Using Your Integral PC

HEWLETT PACKARD

Edition 1 October 1985

Reorder Number
00095-90123

Notice

The information contained in this document is subject to change without notice.

Hewlett-Packard makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Hewlett-Packard assumes no responsibility for the use or reliability of its software on equipment that is not furnished by Hewlett-Packard.

Portable Computer Division
1000 N.E. Circle Blvd.
Corvallis, OR 97330, U.S.A.

Printing History **Edition 1** October 1985 Mfg. No. 00095-90118

HP Computer Museum
www.hpmuseum.net

For research and education purposes only.

Contents

Chapter 9 More About PAM

Chapter 10 Installing and Using Peripherals

Chapter 11 Command and Program Dictionary

Appendixes & Index

Safety and Regulatory Information

For your protection this product has been tested to national and international standards assuring safe use. Regulatory testing does not address the reliability or performance of the product. The scope of regulatory testing includes electrical/mechanical safety, radio frequency interference, ergonomics, acoustics, and hazardous materials. The approvals obtained from independent test agencies are shown on the product label located on the rear panel. In addition, various regulatory agencies require some of the information under the following headings.

Potential for Radio/ Television Interference

This equipment has been certified to comply with the limits for a Class B computing device, pursuant to Subpart J of Part 15 of FCC Rules. Only peripheral (computer input/output devices, terminals, printers, etc.) certified to comply with the Class B limits may be attached to the computer. Operation with non-certified peripherals is likely to result in interference to radio and television reception. All HP-IB cables must be shielded and ferrite-beaded (HP 82977A/B). All other cables except HP-IL must be shielded.

This product generates and uses radio-frequency energy, and, if not installed and used in strict accordance with the instructions in the owner's documentation, may cause interference to radio and television reception. It has been tested and found to comply with the limits for a Class B computing device in ac-

cordance with the specifications in Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference in a residential installation. However, there is no guarantee that interference will not occur in a particular installation. In the unlikely event that there is interference to radio or television reception (which can be determined by removing all power to the product or turning it off and on), you are encouraged to try to correct the interference by one or more of the following measures:

- Reorient the receiving antenna.
- Relocate the product with respect to the receiver.
- Plug the product into a different ac outlet so that the computer and the receiver are on different branch circuits.

If necessary, you should consult your dealer or an experienced radio/television technician for additional suggestions. You may find the following booklet, prepared by the Federal Communications Commission, helpful: *How to Identify and Resolve Radio-TV Interference Problems.* This booklet is available from the U.S. Government Printing Office, Washington, D.C. 20402, Stock Number 004-000-00345-4.

South African Radio Frequency Interference

This product has been certified by S.A.B.S. to meet the RFI requirements published in the Government Gazette of December, 1979 (No. 6794) in Notice R2862 and subsequent amendments known as Rule 68B.

German Radio Protection Mark

This product has been tested together with Hewlett-Packard peripherals and complies with VFG 1046/84, VDE 0871B and similar non-interference standards.

Should you connect equipment that is not manufactured or recommended by Hewlett-Packard Company, that system configuration has to comply with the requirements of Paragraph 2 of the German Federal Gazette, Order (VFG) 1046/84, dated December 14, 1984.

Other Safety Approvals

This product has been listed by the Underwriters' Laboratories, Inc., under UL standard UL 478.

This product has been tested and found to comply with VDE standard DIN VDE 0806 (IEC 380) by TÜV-Bayern e.V.

This product has been tested and found to comply with Norwegian safety and RFI standards by NEMKO.

This product has been tested and found to comply with the applicable Canadian data processing requirements and standards by the Canadian Standards Association (CSA 22.2 No. 154).

Guide to Documentation

If You Are Interested In:	And You Want To:	Read:
Getting started	Set up your computer	*Setup Guide*
	Start the Tutor program	*Setup Guide*
	Learn about operating your computer	Tutor disc or this manual (chapters 1–7)
Personal Applications Manager (PAM)	Learn the basics of PAM	Tutor disc or this manual (chapters 1–7)
	Find reference information on a particular PAM command	This manual (chapter 11)
	Start an application program	Tutor disc or this manual (chapter 2)
Peripheral devices	Connect a printer, plotter, disc drive, memory module, or other device to your computer	*Connecting Peripherals to Your Integral PC*
	Use them	This manual (chapter 10)
HP-UX commands and utilities	Find more information on them	This manual (chapter 11)
Diagnostic program	Test your computer	This manual (appendix A)

1 Getting Started

Contents

1 Getting Started

Are You Ready?

This manual assumes that you have already performed the following steps:

1. Unpacked your computer and checked the contents against the packing list.
2. Read the software license and warranty.
3. Set up your computer according to the instructions in your *Setup Guide*.
4. Started the Tutor program according to the instructions in your *Setup Guide*.
5. Completed chapter 1, "Getting Started," of the Tutor program.

Please proceed now as follows:

- If you have completed all these steps, skip ahead to "About the Tutor" below.
- If you haven't tried these steps yet, do so before continuing with this manual.
- If you tried these steps and had problems, refer to appendix A for help. It discusses

  How to check the voltage-selector switch.

  How to check the fuse.

  How to insert the Tutor disc.

  How to start the Tutor program.

  How to check the keyboard (and mouse).

  How to check other problems.

About the Tutor

The contents of part I of this manual are similar to the contents of the Tutor. If you're a beginner, the Tutor is ideal because you can practice using the computer while you learn. If you're comfortable with the computer, this manual is handy because you can quickly find the steps needed to perform a task. Choose the presentation you prefer.

Here is the table of contents for the Tutor:

Chapter 1: Getting Started

Information for everyone.

Lesson 1: Are You Ready?
For first-time users.

Lesson 2: Using the System
Includes *window shuffling.*

Lesson 3: Setting the Time and Date
You only need to do this once or twice a year.

Lesson 4: Checking the Time
In which you create a small clock window.

Lesson 5: Starting Over
For those rare times when a system reset is in order.

Chapter 2: Running a Program

The reason for having a computer.

Lesson 1: Starting a Program
You'll insert a disc, pick a program, and go.

Lesson 2: Inverting Window Colors
You have a choice—dark on light, or *vice versa.*

Lesson 3: Stopping a Program
Use the (Stop) key or a function key.

Lesson 4: Updating PAM
To reread your disc and update the display.

Chapter 3: Typing Your Information

Keyboard practice.

Lesson 1: Entering Text
You'll practice the typing keys.

Lesson 2: Editing Text
You'll use (Insert char), (Delete char), etc.

Lesson 3: The Electronic Typewriter
Not exactly a word processor but fun nonetheless.

Chapter 4: Viewing and Printing Your Information

Printed copies.

Lesson 1: Viewing a Text File
You'll highlight a file name and press View.

Lesson 2: Positioning the Paper
How to load paper, advance paper, and set top-of-form.

Lesson 3: Copying the Display
You'll use the (Shift)(Print) keystroke.

Lesson 4: Checking the Printer
A simple self-test.

Lesson 5: Printing a Text File
You'll use the `print` command.

Chapter 5: Protecting Your Information

Disc and file backups.

Lesson 1: Safeguarding Your Discs
How to handle and protect discs.

Lesson 2: Making Extra Disc Copies
Includes formatting and copying instructions.

Lesson 3: Making Extra File Copies
You'll use the Copy key.

Chapter 6: Organizing Your Information

File manipulations.

Lesson 1: Finding Files
You'll open and close folders to get at the files within.

Lesson 2: Renaming Files and Discs
You'll use the Rename key and `rename_disc` program.

Lesson 3: Moving Files Into Folders
You'll create a new folder and put a file in that folder.

Lesson 4: Copying Files Across Discs
You'll use the Copy key.

Lesson 5: Deleting Files and Folders
You'll use the Delete key.

Chapter 7: Managing Windows

In which your programs run.

Lesson 1: Selecting Windows
You'll study how programs fit into windows.

Lesson 2: Pausing and Restarting Windows
You'll use the Pause key.

Lesson 3: Hiding and Showing Windows
You can put away and take out windows at will.

Lesson 4: Moving and Stretching Windows
You'll move and change the size of a sample window.

About the Documentation

Your computer is shipped with five documents:

- The *Setup Guide* shows you how to set up your computer, how to prepare its printer for use, and how to prepare the computer for moving.
- The *Software License and Warranty* describes your rights to use the software you received with the computer. It also describes the warranty for the software.
- The *Support Guide* describes the warranty for your computer and how to obtain service.
- This *Using Your Integral PC* manual describes fundamental concepts and gives step-by-step procedures for common tasks in part I. In part II, this manual describes operations in greater detail, provides reference information, and presents additional information, such as using peripherals and using advanced capabilities of the computer.
- The *Connecting Peripherals to Your Integral PC* manual shows you how to connect printers, disc drives, and other peripherals and accessories. If you have no peripherals or accessories, you don't need to refer to this manual.

Each of the discs provides additional documentation. A `README` file on each disc describes the contents of that disc. Detailed information about certain programs on the discs is provided by "doc" files on the discs. The documentation files are labeled with `.doc` suffixes on their names. Sometimes these files are contained in a separate part of the disc—a "folder" whose name has a `.doc` suffix. You'll learn how to display and print these documentation files in chapter 4.

Using This Manual

This manual gives you step-by-step instructions for using the computer (in part I) and detailed reference information (in part II). You should complete at least the first two chapters in this manual (or in the Tutor) before running an application program or going on to later chapters. You can read the later chapters in any order, as you need them.

The first two chapters describe each step in detail. Some steps include alternative methods using the optional mouse. The step-by-step instructions in later chapters omit the details for each step.

This manual uses symbols to help you locate and identify information. The symbols have the following meanings:

Mouse  This symbol marks instructions for using the optional mouse.

Warning  This symbol marks a warning that you should observe to avoid bodily injury.

Caution  This symbol marks a caution that you should observe to avoid damage to the computer or serious loss of data.

Note  This symbol marks a note that you should observe to avoid minor problems.

About the Discs

Your computer is shipped with several discs that are part of the basic computer system.

- **Tutor disc:** an introduction for beginners.
- **Utilities disc:** programs for performing fundamental operations. (The tasks described in part I of this manual use this disc.)
- **Commands 1 and 2 discs:** a set of standard HP-UX commands for advanced users.
- **System disc:** a collection of special-purpose files.
- **Diagnostic disc:** a comprehensive test of the computer system.

Each of these discs contains a text file named `README` that describes the contents of the disc. Chapter 4 shows you how to view and print a `README` file.

Using the Built-In Disc Drive

The computer contains a built-in "microflexible" disc drive. In addition, you can connect peripheral disc drives to your system—these *external* drives are discussed in chapter 10.

The built-in disc drive is the "default" drive that the computer uses for certain operations (unless you specify an external drive).

Whenever you insert a disc into the built-in disc drive, the computer automatically "mounts" the disc—that is, the computer is immediately able to access files contained on the disc. In contrast, this isn't true for discs inserted in external drives (refer to chapter 10). For this reason, the built-in drive is most convenient for "swapping" discs.

The built-in disc drive has two equivalent "names" that you can use to specify that drive for certain operations: `A` and `internal`.

Using a Mouse

You can connect an optional mouse to your computer. (Refer to the *Setup Guide*.) This accessory is a "pointing" device that enables you to make selections more quickly and easily than by using the keyboard alone. Although you can perform all operations without using a mouse, this manual shows alternative methods that *do* use the mouse. If you use a mouse, you'll soon develop a sense for the most efficient uses for the keyboard and the mouse.

In addition, the computer provides a special feature that's particularly useful to mouse users—display *borders* that enable you to perform common operations with a minimum of mouse movement. Because the borders are useful mainly to mouse users, they aren't described in part I of this manual—they're described in chapter 8.

Introducing the Display Pointer

The rest of this chapter reviews concepts covered in chapter 1 of the Tutor. If you haven't run the Tutor yet, do so now. Because these concepts are essential to everything that follows, it's worthwhile to use the Tutor and then come back here for a review.

The display shows a small figure called the *display pointer*. It usually looks like an arrow, but it looks like a diamond when it points to the background, or it can take a special shape when it's used with a program that redefines it. The display pointer allows you to control the computer by choosing an item or location.

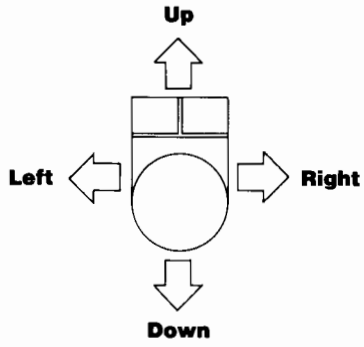
To use the display pointer:

1. Move the display pointer to the desired item or location.

   Hold down the CTRL key while pressing the arrow keys, ▲, ▼, ◀, and ▶. The display pointer moves in the corresponding direction.

   You can move the display pointer diagonally by holding down the CTRL key while pressing a pair of arrow keys.

   If you're using a mouse, just roll the mouse in the direction you want the display pointer to move.

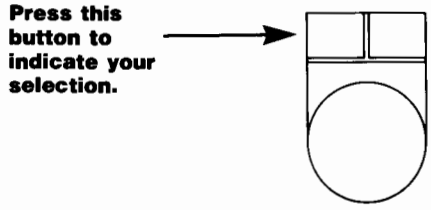


   You can lift and reposition the mouse on your desk without moving the display pointer.

2. Indicate your selection.

   Press the (Select) key (just to the left of the arrow keys).

   If you're using a mouse, press the left button.



   (Pressing the right button turns off the menu. You can turn the menu back on by pressing the right button a second time.)

A few steps in the instructions in this manual *require* you to use the display pointer. More often there are two ways to perform a step, one of which uses the display pointer. If you have a mouse, you'll probably want to use the display-pointer method. This manual marks *optional* display-pointer methods with the mouse symbol. But remember that you can use the display pointer with or without a mouse.

Setting and Displaying the Time and Date

You should have already set the time and date, and then displayed the time and date, when you ran chapter 1 of the Tutor. The time and date are maintained when you turn off the computer, so you won't need to set them again unless your local time changes.

You normally display the time and date using the PAM (Personal Applications Manager) window. (In fact, PAM provides convenient management for *all* of your computer tasks.) Although displaying the time and date may not seem too important, the steps described in detail here appear again and again in this manual for other operations. Later chapters assume that you know how to perform them.

To set the time and date:

1. Select the PAM window.

   This step ensures that PAM is the *active window*—the window that receives your input. Look at the center of the bottom of the display. If it shows "PAM", PAM is already the active window—and you don't need to do anything for this step.

