.**

...continued

The FOR subcommand lets you perform one DOS command in a batch file on several files. The FOR subcommand line is always of the form:

FOR %%variable IN (set) DO command

where:

%%variable is a *one-character* variable name. Be sure to use two percent signs with a variable name, if you use only one percent sign, DOS considers your "variable" as a parameter variable (see the description of parameters at the start of this section).

(set) is the list of the names of the files you want the DOS command to work on. Make sure each file name is separated by a space. You can use wildcards in specifying file names, but you cannot use paths (i.e. all files must be in the current directory).

command is any valid DOS command.

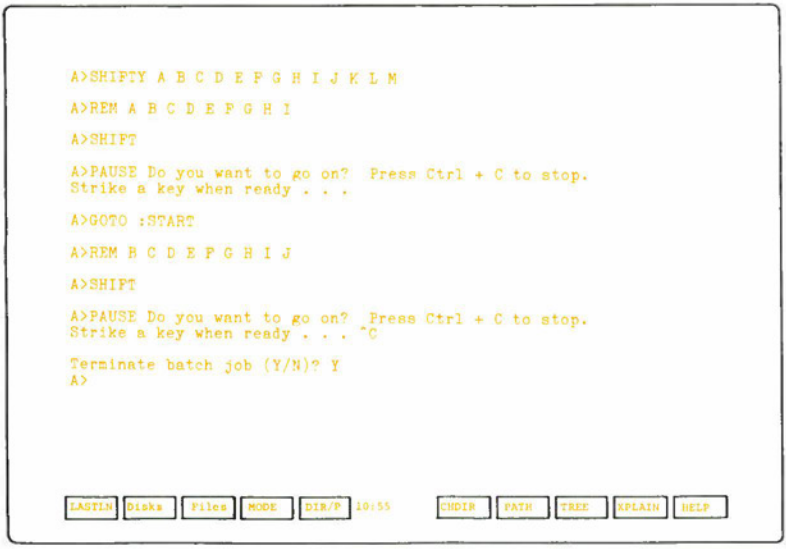


Fig. 6-4: The display produced by the batch file SHIFTY.BAT demonstrating the use of the SHIFT subcommand.

The SHIFT subcommand

Finally, let's look at SHIFT. The following example demonstrates a batch file that displays the first 9 variables entered after the batch filename. On the second loop, it displays the second through tenth variables. On the third loop, it displays the third through eleventh variables, and so on.

The SHIFT subcommand is responsible for the reassigning of variable parameters to the parameters entered after the batch filename.

STEP *(cont)*

23) Type in **COPY CON SHIFTY.BAT** and press **Return.**

24) When the cursor moves to the line under the command you have just entered, type in the following:

```
:START
REM %1 %2 %3 %4 %5 %6 %7 %8 %9
SHIFT
PAUSE Do you want to go on? Press Ctrl + C
  to stop.
GOTO :START
```

(press **Return** after each line, including the last one).

25) Press **Ctrl** + **Z**, then **Return.**

26) When the A> prompt reappears, type in **SHIFTY A B C D E F G H I J K L M** and press **Return.**

The batch commands files (and their output) should resemble Fig. 6-4.

The SHIFT subcommand is used when you want to have more than 9 parameter variables following the batchfilename when you use a batch file.

Since the parameter variable consists only of a single-digit number, the maximum number you can use at one time is 10. One of these is used initially for the name of the batchfile.

SHIFT reassigns each parameter you enter after the batchfilename to the parameter variable in the following manner:

batch file command line:	SHIFTY	A	B	C	D	E	F	G	H	I
parameter variables assigned:	%0	%1	%2	%3	%4	%5	%6	%7	%8	%9

After DOS executes a SHIFT subcommand, the parameter variables are reassigned as:

batch file command line:	A	B	C	D	E	F	G	H	I	J
parameter variables assigned:	%0	%1	%2	%3	%4	%5	%6	%7	%8	%9

After another SHIFT subcommand:

batch file command line:	B	C	D	E	F	G	H	I	J	K
parameter variables assigned:	%0	%1	%2	%3	%4	%5	%6	%7	%8	%9

6.5 SUMMARY OF CONCEPTS

In this lesson you have been introduced to batch files. You have been shown how to:

— create a batch file

— create small batch files directly from the console using the **COPY** command

— insert parameters in batch files

You have also been introduced to the batch file subcommands provided by DOS to make batch files more flexible and powerful:

ECHO — This command controls the displaying of batch file subcommands while they are excuted.