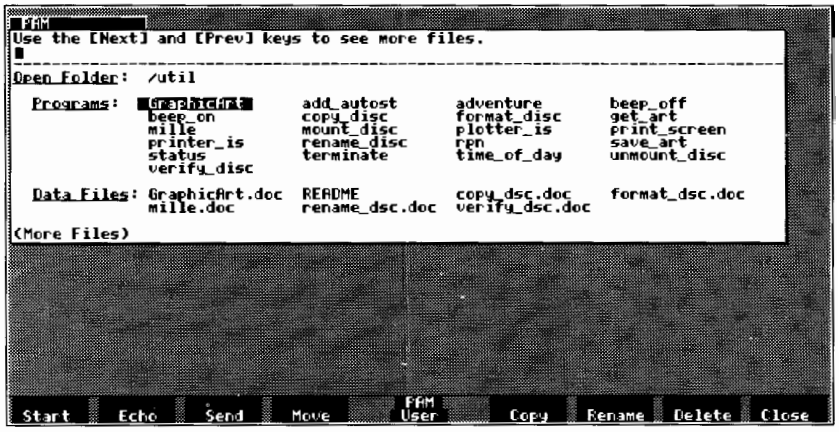
   If PAM isn't already the active window, press (Shift)(Select). This keystroke *shuffles the windows*—it brings another window to the front and makes it the active window. Repeat this keystroke until PAM is the active window.

   With a mouse, move the display pointer to PAM's *window banner* (the "PAM" in the upper-left corner of the display) or to any part of the PAM window, and press the left button on the mouse. This brings the PAM window to the front and makes it the active window.

2. Insert the Utilities disc.

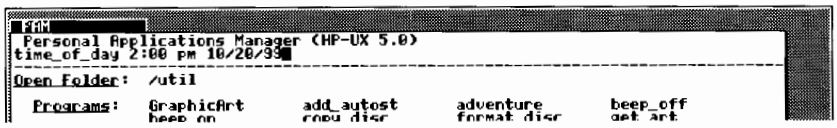
   If there is already a disc in the drive, make sure the colored "drive busy" light is off. Then press the disc-eject button (near the corner of the drive opening), remove the disc, and insert the Utilities disc. The colored "drive busy" light flashes as the computer reads the Utilities disc.

PAM updates its window to show the contents of the Utilities disc:



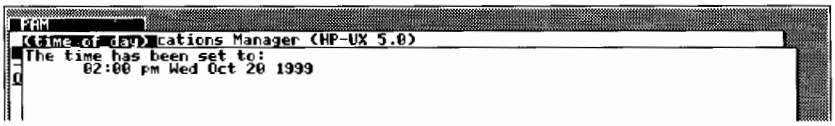
If a different contents is shown, check that you're using the Utilities disc. Then type `/util` (Return) to display the contents of the Utilities disc.

3. Type `time_of_day` followed by the time and date. Type "underscore" characters as shown. Use the (Back space) key to correct mistakes. For example, you might type



4. Press (Return).

   A `time_of_day` window will be created to display the following message:



5. Press (Stop) to remove this message window.

At any time you can display the time and date, as described next. If the time and date are *already* being displayed, they won't be updated until the next minute.

To display the time and date:

1. Select the PAM window.

   This step ensures that PAM is the *active window*—the window that receives your input. Look at the center of the bottom of the display. If it shows "PAM", PAM is already the active window—and you don't need to do anything for this step.

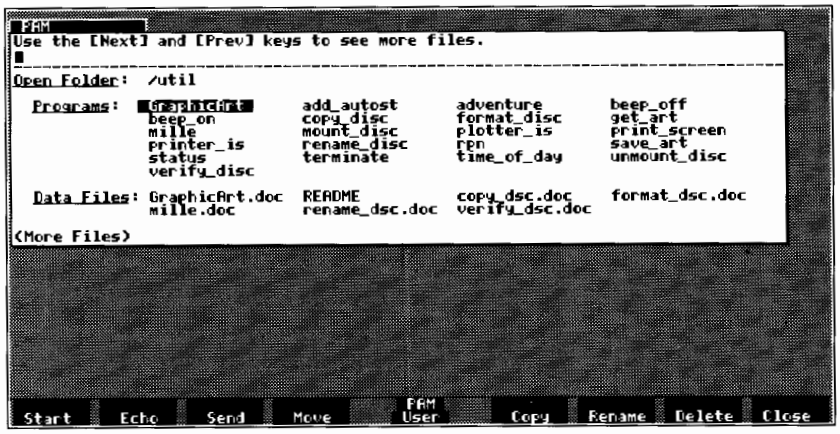
   If PAM isn't already the active window, press (Shift)(Select). This keystroke *shuffles the windows*—it brings another window to the front and makes it the active window. Repeat this keystroke until PAM is the active window.

   With a mouse, move the display pointer to PAM's *window banner* (the "PAM" in the upper-left corner of the display) or to any part of the PAM window, and press the left button on the mouse. This brings the PAM window to the front and makes it the active window.

2. Insert the Utilities disc.

   If there is already a disc in the drive, make sure the colored "drive busy" light is off. Then press the disc-eject button (near the corner of the drive opening), remove the disc, and insert the Utilities disc. The colored "drive busy" light flashes as the computer reads the Utilities disc.

   PAM updates its window to show the contents of the Utilities disc:



   If a different contents is shown, check that you're using the Utilities disc. Then type `/util` (Return) to display the contents of the Utilities disc.

3. Highlight and start the `time_of_day` program.

   Note that the name of the first program listed in the PAM window is highlighted by an inverse-video bar. Press the arrow keys (in the lower-right corner of the keyboard) to highlight `time_of_day`.

Now find the function-key label Start in the lower-left corner of the screen. (This label appears in the PAM menu only when a program is highlighted.) Press corresponding function key (f1) to start the `time_of_day` program.

With a mouse, move the display pointer to `time_of_day` and press the left button on the mouse once to highlight `time_of_day`. Then press the left button a second time to start the `time_of_day` program.

PAM starts the `time_of_day` program in a small "hidden" window in the lower-left corner of the display. After a few moments, the window changes to the normal `time_of_day` window in the upper-right corner of the display:

The `time_of_day` program updates the displayed time every minute.

The `time_of_day` window is designed to remain visible without interfering with your handling of other windows. Because you never need to direct input to the `time_of_day` window, it doesn't take a turn as the active window when you shuffle through all the windows. This saves you from extra shuffling. You may encounter similar windows with some application programs.

To stop the time_of_day program:

1. Select the `time_of_day` window.

   Because you can't select the `time_of_day` window by shuffling windows, you must use the display pointer. First move the display pointer to the `time_of_day` window, and then indicate your selection. (Instructions for using the display pointer are under "Introducing the Display Pointer" above.)

2. Press (Stop).

   This keystroke stops the `time_of_day` program and eliminates its window.

The PAM Window

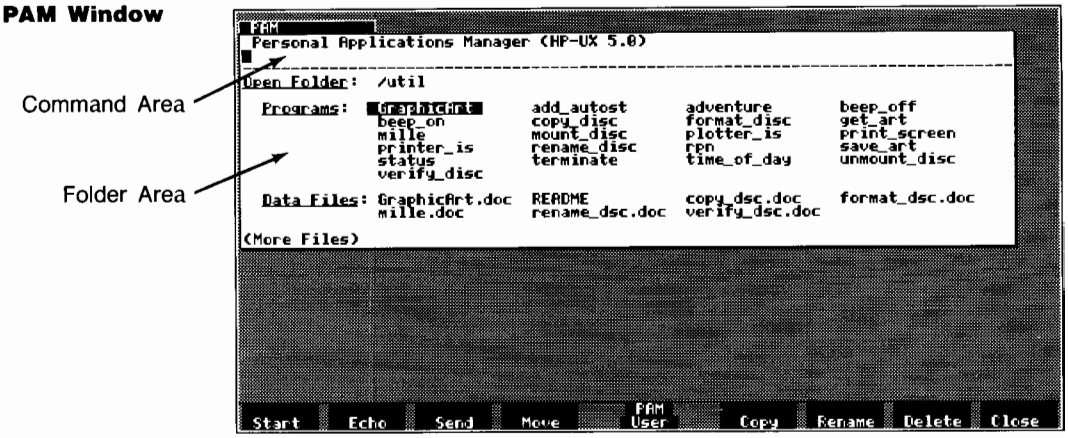
PAM—the Personal Applications Manager—is always available. Whenever you turn on the computer, it automatically creates a PAM window. Accessing PAM is simply a matter of selecting the PAM window ("shuffling" windows until the PAM window is in front of other windows in the display).*

The PAM window is divided into two sections: the command area and the folder area. You issue commands to PAM and receive responses from PAM in the upper portion, the command area. PAM uses the lower portion, the folder area, to display the names of files in the *open folder*—the group of files that you can access most directly.

* If there is an autostart file on a disc in any disc drive, the file is executed when the computer is first turned on, as though you typed it as the first command. For example, the Tutor program is started automatically by an autostart program on the Tutor disc. The PAM window will not normally appear until the program has completed.

A typical PAM window looks like this:

The PAM Window



The PAM command area contains two lines. The first line is the message line, and the second line is the command line.

PAM uses the message line to display instructions and messages. The messages inform you that an operation in done or that a problem exists. For some cases, the message contains suggestions about how to fix the problem. The most common errors include syntax errors in commands and attempts to access files that do not exist.

Other messages may prompt you to complete a certain task. For example, PAM uses prompting when you copy a file—PAM uses the message line to tell you how to specify the destination of the copy operation. Appendix E contains a complete list of PAM messages and their meanings.

The command line allows you to enter commands directly from the keyboard. This lets you specify operations that involve files in folders other than the open folder and to enter commands that require parameters.

The folder area shows you the files that you can use most directly. You can use the arrow keys (or mouse) to move the highlight around the area and "point" at individual files.

If there are too many files to show in the PAM window, `<More files>` appears at the bottom of the window. You can press (Shift)(Next) or (Shift)(Prev) to page through the listing.

With a mouse, you can move the display pointer to the `<More files>` position and press the left button on the mouse. This shows the next "page" of files (and eventually cycles back to the first "page").

Starting Over

As you learn about your computer and try new things, you may get confused about where you are and what to do next. This is OK—it's important to experiment, and you can't hurt the computer by pressing keys. If you want to start over, press (Shift)(Reset). This keystroke stops all programs and eliminates all windows. If the Tutor disc is in the built-in drive, the Tutor program will start automatically.

2 Running a Program

Contents

Chapter 2

2 Running a Program

This chapter completes the basic skills you need for running application programs. These skills are described in detail in this chapter—later chapters assume that you know how to perform them.

Starting a Program

You've already started two programs, the Tutor program and the `time_of_day` program. You started each of these programs in a different way. This points out that there are two ways to start programs: automatic starting using an *autostart* program, and manually starting a program.

The Tutor program is started by an autostart program—its name is `Autost`. If you insert the disc and turn on the computer, the program starts automatically. Some of your applications may be programs named `Autost`. If so, you can start them in the same way you started the Tutor.

The `time_of_day` program, which you started in chapter 1, is not an autostart program—you started it manually. You'll start most of your applications manually—in the same way you started the `time_of_day` program. This section begins with a review of the three steps you learned to start the `time_of_day` program.

To start a program:

1. Select the PAM window.
2. Insert the disc containing the desired program.

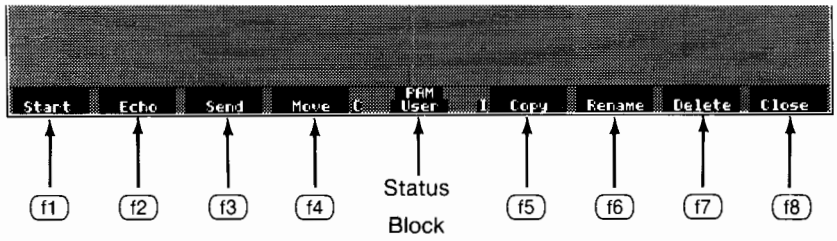
   PAM updates its window to show the contents of the disc. (If the PAM window doesn't show the disc contents, type the disc name—such as `/util`—and press (Return).)
3. Highlight and start the program.

   PAM starts the program in a new window, which appears in front of the PAM window. Usually the new window becomes the active window. (The `time_of_day` window is an exception.)

Additional Information. If you have no external disc drive and you want to run a program on one disc that accesses another disc, you can copy the program file into computer memory. Then you can run the program while the other disc is inserted. Refer to "Making a Copy of a File" in chapter 5.

Controlling a Program

You can control many applications by simply executing commands from the *menu*. The menu appears across the bottom of the display. It contains *function-key labels*, which show the commands currently assigned to the function keys (f1) through (f8). In the center of the menu is the *status block*, which identifies the set of commands currently assigned to the function keys. For example:



To "press" a function key with a mouse, move the display pointer to the function-key label and press the left button on the mouse.

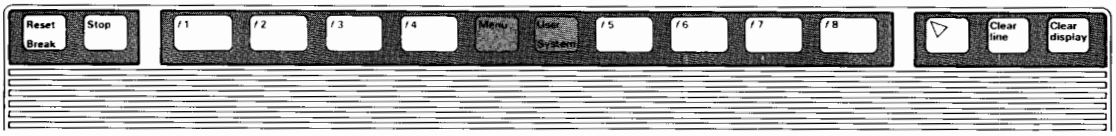
The status block has two parts that combine to identify the current menu. The upper part identifies the *active window* ("PAM" in the illustration above), and the lower part identifies the *active menu* ("User" in the illustration above).

- The *active window* is normally the one that appears in front of the other windows. It receives the commands you give from the menu and receives the characters you type from the keyboard. When you select a window, it becomes the active window. You can check that you selected the correct window by checking the status block.
- The *active menu* is normally the *User* menu, which contains commands to control the program. The commands in the User menu differ from program to program because each program has different commands. There is also a *System* menu, which contains commands to control the window. The commands in the System menu are similar for all windows.

In addition, the status block provides two indicators. The *caps indicator*—a "C"—notes that letters will be entered in upper case (unless you hold down the Shift key). The *insert indicator*—an "I"—notes that characters will be inserted, moving existing characters to the right. An indicator is displayed only while the corresponding feature is active.

Controlling the Menu

You can control the menu by pressing the (Menu) and (User/System) keys. They are located between function keys (f4) and (f5).



To turn the menu on or off: Press the (Menu) key.

With a mouse, press the right button.

To switch between the User menu and the System menu: Press the (User/System) key.

With a mouse, move the display pointer to the status block (in the center of the menu) and press the left button on the mouse.

Changing a Window Background

The Invert command on the System menu changes the color of the window background. You can choose whether a window has an light background or a dark background. This section gives detailed instructions on how to use Invert as an example of how to use the System menu. Other commands on the System menu appear in chapter 7, "Managing Several Programs at Once."

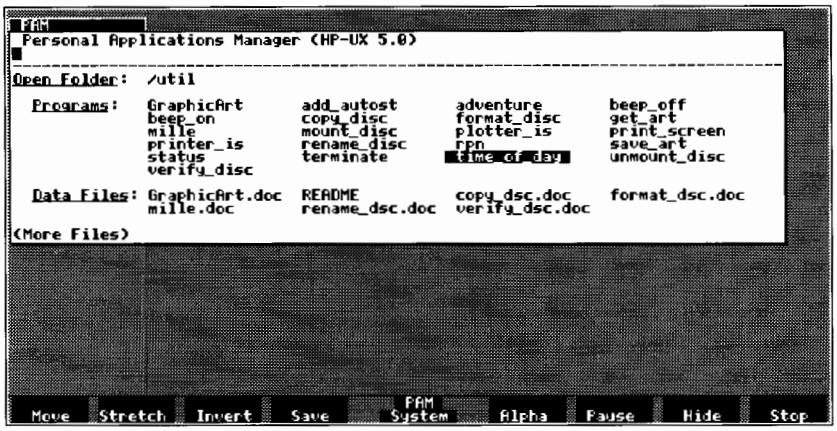
To change a window background:

1. Switch to the System menu.

   If the status block shows "User", press (User/System). The "System" label should appear.

   With a mouse, move the display pointer to the status block (in the center of the menu) and press the left button on the mouse.

   The display shows:



2. Press Invert ((f3)).

With a mouse, move the display pointer to Invert and press the left button on the mouse.

The display shows:

Note that an asterisk appears in the Invert label (Invert*). If you press Invert a second time, the window returns to its original appearance, and the asterisk disappears.

3. Switch back to the User menu.

Press (User/System).

With a mouse, move the display pointer to the status block (in the center of the menu) and press the left button on the mouse.

The above example shows an inverted PAM window. If you start a program from an inverted PAM window, the new program window will also be inverted.

The Move, Stretch, Pause, and Hide operations are described in chapter 7, "Managing Several Programs at Once." Refer also to "Menus" in chapter 8 for additional information about the System menu.

Stopping a Program

Often a program has a command in its User menu to save your data and stop the program. Refer to the documentation for each program. If a program has a command such as Save, Quit, or Exit, be sure to use these commands.

Some programs don't have commands in their User menus. If you need to save your data from such a program, refer to the documentation for the program. To stop such a program, make its window the active window and then press (Stop). (The steps you followed to stop the `time_of_day` program are on page 1-5.)

If pressing (Stop) doesn't stop the program, press (Shift)(Stop). Doing so risks losing the program data, so try (Stop) first. Refer also to "Saving and Removing Windows" in chapter 7 for more information about stopping a program.

Note

Your computer is a multiprocessing system. You don't need to stop one program to run another. This is discussed in chapter 7, "Managing Several Programs at Once."

Updating PAM

Some application programs can create new files. For example, a word processing program can create new text files, and a spreadsheet program can create new spreadsheet files. Unlike the files that PAM creates, the names of these new files won't automatically appear in the PAM window. You can tell PAM to check for and display new files as follows.

To update PAM:

1. Select the PAM window.
2. Highlight the name of the open folder (upper-left corner of the PAM window).
3. Press Reread (f1).

   PAM updates its window to include all new files in the open folder.

More About Applications

If you're primarily interested in running application programs, you can skip certain portions of this manual for now. You already know how to start a program—you've already started the Tutor program and the `time_of_day` program—and you know how to stop a program. But it's also important to understand the main components of the computer and how to manage the files you'll be using.

Before trying your application programs, consider these suggestions:

- Before running your application programs, work through chapter 3, "Typing Your Information," in the Tutor or in this manual. Almost all application programs require keyboard input.
- When required, work through chapter 4, "Viewing and Printing Your Information." You'll probably want printed copies of your work.
- When required, work through chapter 5, "Protecting Your Information." It's important to safeguard your programs and data against loss.
- You should refer to the manual that comes with the application. The manual describes the application's operation, menus, and system requirements.

3 Typing Your Information

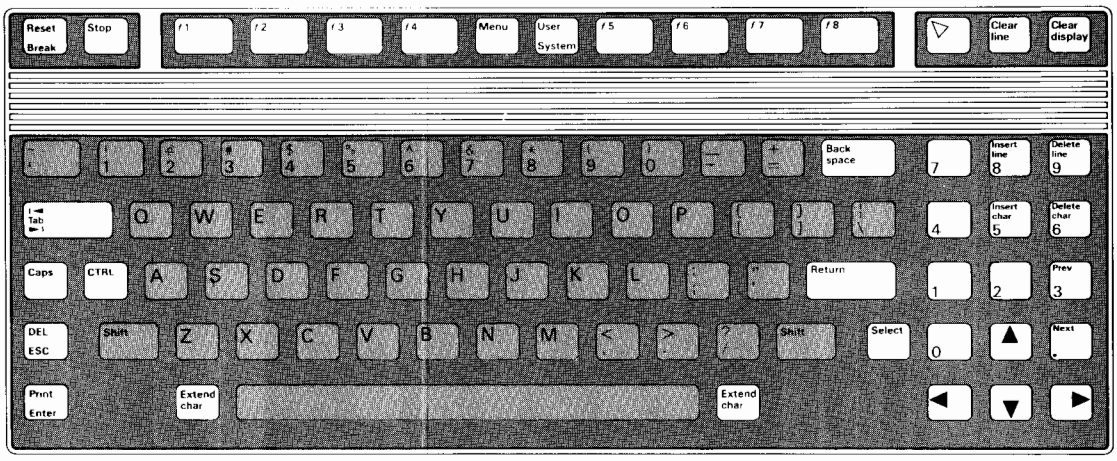
Contents

3 Typing Your Information

When you're running an application such as word processing or a spreadsheet, you'll need to type information into the computer. This chapter tells you the basics about entering and editing that text. However, your application may give you additional capabilities or redefine some operations. Be sure to check the documentation for the application.

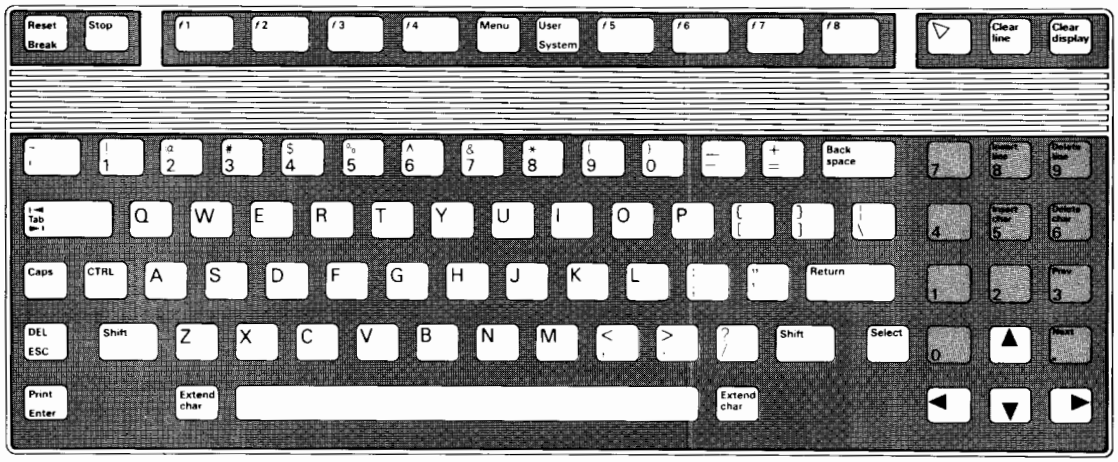
Entering Text

Most of the keys on the main keyboard are like those on a standard typewriter keyboard. You can type uppercase letters, lowercase letters, numbers, and symbols. Be sure to distinguish between the letter "l" and the number "1", and between the letter "O" and the number "0".



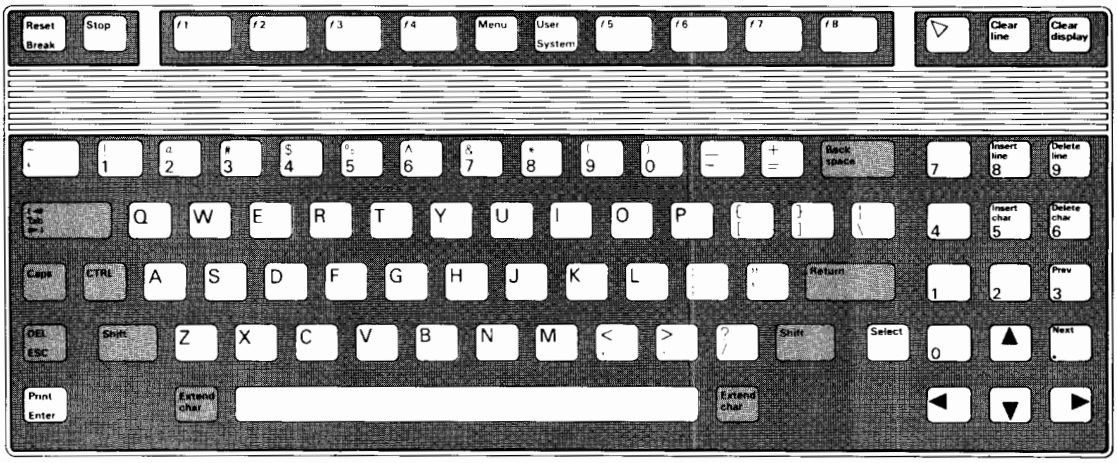
The Numeric Keypad

On the right end of the keyboard is the numeric keypad, with digits 0 through 9 and a decimal point. If you are accustomed to an adding machine or calculator, the numeric keypad lets you enter numbers quickly and accurately. The effect is identical to using the corresponding keys on the main keyboard.



Special Typing Keys

The following keys are also similar to typewriter keys. They modify the character keys or move the cursor.

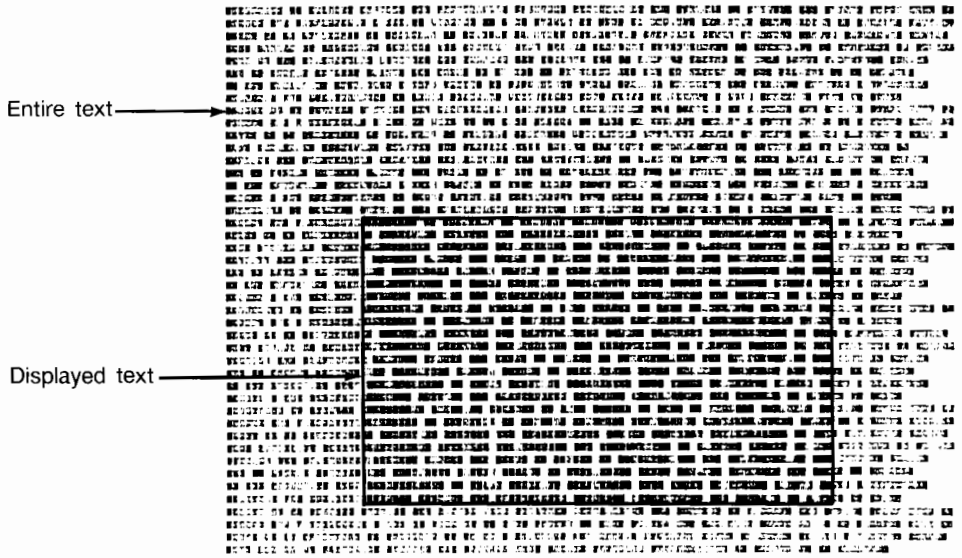


Key	Operation
Shift	Modifies the letter keys to produce uppercase letters; modifies other keys to produce the symbol (or command) on the upper half of a key.
Caps	Reverses lowercase and uppercase letters on the letter keys. Pressing Caps once causes the letter keys to produce uppercase letters without Shift, and to produce lowercase letters with Shift. This condition is noted by a *caps indicator*—a "C"—in the menu status block. Pressing Caps a second time returns the Shift key operation to normal, and deletes the caps indicator.
Extend char	Modifies the character keys to produce characters not shown on the keyboard. A map showing the location of the "extended" characters appears in appendix B, "Keyboard Maps," and a list of the entire Roman Extension character set appears in appendix C, "Character Set."
Tab	Moves the cursor to the next tab stop.
Shift Tab	Moves the cursor to the previous tab stop.
Return	Terminates a line of text and moves the cursor to the left end of the next line.
Back space	Moves the cursor left one column and writes a space there.
CTRL	Used in combination with other keys to send control characters to the computer. For more information, refer to appendix B.
ESC	Used in combination with other letters to send escape sequences to the computer. For more information, refer to "Alpha Windows" in chapter 8.
Shift DEL	Generates the DEL (delete) control character.

Editing Text

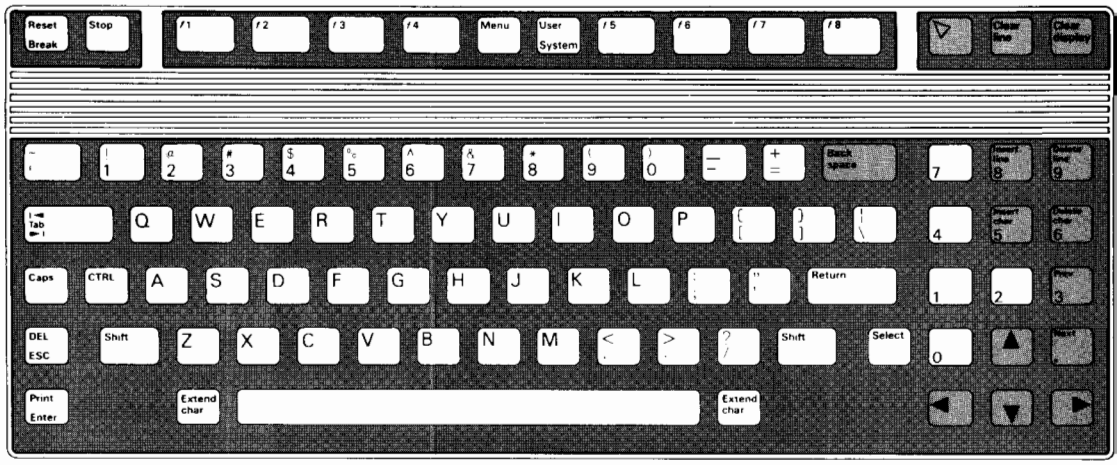
In most applications, after you've typed in your text, you can review, add, modify, or delete the text. This section describes such editing operations.