FOR — This command lets you use FOR loops in batch files.

GOTO — This command tells DOS to jump to another line to continue executing commands.

IF — This command lets you use conditional statements in batch files.

SHIFT — This command lets you use more than 9 variable parameters.

PAUSE — This command halts processing to allow you to perform manual activities.

GOTO — This command tells DOS to jump to another line to continue executing commands.

Appendix A

GLOSSARY

Appendix A

GLOSSARY

backslash (\): The character which separates the names of each level in a subdirectory's name. See also: path.

batch file: A file containing a series of DOS commands to be executed sequentially without operator input. All batch files must have the extension .BAT. When the filename of the batch file is entered, DOS executes the commands in the file.

byte: A piece of information which can be either a character or a digit. A floppy diskette holds about 327,000 to 369,000 bytes, depending on how it's formatted. A hard disk can have 10 megabytes (million bytes) of storage or more. See also: FORMAT.

CHDIR: A DOS command used to change the current directory used by DOS to another directory. CHDIR stands for CHange DIRectory. See also: tree-structured directories.

child (directory): A subdirectory contained in a directory is referred to as the *"child"* of that directory. The directory containing the subdirectory is called the *"parent"*. See also: parent(directory), tree-structured directories.

CHKDSK: A DOS command which checks the disk and the information stored on it. CHKDSK's output is a report on the space available, and its condition.

command: An instruction telling DOS to perform a certain action. DOS recognizes only certain words and contractions of words (e.g. MKDIR for MaKe DIRectory, or CHKDSK for CHecK DiSK). See also: parameter, subcommand.

Glossary (cont)

COPY: A DOS command that looks at the contents of a file and makes an exact copy at the specified location. If a file of the same name as the file being copied exists at that location, it is deleted to make way for the new copy. See also: DISKCOPY.

current directory: The directory DOS searches for a command or file name when a command is entered without specifying a path. (i.e. the default value used by DOS for directory). If a PATH command has been entered, DOS searches the directories specified by PATH after failing to find the command or file name in the current directory. See also: CHDIR default, PATH.

data: Data refers to the information used by the computer: the values entered into the computer, processed by it and stored in files. The words data and information are used interchangeably in the manuals accompanying your Hyperion.

DATE: A DOS command which allows you to display or set the system date.

default: A predetermined setting or value for a parameter, which DOS uses if the parameter is not specified. Examples include the default drive, and the current directory. See also: current directory, default drive.

default drive: The drive DOS searches for a command or file name when a command is entered without specifying the disk drive location. See also: default.

device: The name for any "thing", logical or real, that DOS can manipulate and use.

DIR: A DOS command which displays a listing of the files in the specified directory on the specified drive.

DISKCOMP: A DOS command which compares two diskettes to see if they are identical.

Glossary (cont)

DISKCOPY: A DOS command which copies the contents of diskette A to diskette B. The copy made is an exact duplicate. See also: COPY.

diskette: The storage medium used to hold the files the system uses. The Hyperion uses diskettes that are 5-1/4 inches in diameter and that conform to the IBM PC formatting standard.

drive: The physical mechanism into which you insert floppy diskettes. DOS refers to any storage device as a drive: not only the floppy drives (A and B), but also the hard disk (s) attached to the system, and the RAMdisk.

ECHO: A batch file subcommand which turns the display of commands on and off, and displays messages. See also: batch files.

error message: When DOS is asked to perform an impossible task, it reports why it can't do it. The error message will indicate what went wrong, and where it went wrong.

filename: Part of the label which identifies the file to the system. Filenames can be from one to eight legal characters long. See also: filename extension.

filename extension: A one to three character extension after the filename. The extension must be preceded by a period (.). DOS uses certain extensions to identify special categories of files. See also: filename, batch file.

filter command: A filter command is a DOS command which takes data from the standard input device, changes, or "filters" it, and sends the modified data to the standard output device. See also: redirecting input, redirecting output, piping.

Glossary (cont)

FIND: A filter command which locates a user-supplied character string and displays the line on which it is found.

FOR: A batch file subcommand which allows the use of the same command for a series of files which meet a given condition. See also: batch files, set.

FORMAT: A DOS command which prepares a diskette for use by arranging its magnetic pattern into tracks and sectors. See also: hard disk, diskette.