When you're working on a text file or a spreadsheet, you may not be able to see all of the text at once. In the following diagram, "entire text" indicates the block of text maintained by the application, and "displayed text" is how much of the entire text you can see in the window at one time.



In the following descriptions, to *move the text* means to relocate the text relative to the window, changing the part of the text that's displayed. Also, *current line* means the line containing the cursor.

Most of the editing operations listed below appear on the shifted numeric keypad.



Remember that an application can redefine any of these keys—be sure to read the application manual! For example, PAM redefines several of these keys.

Editing Operations

Operation	Key	Description
Cursor right.	(►)	Moves the cursor right one column.
Cursor left.	(◄)	Moves the cursor left one column.
Cursor up.	(▲)	Moves the cursor up one row.
Cursor down.	(▼)	Moves the cursor down one row.
Home up.	(►)	Moves the cursor to the upper-left corner of the entire text.
Home down.	(Shift)(►)	Moves the cursor to the lower-left corner of the entire text.
Roll left.	(Shift)(◄)	Moves the text left one column.
Roll right.	(Shift)(►)	Moves the text right one column.
Roll up.	(Shift)(▲)	Moves the text up one row.
Roll down.	(Shift)(▼)	Moves the text down one row.
Previous page.	(Shift)(Prev)	Moves the text down by the number of rows in the window.
Next page.	(Shift)(Next)	Moves the text up by the number of rows in the window.

Editing Operations (Continued)

Operation	Key	Description
Back space.	(Back space)	Moves the cursor left one column and writes a space there.
Delete character.	(Shift)(Delete char)	Deletes the character at the cursor position and moves the remaining characters left one column.
Insert line.	(Shift)(Insert line)	Inserts a blank line, moving the current line and all subsequent lines down one row. The cursor moves to the left end of the blank line.
Delete line.	(Shift)(Delete line)	Deletes the current line, moving all subsequent lines up one row. The cursor moves to the left end of the line that followed the deleted line.
Clear to line end.	(Clear line)	Clears text from the cursor to the end of the line.
Clear line.	(Shift)(Clear line)	Clears all text in the current line.
Clear to text end.	(Clear display)	Clears text from the cursor to the end of the entire text.
Clear text.	(Shift)(Clear display)	Clears the entire text.

Editing Operations (Continued)

Operation	Key	Description
Insert character.	(Shift)(Insert char)	Switches between Replace Character mode and Insert Character mode. The default mode is Replace Character mode, where a new character replaces the existing character at the cursor position. Pressing (Shift)(Insert char) switches to Insert Character mode, where a new character is inserted at the cursor position, moving the existing characters right one column. This condition is noted by an *insert indicator*—an "I"—in the menu status block. Pressing (Shift)(Insert char) again switches back to Replace Character mode.

4 Viewing and Printing Your Information

Contents

4 Viewing and Printing Your Information

This chapter describes how to display and print the contents of a *text file*, which is a data file consisting of words. You can also display and print the contents of other data files (such as spreadsheet data files), but the result would probably be unintelligible.

An example of a text file is the `README` file on the Utilities disc. The `README` file describes the contents of the disc. The instructions for displaying and printing a text file use this `README` file as an example.

This chapter also describes how to print a copy of the display. If you need information about connecting and using peripheral printers, refer to *Connecting Peripherals to Your Integral PC* and to chapter 10, "Installing and Using Peripherals."

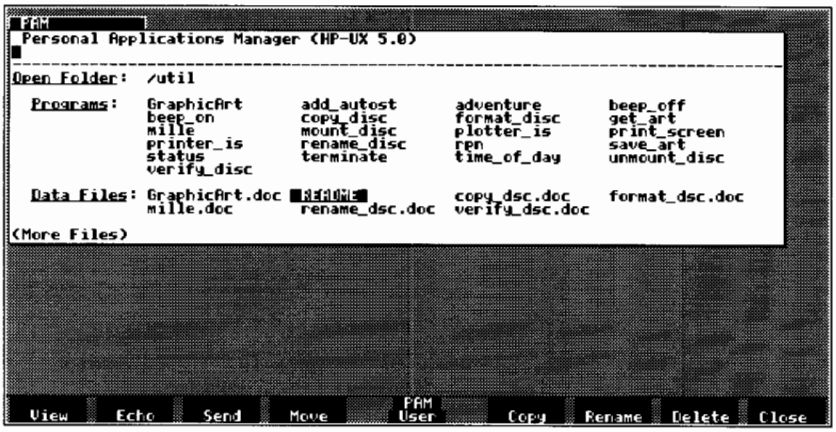
Viewing a Text File

If you want to follow the example that views a `README` file, insert the Utilities disc now.

To view a text file:

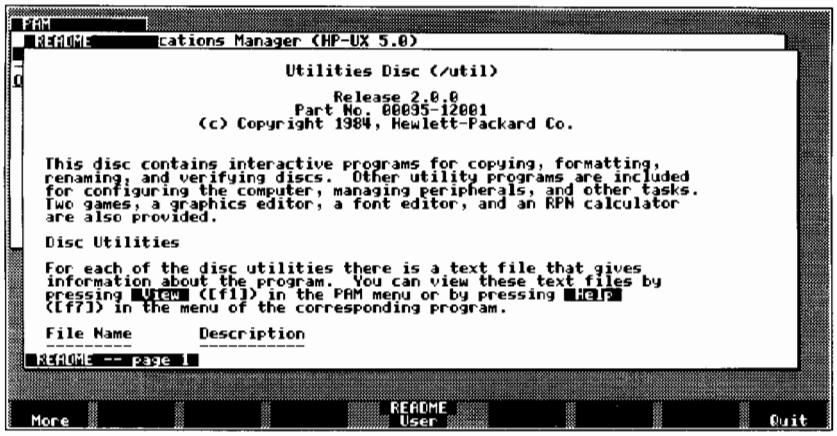
1. Select the PAM window.
2. Highlight the name of the text file.

   The display shows:

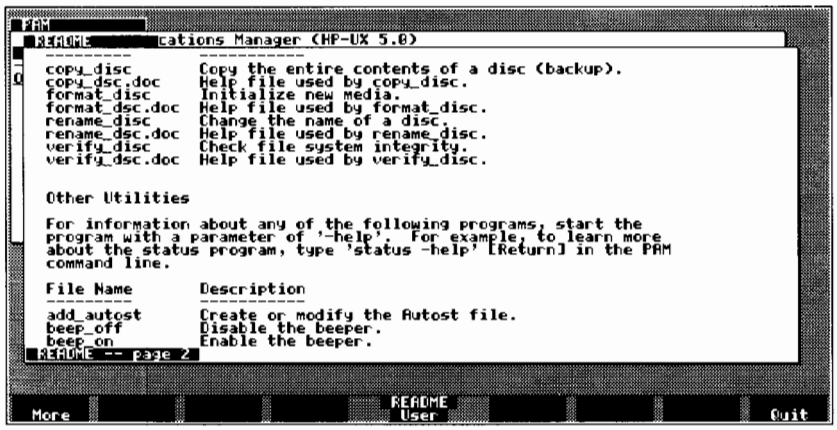


3. Press View (f1).

   PAM creates a new window with the same name as the text file, and then displays the first page of the text file.



4. Press More (f1) to show each succeeding page of the file. (More appears only if there is more text to display.)



5. Press Quit (f8) to end the view command and eliminate the window.

Additional Information. To view a data file, you can use one of these alternate methods:

- Highlight the name of the data file and press (Return).
- Enter the name of the data file on the command line and press (Return).
- Enter `view` *file_name* in the command line and press (Return).

The viewing window takes the same name as the file to be viewed except when the `view` command is explicitly used (then it's named `view`).

The `view` command should be used to display only data files or script files. Program files (that are not script files) and other files that do not contain text should not be displayed with `view`. If the file accessed does not contain readable text, the data it contains will be interpreted as control characters and other non-displayable characters.

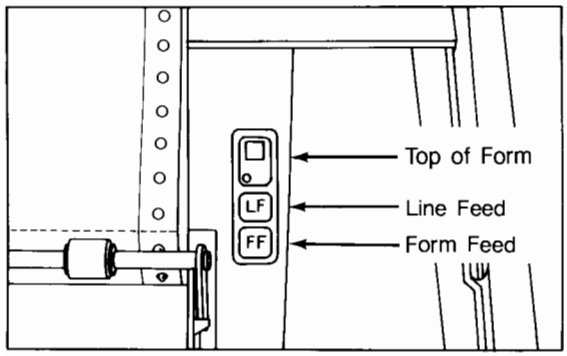
To view successive pages, you can press the space bar instead of More ((f1)).

To end the view process and eliminate the window, you can press (Q). This is convenient if the viewed file happens to cancel the User menu.

Positioning the Paper

The *Setup Guide* shows you how to load paper into the built-in printer. It also tells you to press the top, gray button on the printer to set the *top of form*, which is the position of the first printed line on each page. If there is paper in the printer when you switch on the computer, the current position of the paper becomes the top of form.

You can control the movement of paper through the printer by pressing the buttons on the printer control panel. The panel is located on top of the computer, to the right of the printer:



To advance the paper by one dot: Press (LF) quickly.

To advance the paper by one line: Press (LF) firmly.

To advance the paper by several lines: Press and hold (LF). The paper advances one line, pauses, and then advances until you release the button.

To advance the paper to the next page: Press (FF).

If the printer runs out of paper, it stops printing and blinks the out-of-paper light on the control panel. You should load more paper, position the paper, and press the top, gray button to set the top of form. (These steps are shown in the setup guide). The light then stops blinking and the printer resumes.

Printing a Copy of the Display

You can also print a copy of the contents of the display. You can do this only with the built-in printer or with a peripheral printer having raster graphics capability, such as the HP LaserJet printer.

To print a copy of the display: Press (Shift)(Print).

PAM creates a window for the `print` process. It's normally "hidden"—only the banner shows. When the process finishes, the window name is shown in parentheses: `(print_screen)`. Such windows are automatically removed when you create the next window—or you can select and remove the window as described in chapter 7.

Additional Information. Sometimes the information you want to print exceeds the size of the window. If you're printing a data file (or another "printable" file), you can print it directly (refer to the next topic). Otherwise, you can use (Shift)(Next) and (Shift)(Prev) to display successive "pages" for printing.

If you want to modify the printed image, you can use the `print_screen` utility on the Utilities disc. You can include optional parameters that specify several options, such as rotating the image, expanding the image, and printing an individual window from the display. Refer to `print_screen` in chapter 11 for detailed information about printing the display. For example, this command prints the display rotated and enlarged:

```
print_screen sideways expanded
```

This command prints the window `showme` without a frame:

```
print_screen window showme unframed
```

If you're using a peripheral printer with raster graphics capability, you'll have to designate it as the "active" printer. "Using Peripheral Printers" in chapter 10 describes how to do this using the `printer_is` command.

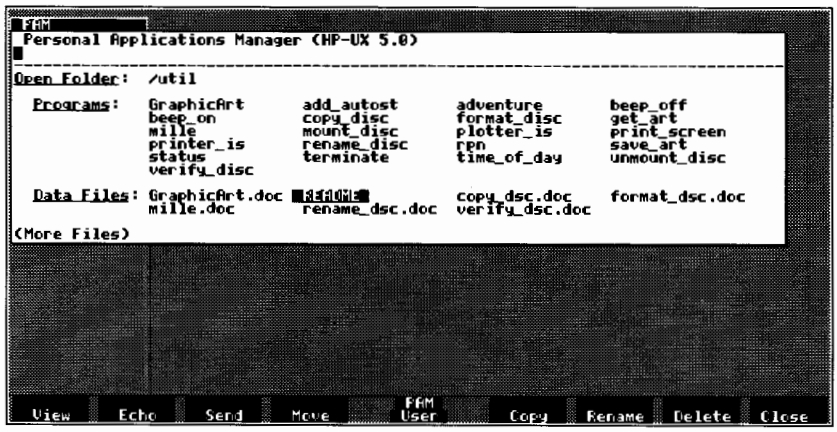
Printing a Text File

If you want to follow the example that prints a `README` file, insert the Utilities disc now.

To print a text file:

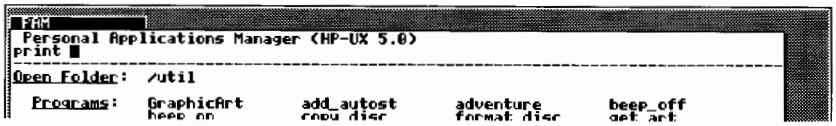
1. Select the PAM window.
2. Highlight the name of the text file.

   The display shows:



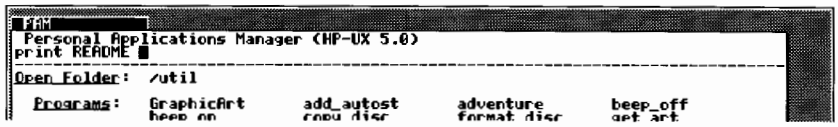
3. Type `print` followed by a space.

   The command line shows:



4. Press Echo ((f2)).

   This copies the name of the text file to the command line:



5. Press (Return).

   PAM creates a window for the `print` process. It's normally "hidden"—only the banner shows. When the process finishes, the window name is shown in parentheses: `(print)`. Such windows are automatically removed when you create the next window—or you can select and remove the window as described in chapter 7.

Additional Information. If you enter several file names after the `print` command, then one file starts printing, and the additional files are placed in the *printer waiting list* for the active printer. When the file being printed completes printing, another file in the printer waiting list begins printing. When several file names are specified with a single `print` command, the files are printed in the order that they're listed in the command line. If separate `print` commands are used, the files aren't necessarily printed in the order in which the commands were executed. The printer continues to operate until the printer waiting list is empty.

Each file to be printed must remain accessible to the system until it is finished printing. If all the discs containing the files to be printed cannot be included in the file system at the same time, then the root folder can be used for temporarily storing the files. (The root folder is discussed in chapter 8.) Use the `copy` command to place a copy of each file to be printed in the root folder before executing the `print` command. After printing is completed, you can remove these files from the root folder with `delete`.

Only files containing readable text should be printed. The printing of program files or other files that do not contain readable text can result in control characters and other non-printable characters being output.

If, for some reason, you wish to stop a file from printing, use one of the following methods:

- Type `stopprint` (Return). This stops the active `print` command.
- Select the print window, the press (Stop). You can stop any `print` command this way.

If you stop the active `print` command and another `print` is waiting to be processed, the paper moves to the next top-of-form position and printing begins.

The `print` command prints files using the active printer device. The `stopprint` command affects only the active printer—the (Stop) key can stop printing on any printer. You can change the active printer by using the `printer_is` utility—refer to chapter 11.

If you've stopped a `print` command and new `print` commands don't produce output, refer to `print` in chapter 11 for information about fixing the problem.

Printing Output

You can print many types of output besides the contents of a text file. The following suggestions are a preview of some of the ways you'll learn to use your printer—they include references to other parts of this manual for detailed information.

- To print output that normally would go to the display, you can use the "output redirection" symbol or the "pipe" symbol to send the output to the `print` process. ("Redirection" and "piping" are discussed in chapter 9.) The following example "pipes" to the active printer a list of files and folders for the folder shown in the PAM window. (The `ls` program must be accessible—either insert the Commands 1 disc or copy the `ls` file from that disc to computer memory as described in chapter 5.)

  ```
  ls | print
  ```

  This example "redirects" to the active printer the same information.

  ```
  ls >/dev/lp
  ```

- To set your printer for optional printing modes, you can use the `printer_is` utility. (This utility is described in chapter 11.) For example, this command sets the printer to compressed printing with eight lines per inch. (The `printer_is` file must be accessible—insert the Utilities disc.)

  ```
  printer_is compressed line_spacing 8
  ```

5 Protecting Your Information

Contents

5 Protecting Your Information

There are a few rules you should follow to protect your information stored on flexible discs. Following these rules prevents most problems and minimizes the damage if a problem does occur.

Protecting Your Discs From Damage

Observe the following precautions to protect the data on your discs:

- Always check that the colored "drive busy" light is off before you eject a disc.
- Keep the disc away from magnetic fields, such as those generated by permanent magnets, transformers, motors, telephones, and television sets.

Observe the following precautions to protect your flexible discs from physical damage:

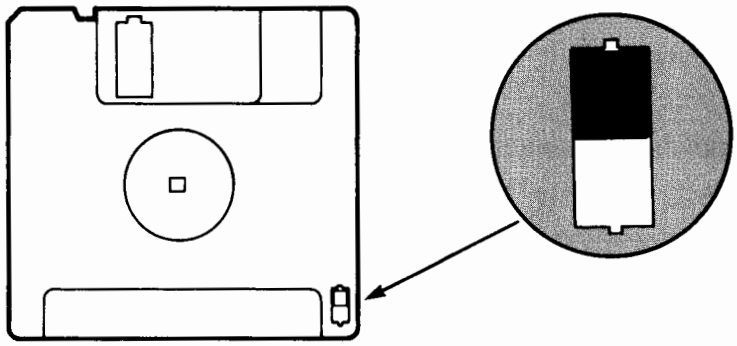
- Make sure the metal shutter is closed. Never touch the magnetic surface of the disc.
- Remove the disc from the built-in drive *before* you turn off the computer.
- If the disc won't be used for an extended period, remove it from the drive.
- Store the disc away from contaminants such as dust, smoke, ashes, eraser crumbs, bread crumbs, and coffee.
- Do not expose the disc to temperatures below −40°C (−40°F) or above 60°C (140°F).
- Do not store the disc at temperatures below 10°C (50°F).

Protecting Your Discs From Accidental Changes

If you don't expect to change the files on a flexible disc for a while, or have very important files on a disc, you can *write-protect* the disc. This allows you to read the files, but prevents you from changing the files or adding new files until you *write-enable* the disc again.

You write-protect or write-enable a disc by sliding its *write-protect tab*:

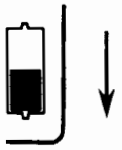
Write-Protect Tab



To write-protect a disc:

1. Locate the colored write-protect tab on the back of the disc.
2. Slide the tab down (toward the corner) as illustrated. Make sure it snaps into place.
3. Turn the disc over and make sure the tab isn't visible through the hole in the corner. (You can think of this as providing a hole for an imaginary padlock to protect your data.)

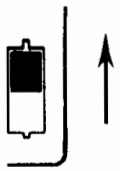
Write-Protected



To write-enable a disc:

1. Locate the colored write-protect tab on the back of the disc.
2. Slide the tab up (away from the corner) as illustrated. Make sure it snaps into place.
3. Turn the disc over and make sure the tab blocks the hole in the corner.

Write-Enabled



Recognizing When a Disc Is Worn Out

A disc will eventually wear out from contact with the disc drive. On each disc the computer keeps a record of the wear on that disc. If the disc is approaching the end of its useful life, the computer warns you as follows.

Note

If the computer clicks every second or so and blinks the colored "drive busy" light, copy the disc to a new disc as soon as possible and discard the old disc.

If you continue to use the worn out disc, you risk losing the data on that disc, and the computer will eventually refuse to write data to the disc.

If you want to verify the integrity of the data on a disc, you can use the `verify_disc` utility. Refer to `verify_disc` in chapter 11.

Making an Extra Copy of a Disc

Despite preventative measures, there is always a possibility of problems with your discs. For example, you might lose one! You can minimize the damage from any type of problem by making copies of your discs before the problem arises. Such extra copies are called *backup* discs.

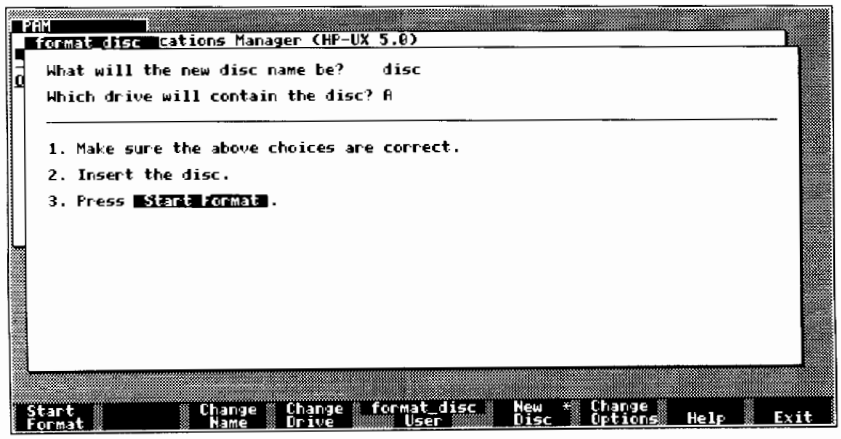
Preparing a New Disc

A new disc must be initialized and formatted before you can use it. If you've already prepared the disc, or if the disc already contains some files, you can skip ahead to "Copying an Entire Disc."

To prepare a new disc:

1. Select the PAM window.
2. Insert the Utilities disc.
3. Highlight and start the `format_disc` program.

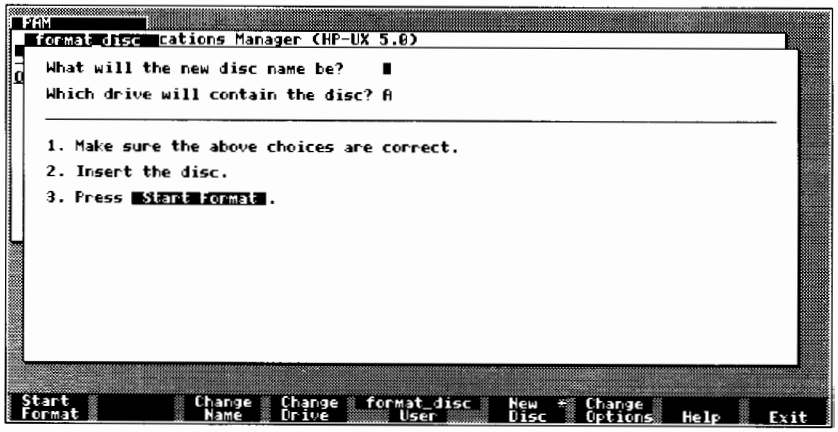
   PAM starts the `format_disc` program in a new window:



This display indicates that the program will format the disc inserted in the drive named "A" (the primary drive) and will name the formatted disc "disc". In this example, assume that you want to use the built-in drive, and you want to name the disc something more descriptive than "disc". (If you're going to copy an *entire* disc to this disc, don't bother choosing a name—the `copy_disc` utility will rename this disc to match the original.)

4. Press Change Name ((f3)).

   "disc" is replaced by a cursor:

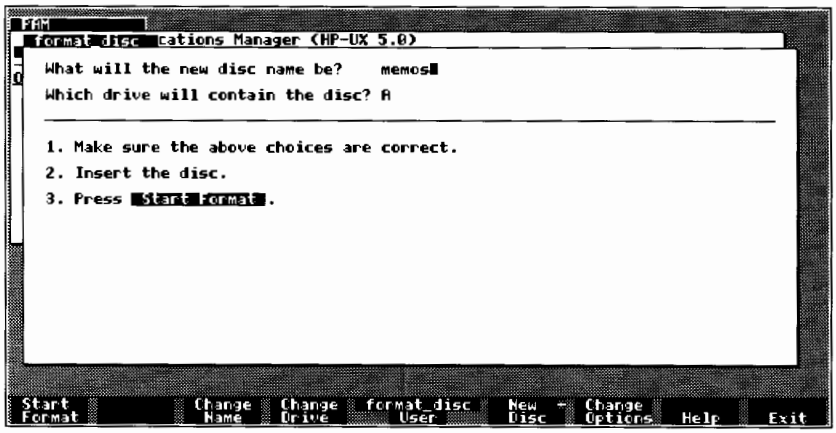


5. Type in the name you want.

   You can use up to six characters for a disc name, including uppercase letters, lowercase letters, and digits. You can also use percent (%), dash (-), underscore (_), plus (+), comma (,), and period (.); but don't use these for the first character.

   If you make a typing error, use the (Back space) key to correct it.

Suppose you want to name the disc "memos":



6. Press (Return).

The computer accepts your input and displays it as the specified name. Check that it is correct. If you made a typing mistake or if you change your mind, repeat steps 4, 5, and 6.

(You can change the specified drive in the same way. Refer to chapter 10, "Installing and Using Peripherals," for information about specifying peripheral disc drives.)

(If necessary, you can specify the interleave factor for the disc by pressing Change Options ((f6)). Usually the default value is adequate. Refer to `format_disc` in chapter 11 for more information.)

7. Insert the disc to be formatted into the specified drive.
The disc should be write-enabled (the colored tab blocks the hole in the corner).

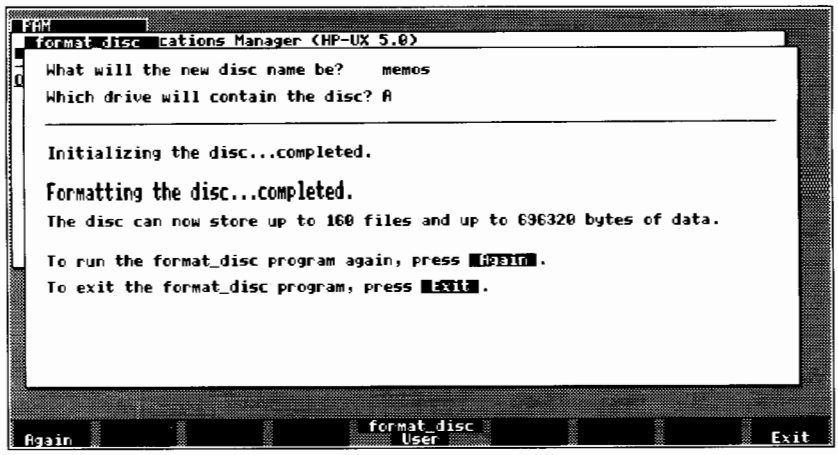
Caution 

Be sure that the disc in the specified drive is the disc you want to format. If you accidentally leave the Utilities disc in the specified drive, you will erase all files on the Utilities disc.

---

8. Press Start Format (f1).

   The program begins to format the disc. As it proceeds, it displays the status of the process. When it is done, the display shows:



   If you want to format another disc, press Again (f1), then return to step 4.

9. Press Exit (f8).

Additional Information: If you're using a peripheral disc drive, you don't have to "mount" the unformatted disc—just insert it in the drive when you're ready to format it. After it's formatted, you should "unmount" the disc as described under "Inserting and Removing a Disc" in chapter 10.

Copying an Entire Disc

You should now have a disc ready to use for the backup copy. This disc can be an old disc with files on it (which will be erased), or a new disc that you have formatted. When you copy discs, the original disc to be copied is called the *source* disc, and the disc to become the backup disc is called the *destination* disc. The source and destination discs must be the same size and type (single- or double-sided) discs.

The `copy_disc` program works by copying from the source disc directly to the destination disc *only if the discs are in separate drives*. If you use a single drive, the program copies from the source disc to computer memory, asks you to eject the source disc and insert the destination disc, and then copies from computer memory to the destination disc. When there's not enough memory to store the entire contents of a disc at once, the program repeats this procedure as many times as needed to copy the entire disc. The procedure described below uses a single drive.

Note 

You can minimize the number of times you swap the discs by maximizing the amount of computer memory available. To do so, stop any unneeded programs before starting the `copy_disc` program.

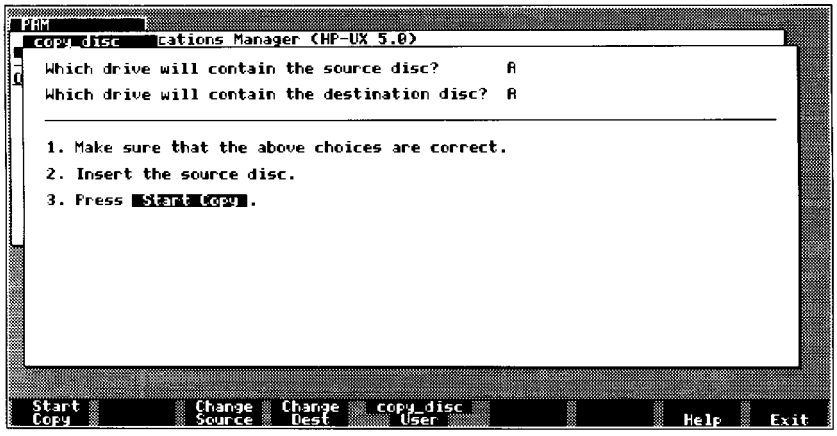
To copy a disc:

1. Make sure that the destination disc is write-enabled (the colored tab blocks the hole in the corner of the disc).
2. Select the PAM window.
3. Insert the Utilities disc.