GOTO: A batch file subcommand which directs the program to the line after a label. See also: batch files.

hard disk: A storage medium made up of a large disk, with increased capacity and quicker access time. Usually a sealed unit; also referred to as a *"fixed disk"*. The hard disk must be set up using the FDISK command and formatted using the FORMAT command. See also: diskette, drive.

IF: A batch file subcommand which allows the conditional execution of a command. See also: batch file.

memory: The area inside the computer which is used to hold data and files.

MKDIR: A DOS command used to create subdirectories. MKDIR is a contraction of MaKe DIRectory. See also: tree-structured directories.

MORE: A filter which takes input and displays a single screenful at a time, prompting **"-- More --"** for a keystroke to continue the display.

Glossary (cont)

parameter: Additional words following a DOS command on the command line which direct or modify the command or batch file being run. See also: default.

parent (directory): The directory containing a particular subdirectory. The ".." entry in each subdirectory tells DOS where the parent is. See also: tree-structured directories.

path: The route of subdirectories that lead to a file. If DOS can't find a file in the current directory and a path hasn't been specified, it checks the default path. See also: PATH (command), tree-structured directories.

PATH (command): A DOS command which tells DOS where to look for a command or file name if the command or file name was not specified in the command line, and is not found in the current directory. See also: current directory, path.

PAUSE: A batch file subcommand which halts execution of the file, prompting you to press a key to continue. See also: batch files.

pipe: A computer-generated link between two DOS commands symbolized by "|", a pipe takes the output of one command and makes it the input for a second command. See also: filter commands redirecting input, redirecting output.

program: Instructions to the computer on how to perform a given task. Programs are stored in files, usually with the filename extension EXE or COM.

redirecting input: DOS's standard input device is the keyboard. You can tell DOS to take input from any other device, e.g. data from a disk file, or the output from a program.

Glossary (cont)

redirecting output: DOS normally sends output to the screen. You can redirect the output to another device, like a printer or a disk file.

REM: A batch file subcommand which displays a message or remark on the screen during the batch file's operation. REM is a contraction of REMark. See also: batch files, ECHO.

RMDIR: This DOS command removes subdirectories from your directory structure. The subdirectory must be empty before DOS can remove it. RMDIR is a contraction for ReMove DIRectory. See also: tree-structured directories.

root directory: The base directory from which the other branches originate. See also: tree-structured directories.

save: Storing a file, so that it can be read at a later date.

sector: A section of a track on a diskette. DOS 2.11 formats and uses diskettes with either 8 or 9 sectors per track. Each sector contains 512 bytes of information. See also: FORMAT, track.

set: A group of files used by FOR to generate a particular command. See also: batch files, FOR.

SHIFT: A batch file subcommand which moves command line parameter values from their current marker to the preceding marker. See also: batch files, variable parameters.

SORT: A DOS filter command which takes input and arranges it in ascending alphabetical order. The resulting output can be directed to another program, or any other device. See also: redirecting input, redirecting output, piping.

system prompt: A character or string of characters which indicate you are at the DOS command level. The prompt is usually a letter which corresponds to the default drive, i.e. A, B, C, or D. See also: PROMPT.

Glossary (cont)

temporary file: A short-lived DOS file created to hold the output of one command or filter before it is sent to its destination. DOS erases these file automatically when their purpose has been served.

TIME: A DOS command which allows you to set and display the system time.

track: A circle in the magnetic pattern on the diskette. DOS divides each track into either 8 or 9 sectors, each of which can contain 512 bytes of information. See also: sector, FORMAT.

tree-structured directories: The method DOS uses to organize and find files. From the root directory, subdirectories branch out, each with its own files and subdirectories.

TREE: A DOS command used to list all the directories (and files, if the /F parameter is specified) on a diskette. See also: tree-structured directories.

variable parameter: A special two-character symbol in a batch file, used to tell DOS where to place a parameter. The first symbol is always the percentage sign (%), the second is a digit from 0 to 9. The digit indicates which parameter on the command line is to be substituted for the marker. See also: batch file, parameter, SHIFT.

volume: The name given to a disk storage area. This usually means one of the diskette drives or else the DOS partition (section of disk DOS uses) on the hard disk, but not the RAM disk. A diskette or the DOS partition on a disk can be given a Volume Name with the VOL or DISKNAME commands, or a /V parameter following the FORMAT command. See also: FORMAT.