   PAM updates its window to show the contents of the Utilities disc.

4. Highlight and start the `copy_disc` program.

   PAM starts the `copy_disc` program in a new window:

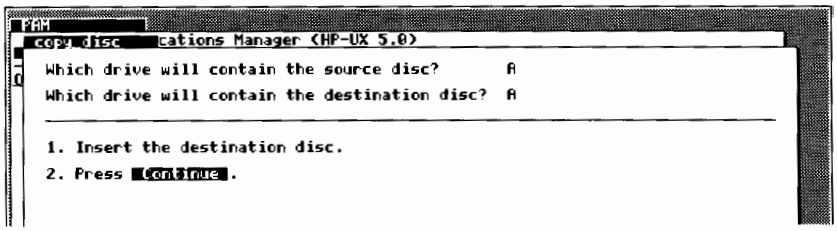


   This message indicates that the program will use the drive named "A" (the built-in drive) for copying the disc. In this example, assume that you do want to use the built-in drive. (You can change either of the specified drives by using the Change Source (f3) and Change Dest (f4) keys. Refer to chapter 10, "Installing and Using Peripherals," for information about specifying peripheral disc drives.)

5. Insert the source disc.

6. Press Start Copy ((f1)).

   The program checks how much computer memory is available, displays how many times you'll swap discs, and then copies from the source disc to computer memory. When there is no more room in memory, the program asks you to swap discs:

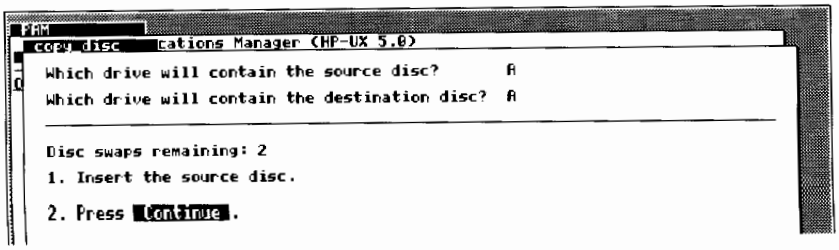


7. Eject the source disc and insert the destination disc.

   Remember that any old files on the destination disc will be erased.

8. Press Continue ((f1)).

   The program copies from memory to the destination disc and asks you to swap discs again:



9. Eject the destination disc and insert the source disc.

10. Press Continue (f1).

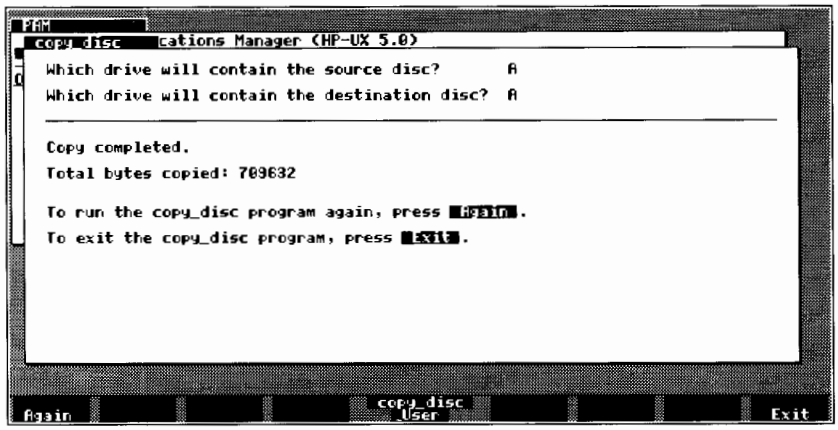
    The program again copies from the source disc to memory. When there is no more room in memory, the program asks you to swap discs.

11. Eject the source disc and insert the destination disc.

12. Press Continue (f1).

    The program copies from memory to the destination disc. If there is more to copy from the source disc, the program again asks you to swap discs. In this case, return to step 9.

    If the entire contents of the source disc have been copied, the display shows:



    If you want to copy another disc, press Again (f1), then return to step 5.

13. Press Exit (f8).

We recommend that you write-protect the original disc, store it in a safe place, and use it only to make new copies.

Additional Information. If you have two disc drives available, you can avoid swapping the source and destination discs. Refer to `copy_disc` in chapter 11.

Making a Copy of a File

You can make an extra copy of a file for backup or with the intention of changing the new copy. You can also copy a file into computer memory. Some examples:

- You have an important file on a disc that you don't want to write-protect. In this case you will make a copy of the file on a different disc, and the copy will have the same name as the original. You can then write-protect the disc containing the copy.
- You plan to edit a file, but you're not sure you'll like the result better than the original. In this case you will make a copy of the file on the same disc, and the copy will have a different name from the original. You can then change the original file, see how you like the result, and delete the version you don't like.
- You have no external disc drive for inserting a disc that contains a program you want to use. You want to use the built-in drive for another disc that the program will access. In this case you can copy the program into computer memory, and then run it while the other disc is inserted.

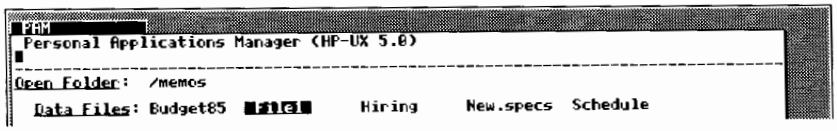
To make an extra copy of a file (on a different disc):

1. Select the PAM window.
2. Insert the disc containing the file to be copied.

   PAM updates its window to show the contents of the disc you inserted.

3. Highlight the name of the file you want to copy.

   If you're copying "File1", the display shows:



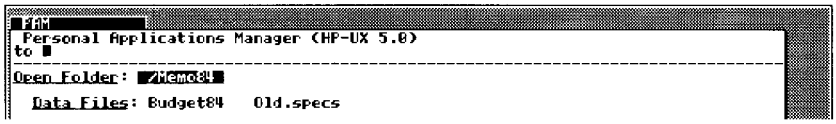
4. Press Copy ((f5)).

   PAM asks you to wait while it makes an internal copy of the file, and then displays:

5. Make sure that the *destination* disc, which will contain the new copy, is write-enabled (the colored tab blocks the hole in the corner).
6. Remove the original disc and insert the destination disc.

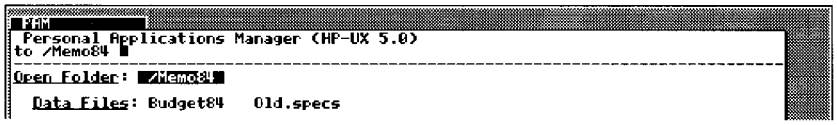
   PAM updates its window to show the contents of the destination disc.
7. Highlight the name of the open folder.

   If the disc's name is "Memo84", the display shows:



8. Press Echo ((f2)).

   This copies the name of the open folder to the command line:



9. Press (Return).

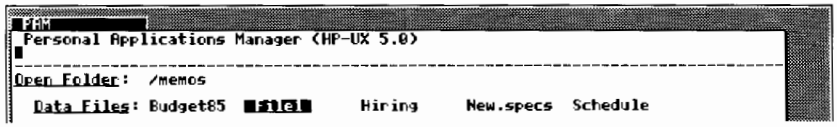
   PAM indicates when the copy has been completed, and then updates its window to include the new file in the contents of the destination disc:

If you have two disc drives available, you can avoid swapping the source and destination discs. Refer to `copy` in chapter 11.

To make an extra copy of a file (on the same disc):

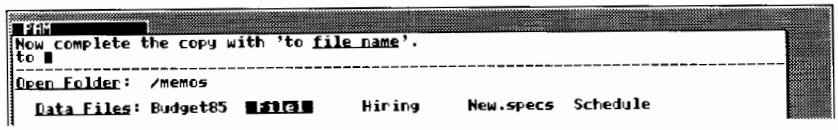
1. Select the PAM window.
2. Make sure that the disc is write-enabled (the colored tab blocks the hole in the corner).
3. Insert the disc.

   PAM updates its window to show the contents of the disc you inserted.
4. Highlight the name of the file you want to copy. If you're copying "File1", your display shows:



5. Press Copy ((f5)).

   PAM asks you to wait while it makes an internal copy of the file, and then displays:

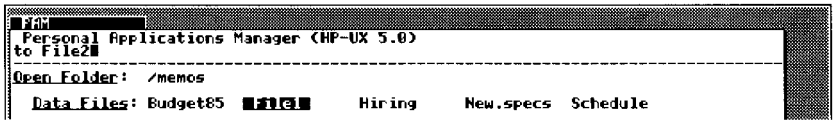


6. Type in the name you want. This name must be different from the original name.

   You can use up to 14 characters for file names, including uppercase letters, lowercase letters, and digits. You can also use percent (%), dash (-), underscore (_), plus (+), comma (,), and period (.); but don't use these for the first character.

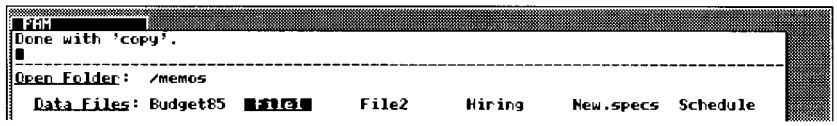
   If you make a typing error, use the (Back space) key to correct it.

   If the new file's name is "File2", the display shows:



7. Press (Return).

   PAM indicates when the copy has been completed, and then updates its window to include the new file in the open folder:



To make a copy of a file (in computer memory):

1. Select the PAM window.
2. Insert the disc containing the file to be copied.

   PAM updates its window to show the contents of the disc you inserted.
3. Highlight the name of the file you want to copy.

4. Press Copy ((f5)).

   PAM asks you to wait while it makes an internal copy of the file, and then prompts for the destination by displaying `to` in the command line.

5. Type / and press (Return). This specifies the "root" folder in computer memory (the electronic disc) as the destination.

   PAM indicates when the copy has been completed. The file isn't shown in the display because it's located in the root folder. (You can display the contents of the root folder by typing / (Return).)

6 Organizing Your Information

Contents

Chapter 6

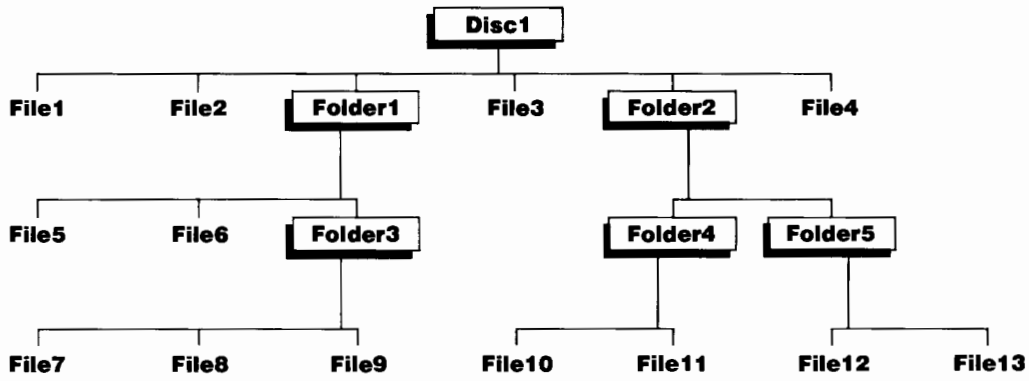
Organizing Your Information

6 Organizing Your Information

This section describes ways you can organize your information. The organization of files on your discs is much like the organization of files in a file cabinet. Each disc is like a drawer in the file cabinet, containing one or more folders, and each folder contains one or more files. In this chapter, "file" refers to either a data file or a program.

Folders are powerful because they can contain other folders as well as files. This leads to a hierarchy of folders: you can store a file in a folder that is stored in another folder, which you can store in another folder, and so on.

You can use a folder to collect related files, or related folders, or related files and folders. The following diagram shows an example of a disc's file structure:



The PAM window shows the contents of one folder at a time. That is, PAM shows only the file names and folder names in the *next level* of the structure. The table below lists each of the folders in the example above, along with the contents displayed in the PAM window when that folder is the open folder.

Folders and Contents

When the open folder is:	PAM displays the contents:
Disc1	File1, File2, File3, File4, Folder1, Folder2
Folder1	File5, File6, Folder3
Folder2	Folder4, Folder5
Folder3	File7, File8, File9
Folder4	File10, File11
Folder5	File12, File13

The open folder is identified in the PAM window by an extended name called a *path name*. Each folder's path name defines a path from the disc to that folder. For example:

Disc1 is `/Disc1`
Folder2 is `/Disc1/Folder2`
Folder4 is `/Disc1/Folder2/Folder4`

Files also have path names. A file's path name is simply the file name appended to the path name of the folder that contains the file. For example:

File10 is `/Disc1/Folder2/Folder4/File10`
File11 is `/Disc1/Folder2/Folder4/File11`

The slashes / are required delimiters in path names. The first slash—which can be used by itself—identifies a folder called the *root*, which is the master folder. If each disc is like a file drawer, the root is like the whole file cabinet. A file's path name actually defines a path from the root to the file.

The instructions in part I of this manual don't involve path names or the root folder, but you might encounter these terms in your applications. For more information about the path names and the file system, refer to "File System" and "File Structure" in chapter 8.

Finding a File

Finding a file on a disc involves finding its folder, and this involves moving around in the file structure. PAM puts you at the outermost level when you first insert a disc—unless you already have another folder open. You can then move inward, opening folder after folder, until you find the file you want. Taking the illustration above as an example, suppose you wanted File9. You would first insert Disc1, then open Folder1, and then open Folder3. Once you've opened Folder3 you can easily access File9.

You can move between any two folders on the same disc by closing folders until you're back to the outermost level (the disc) and then opening folders until you reach the one you want. In the example above, this is how you move from Folder3 to Folder4: from Folder3 to Folder1 to Disc1, and then from Disc1 to Folder2 to Folder4.

But it's not always necessary to go all the way back to the outermost level. For example, you can move from Folder4 to Folder5 by moving from Folder4 to Folder2, and then from Folder2 to Folder5.

Opening a Folder

To open a folder and view its contents:

1. Select the PAM window.
2. Highlight the name of the desired folder. (Folders are listed separately in the PAM window.)

   If there are more files and folders than PAM can show in the window, you can press (Shift)(Next) and (Shift)(Prev) to see the next and previous "pages" of contents.
3. Press Open ((f1)).

   PAM opens the folder and updates its window to show the contents of the folder.

Additional Information. To open an folder, you can use one of these alternative methods:

- Highlight the folder name and press (Return).
- Enter the full path name of the folder on the command line and press (Return). You can directly open any folder this way.
- Use the `cd` command (refer to chapter 11).

Closing the Open Folder

To close the open folder:

1. Select the PAM window.
2. Press Close ((f8)).

   PAM closes the folder and updates its window to show the contents of the next outer folder.

Additional Information. You can also type `close` and press (Return) to close a folder, or you can use the `cd ..` (dot dot) command (refer to `cd` in chapter 11).

Renaming a File or Folder

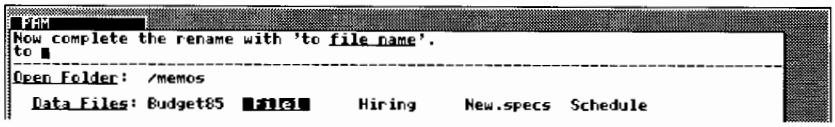
You might want to change the name of a file in the following situations:

- An application has created a file with a default name like "workfile", and you want to give the file a meaningful name.
- You have a series of existing files that you want to name consistently, like "abc.memo1", "abc.memo2", and "abc.memo3".
- You want to move a file named "abc" to another folder, but there is already a file named "abc" in that folder. If you don't rename one of the files, the file "abc" that you move into the folder will replace the file "abc" already in the folder—the old file is lost.

To rename a file or folder:

1. Select the PAM window.
2. Make sure the disc containing the file is write-enabled (the colored tab blocks the hole in the corner).
3. Insert the disc.
4. Highlight the name of the file to be renamed.
5. Press Rename (f6).

   If you're renaming a file named "File1", the PAM window shows:

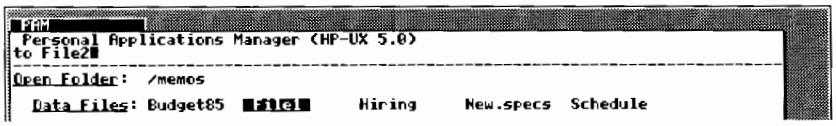


6. Type in the name you want.

   You can use up to 14 characters for the file name, including uppercase letters, lowercase letters, and digits. You can also use percent (%), dash (-), underscore (_), plus (+), comma (,), and period (.); but don't use these for the first character.

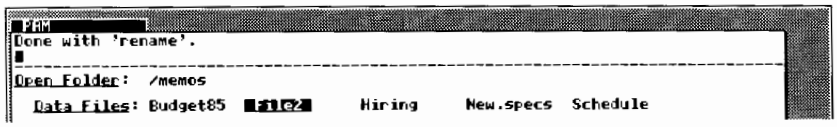
   If you make a typing error, use the (Back space) key to correct it.

   If you want the new name to be "File2", the PAM window shows:



7. Press (Return).

   PAM renames the file and updates its window to show the new name.



Additional Information. You can also use the `rename` command to rename a file or folder. Refer to `rename` in chapter 11.

Renaming a Disc

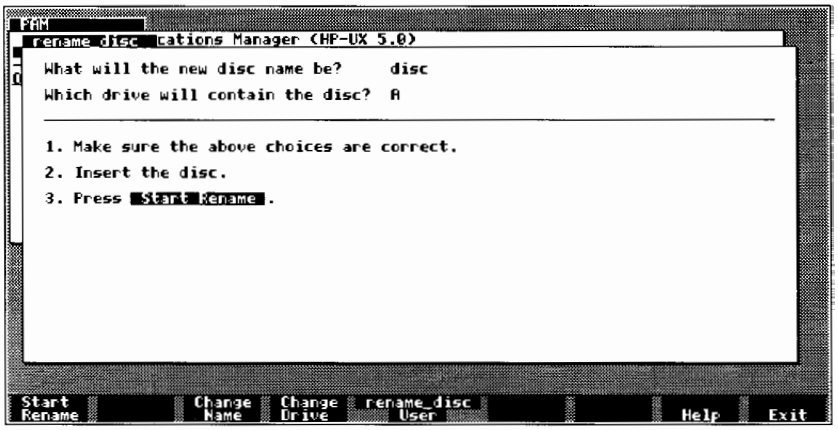
To rename a disc:

1. Select the PAM window.
2. Insert the Utilities disc.

   PAM updates its window to show the contents of the Utilities disc.

3. Highlight and start the `rename_disc` program.

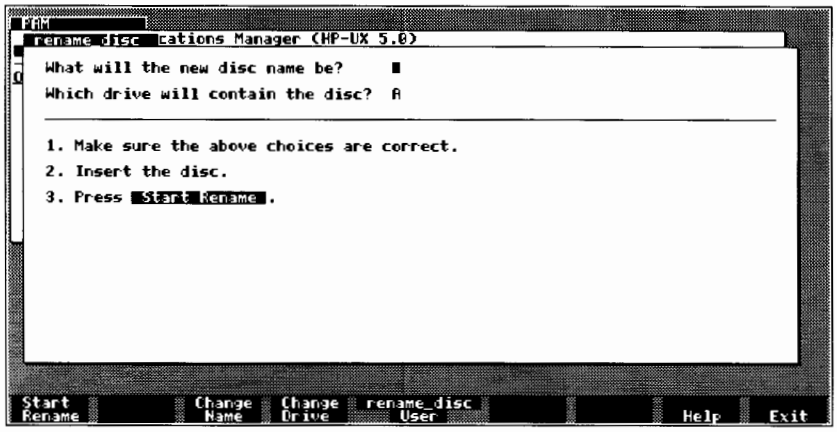
   PAM starts the `rename_disc` program in a new window:



This message indicates that the program will rename the disc inserted in the drive named "A" (the built-in drive) and will rename the disc "disc". In this example, assume that you want to use the built-in drive, and you want to rename the disc something more descriptive than "disc".

4. Press Change Name (f3).

   "disc" is replaced by a cursor:

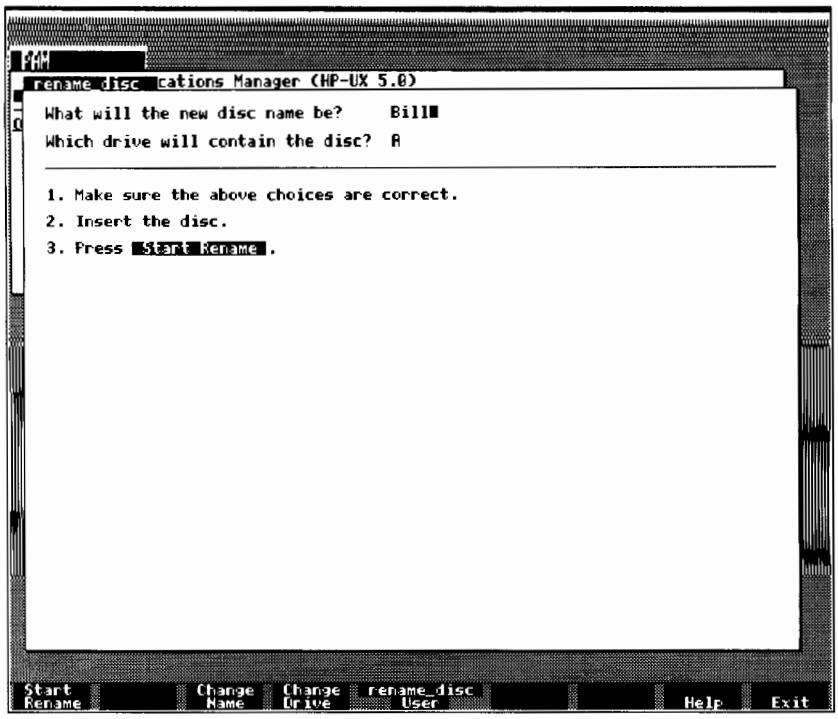


5. Type in the name you want.

   You can use up to six characters, including uppercase letters, lowercase letters, and digits. You can also use percent (%), dash (-), underscore (_), plus (+), comma (,), and period (.); but don't use these for the first character.

   If you make a typing error, use the Back space key to correct it.

Suppose you want to rename the disc "Bill":



6. Press (Return).

   The computer accepts your input and displays it as the specified name. Check that it is correct. If you made a typing mistake or change your mind, repeat steps 4, 5, and 6.

   (You can change the specified drive in the same way. Refer to chapter 10, "Installing and Using Peripherals," for information about specifying peripheral disc drives.)

7. Make sure that the disc to be renamed is write-enabled (the colored tab blocks the hole in the corner of the disc).

8. Insert the disc.

9. Press Start Rename ((f1)).

   The program renames the disc and indicates when it's done. If you want to rename another disc, press Again ((f1)), then return to step 4.

10. Press Exit ((f8)).

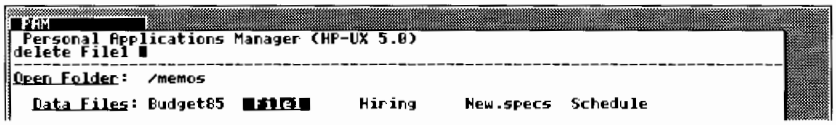
Additional Information. If you have two disc drives available, you can avoid swapping the Utilities and data discs. Refer to `rename_disc` in chapter 11.

Deleting a File

To delete a file:

1. Select the PAM window.
2. Make sure the disc containing the file is write-enabled (the colored tab blocks the hole in the corner).
3. Insert the disc.
4. Highlight the name of the file you want to delete.
5. Press Delete ((f7)).

   If you're deleting a file named "File1", the PAM window shows:



6. Press (Return).

   PAM deletes the file and updates the PAM window.

Note

You can't recover files once you've deleted them.

Additional Information. You can also delete files using the `delete` command. Refer to `delete` in chapter 11.

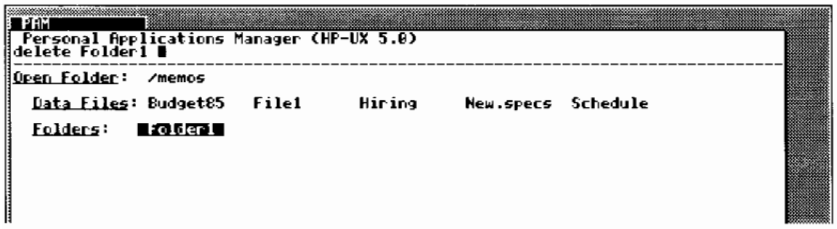
Deleting a Folder

To delete a folder:

1. Select the PAM window.
2. Make sure the disc containing the folder is write-enabled (the colored tab blocks the hole in the corner).
3. Insert the disc.

4. Open the folder that you want to delete.
5. Delete or move any files that are in the folder.
   (Instructions for moving a file appear in the next topic.)
6. Close the empty folder.

   PAM displays the contents of the next outer folder, which includes the empty folder to be deleted.
7. Highlight the name of the folder to be deleted.
8. Press Delete (f7).
   If you're deleting a folder named "Folder1", the PAM window shows:



9. Press Return.
   PAM deletes the empty folder and updates the PAM window.

Additional Information. You can also delete folders using the `delete` command. Refer to `delete` in chapter 11.

If a folder appears empty in the PAM window but a message indicates that it is not empty when you attempt to delete it, it probably contains a hidden file or hidden folder (one whose name begins with "."). Use the `ls -a` command to list all files and folders in the folder. Refer to `ls` in chapter 11.

Caution

Do not attempt to delete files or folders created for use by the system (`rom` and `dev` for example). It is possible to delete files and folders that are necessary to the functioning of the computer. The folders `dev` and `tmp` are examples of system folders stored in the root folder (/). If you delete these folders, or any of the files they contain, the computer will fail to operate properly when certain operations are performed. System files and folders are recreated from internal storage at each power on by the operating system. If you accidentally delete one of these files or folders, you should turn the computer off then on again to restart the operating system.

Collecting Files Into a Folder

Folders help you organize your files by grouping them together. Suppose you're working on three projects, and you've created several spreadsheet files for each project. First create three folders, one for each project. Then move the files for each project into the folder for that project. If you later need another file for one of the projects, create the new file within the proper folder.

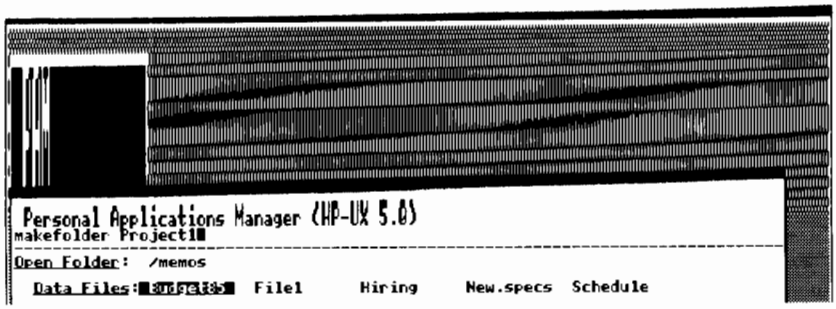
Creating a New Folder

To create a new folder:

1. Select the PAM window.
2. Make sure the disc to contain the new folder is write-enabled (the colored tab blocks the hole in the corner).
3. Insert the disc.
4. Open the folder that's to contain the new folder.
5. Type `makefolder` followed by a space.
6. Type in the name you want.

You can use up to 14 characters, including uppercase letters, lowercase letters, and digits. You can also use percent (%), dash (-), underscore (_), plus (+), comma (,), and period (.); but don't use these for the first character.

If you're naming the new folder "Project1", the PAM window shows:



7. Press (Return).

   PAM creates the folder and updates the PAM window to show the new folder.

Additional Information. To find other ways of using this command, refer to `makefolder` in chapter 11.

Moving a File or Folder

You can move a file to any folder on the same disc. You can also move a folder, provided you don't introduce circularity. (That is, you can't move Folder1 to Folder2 if Folder1 contains Folder2.)

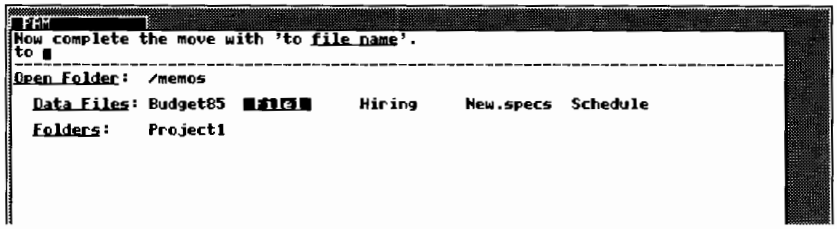
The procedure for moving a folder is identical to that for moving a file. The following example moves a file.