Appendix B

THE HYPERION FAMILY

Appendix B

THE HYPERION FAMILY

There are currently two models in the family of portable business computers manufactured by Comterm Inc. These are:

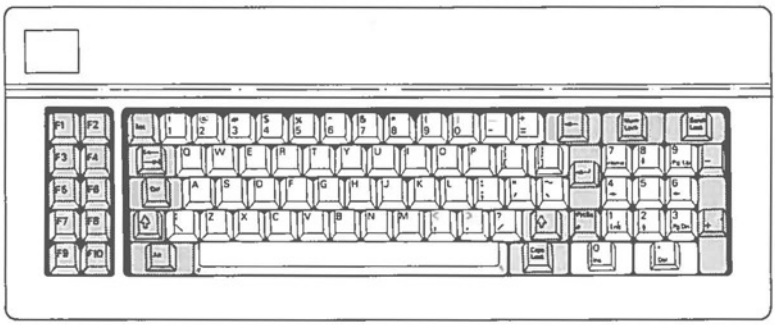
* the Hyperion PC, the newest model in the Hyperion family, featuring graphics conforming to the standard found on most members in the MS-DOS family of business microcomputers.

* the Hyperion business computer, featuring high-resolution monochrome graphics and an ergometrically designed keyboard. Two versions of this popular machine are available, the Revision 9, and Revision 5 models. They are functionally identical in all respects except for the design of their disk drive latches.

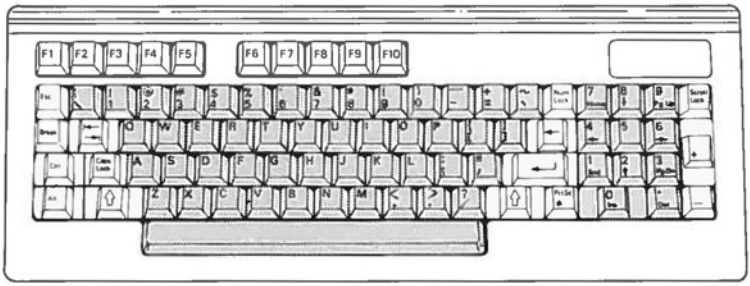
KEYBOARD DIFFERENCES

DOS, version 2.11, is meant to work on several different machines, notably the Hyperion PC, the Hyperion, and the IBM PC. These three machines, though they have the same keyboard type with 83 keys, have certain keys that are designated differently depending on the machine.

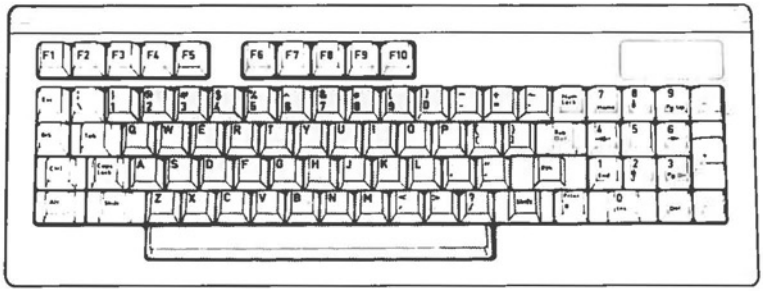
Fig. B-1 shows three different keyboards. Note that throughout this manual the standard key names (which can be different from the key cap labels found on the keyboards) are used.



(a) The IBM PC keyboard.



(b) The Hyperion PC keyboard.



(c) The Hyperion keyboard.

Fig. B-1 — Different keyboards for the machines that accept DOS (2.11).

KEYCAPS WHICH DIFFER FROM MACHINE TO MACHINE

KEY NAME	IBM PC	HYPERION PC	HYPERION
Tab			Tab
Shift			Shift
Print	PrtSc	PrtSc	Print
Rub Out			Rub Out
Break	N/A	Break	Brk
Return			Return

FEATURES AVAILABLE FOR THE HYPERION FAMILY:

FEATURE:	Hyperion	Hyperion PC
8088 processor	STD	STD
8087 co-processor	OPT	OPT
MS-DOS v. 2.11	STD	STD
Built-in monochrome monitor	STD	STD
External monochome jack	STD	STD
External color (RGB) jack	N/A	STD
Parallel port connector	STD	STD
Serial port connector	STD	OPT
Built-in modem connector	STD	OPT
Acoustic cup connectors	STD	OPT
Expansion chassis connector	STD	OPT

620035-000-00