To move a file or folder:

1. Select the PAM window.
2. Make sure the disc containing the file is write-enabled (the colored tab blocks the hole in the corner).
3. Insert the disc.
4. Open the folder containing the file to be moved.
5. Highlight the name of the file to be moved.

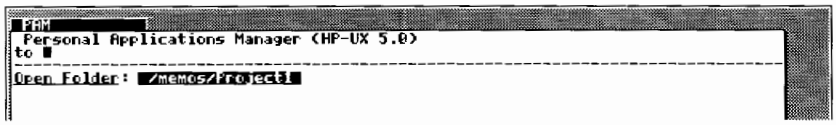
6. Press Move (f4).

   If you're moving a file named "File1", the PAM window shows:



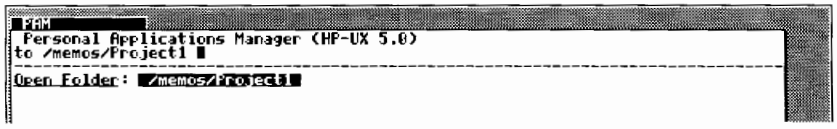
7. Open the folder to which you want to move the file.
8. Highlight the name of the open folder.

   If you're moving "File1" to folder "Project1", the PAM window shows:



9. Press Echo (f2).

   This copies the highlighted name of the open folder to the command line:



10. Press Return.

    PAM indicates when the move is complete and updates the PAM window to show the file moved into the folder.

Additional Information. You can change the file name when the file is moved, and you can move more than one file at a time. Refer to `move` in chapter 11.

If you want to move a file, but the file and the destination folder are on separate discs (and can't be mounted simultaneously), it's necessary to use a temporary storage area while the discs are exchanged. First, move the files to the root folder using

```
move file_name /
```

Then exchange discs, open the root folder, and move the files to the destination folder (using its full path name)

```
move file_name folder_name
```

You can move the entire contents of a folder in the same way as moving a single file, except that the destination *must* be on the same disc as the original folder.

Collecting Files Onto One Disc

You can minimize how often you remove and insert discs by collecting the files you use most onto one disc, which is called a *working disc*. In this case you're grouping your files by usage rather than by topic.

To collect files onto one disc:

1. If you have a brand new disc, prepare it according to the instructions under "Preparing a New Disc" in chapter 5.
2. Create the desired folders.

   Plan the arrangement of folders by drawing a diagram similar to the one on page 6-1. Then create the folders, starting with the outermost folder, according to the instructions under "Creating a New Folder" in this chapter.

3. Copy your existing files into the new folders, following the instructions under "Making a Copy of a File" in chapter 5.

Additional Information. As an alternative, in step 3 you can *move* (rather than *copy*) your files into the new folders, following the instructions under "Moving a File or Folder" in this chapter.

7 Managing Several Programs at Once

Contents

7 Managing Several Programs at Once

Your computer is multi-tasking, which means you can give it several tasks to perform at the same time. The computer divides its resources among the various programs that are running. You can start tasks that don't require your participation, such printing a file or sending a file to a remote computer, and then start an interactive program. This chapter tells how you can keep track of several programs running at once.

The commands used in this chapter are found on the System menu. If you don't remember how to switch to the System menu, refer to "Controlling the Menu" in chapter 2.

Shuffling and Selecting Windows

Each program executes in its own window. You can choose the program that you'll interact with by *shuffling* through the windows until the desired one is the active window, or by directly *selecting* a window as the active window.

To shuffle through the windows: Press (Shift)(Select) until the desired window is in front. The name of the active window appears in the status block at the center of the menu.

Each time you press (Shift)(Select) the back-most window moves to the front and becomes the active window (unless this isn't allowed by the program, such as for the `time_of_day` program). This is much like leafing through a pile of papers on your desk top.

Moving a window to the front doesn't change its contents—it just changes its visibility relative to the other windows.

To select a window:

1. Move the display pointer onto a portion of the desired window, which may be partially hidden behind other windows. (If necessary, refer to "Introducing the Display Pointer" in chapter 1.)
2. Press (Select).

   The selected window immediately moves to the front and becomes the active window.

Moving and Stretching a Window

You can control the placement and size of your windows by moving and stretching them. For example, you might want to compare the contents of overlapping windows, or monitor the output from two programs. You could see both windows at once by reducing their size and then moving them so they don't overlap. (Some programs create windows that can't be moved or stretched.)

To move a window:

1. Select the window you want to move.
2. Switch to the System menu.
3. Press Move ((f1)).

   The display pointer becomes a right-angle marker at the upper-left corner of the window.

4. Move the display pointer to the desired location for the upper-left corner.

   If part of the window moves off the screen, the window shape will change accordingly.

5. Press (Select).

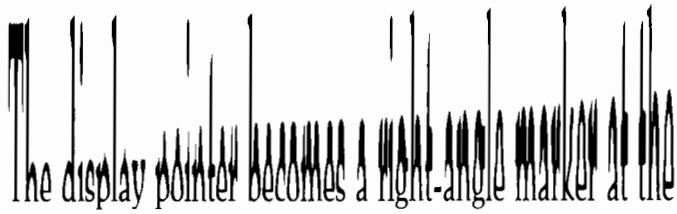
   The window moves to the indicated position, and the display pointer reappears at its previous position.

6. Switch back to the User menu.

If you decide not to move a window after pressing Move, press any other key to cancel the move operation. However, the key you press performs its normal function, too.

To stretch a window:

1. Select the window you want to stretch.
2. Switch to the System menu.
3. Press Stretch (f2).



lower-right corner of the window.

4. Move the display pointer to the desired location.
5. Press Select.

   The window grows or shrinks to stretch from its previous upper-left corner to the indicated lower-right corner, and the display pointer reappears in its previous location.
6. Switch back to the User menu.

If the information in the window extends beyond the window boundary, you can scroll the contents by pressing Shift and the arrow keys.

To restore a window to its default size and location:

1. Select the window as the active window.
2. Press and hold CTRL and Shift, and press Menu.

Additional Information. You can move and stretch a window using the window border. Refer to "Window Borders" in chapter 8.

If you have two or more programs running simultaneously, you may observe the windows "flashing." You can eliminate this by moving the windows so they don't overlap or by hiding one or more windows.

Hiding and Showing a Window

If you have several programs running, you might not want all the windows displayed. Suppose you're shuffling back and forth between two windows, and you have to shuffle several windows each time you want to select the other window. You can hide the windows that you don't want, without affecting the programs running in those windows—their contents aren't affected by hiding them.

To hide a window:

1. Select the window you want to hide.
2. Switch to the System menu.
3. Press Hide (f7).

   The window disappears except for its banner, which is placed in the lower-left corner of the display.
4. Switch back to the User menu.

To show a hidden window:

1. Move the display pointer to the banner of the window you want to show.

   If there are windows covering the lower-left corner of the display, move or hide those windows until you can find the desired banner.
2. Press Select.

   The window reappears as the active window. Its contents are the same as before (unless a program has changed its contents).

Additional Information. You can hide a window using the window border. Refer to "Window Borders" in chapter 8.

Suspending and Resuming a Window

If you want to hide a window for a while, but you don't want to miss any output from the program running in the window, you can prevent the program from sending output to its window. Doing so is called *suspending* the window. The program will keep running until it would normally display output in the window. If the program accumulates too much undisplayed output, it will stop running until you *resume* the window. No information is lost by suspending and resuming a window.

To suspend a window:

1. Select the window of the program you want to suspend.
2. Switch to the System menu.
3. Press Pause (f6).

   All activity in the window stops (although the program is probably still running). Note that an asterisk appears in the Pause label (Pause*).
4. Switch back to the User menu.

To resume a suspended window:

1. Select the window you want to resume.
2. Switch to the System menu.
3. Press Pause (f6).

   The asterisk in the Pause label disappears, and the program resumes output to its window.
4. Switch back to the User menu.

Saving or Removing a Window

Each window uses a portion of system memory (usually 10K bytes or more). Each running program affects the computer's speed. It's often helpful to remove windows that aren't necessary so that system resources are available for current tasks. The system tries to do this automatically—whenever you start a new program, the system removes unused windows, then creates a new window for running the program.

On the other hand, you may want to save the results of a program that has completed. You don't want its window to be removed yet. You can protect a window by *saving* it.

You can remove a window by stopping its program.

To save a window:

1. Select the window you want to save.
2. Switch to the System menu.
3. Press Save (f4).

   Note that an asterisk appears in the Save label (Save*).
4. Switch back to the User menu.

To "unsave" a saved window:

1. Select the window you want to "unsave."
2. Switch to the System menu.
3. Press Save (f4).

   The asterisk in the Save label disappears, indicating that the window can be removed.
4. Switch back to the User menu.

To remove a window:

1. Select the window you want to remove.
2. If a program is running, try to end it using a method provided by the program itself. Often, a program has a Quit, Exit, or Save command in its User menu for ending the program. Refer to the documentation for the program. This may also remove the window.
3. Press (Stop) or press Stop ((f8) in the System menu) to remove the window (and stop the program).

   If this doesn't remove the window, press (Shift)(Stop). Doing so risks losing program data and leaving extraneous files in the electronic disc, so try (Stop) first.

Additional Information. The following list describes keystrokes you can use to stop a program that doesn't have an exit procedure.

Keystroke	Description
(Stop) *or* Stop	Signals the program using the active window to stop. The window is not removed until the program has stopped. With application programs, this may have the same result as the user exit procedure. That is, the program might not stop immediately. (Sends a "quit" signal to the program.)
(Shift)(Stop)	Forces the program using the active window to stop. The window is removed after the program stops. Normal exiting procedures will not be allowed to complete; the program stops immediately. (Sends a "kill" signal to the program.)
(Shift)(Reset)	Stops all programs and removes all windows. Any information in memory that's not stored in the electronic disc is lost.

Note

If you stop a program, you may cause disc activity that's part of the normal exiting procedure for the program. Disc activity is indicated by the colored disc-drive light. You should delay taking additional steps to remove a window until related disc activity stops.

Another option for stopping a specific program is to use the `terminate` utility (refer to chapter 11). This utility stops a program but leaves its window on the screen. After the program has stopped executing, you can use (Stop) or **Stop** ((f8) in the System menu) to remove the window.

You can stop and remove a window using the window border. Refer to "Window Borders" in chapter 8.

8 System Description

Contents

Chapter 8 System Description

8 System Description

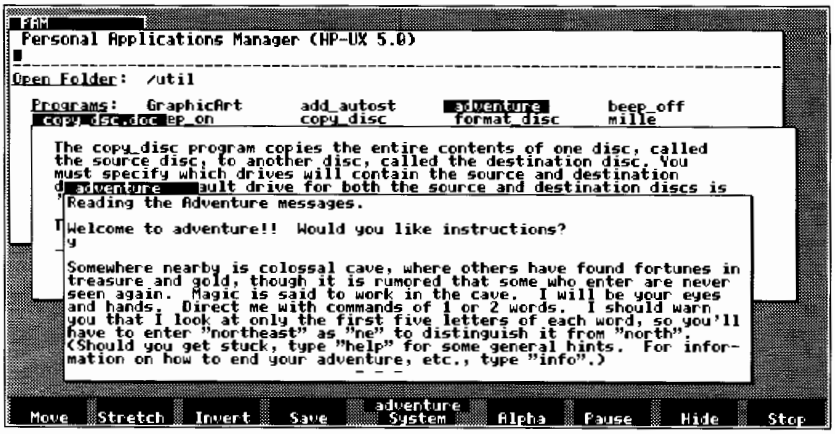
This chapter describes key aspects of the computer's operation. Some of the concepts are introduced in part I, "Fundamentals." Other concepts, such as PAM and individual commands, are described in greater detail in chapters 9 and 11.

Windows and Multitasking

A striking difference between this computer and other computers is its multiple window capability. Each window is an area in the display through which you communicate to a program (including PAM).

The computer's powerful operating system allows more than one program to run at one time. This is known as *multitasking*. When you run multiple applications, one or more windows may be displayed for each application. The following figure shows a typical display having multiple windows.

Display with Multiple Windows



Each window maintains the information displayed by its associated program. Programs run independently and are visually separated on the display.

The keyboard and optional mouse are connected to only one window (or program) at a time. This window is the *active window*. The active window is identified in the status block at the middle of the menu.

You can manipulate windows in various ways: moving them, stretching them, even hiding them. Chapter 7 explains how to control windows and their contents.

Each window contains a *banner* at the top that shows the window name. Many commands and utility programs create windows for their processes to run in—the name for such a window is usually the same as the process name. When the process finishes, the banner usually indicates this by showing the window name in parentheses—such windows are removed whenever the next window is created.

System Keys

The keyboard contains keys that enable you to control computer operations manually. The system keys are:

Key	Operation
Break	Sends a break to the host computer in data communications.
Shift Reset	Stops all programs and removes all windows, then displays PAM window.
Stop	Stops a program and removes its window. (The program might not stop immediately.)
Shift Stop	Abruptly stops (kills) a program and removes its window without allowing operations to complete.
Menu	Turns the function or system key labels off and on. Even though the labels are not displayed, the current function or system keys are still active. The right button on the mouse corresponds to this key.
Shift Menu	Turns the optional border on and off.
CRTL Shift Menu	Returns a window to its default size and location.
System	Toggles between the System and User menus.
Shift User	Selects the User menu.
Select	Selects the choice indicated by the display pointer. The left button on the mouse corresponds to this key.
Shift Select	Shuffles windows on the display screen.

Key	Operation
Enter	Sends the line that the cursor is on to the application. Also used in digitize mode of a graphics window.
Shift Print	Prints an exact copy of the screen image on the active printer.
CTRL Select	Displays a window that isn't shown on the screen.
CTRL ◄ CTRL ►	Moves the display pointer left or right, respectively.
CTRL ▲ CTRL ▼	Moves the display pointer up or down, respectively.
CTRL ▼ ►	Moves the display pointer diagonally toward the lower-right corner.
CTRL ▼ ◄	Moves the display pointer diagonally toward the lower-left corner.
CTRL ▲ ◄	Moves the display pointer diagonally toward the upper-left corner.
CTRL ▲ ►	Moves the display pointer diagonally toward the upper-right corner.

Menus

The computer provides labels at the bottom of the display that define the current functions of the eight function keys (f1 through f8). Three menus are available to you:

- System menu. This menu contains a fixed set of the most common commands for window control. The System menu is available in any window.

- User menu. This menu contains a set of commands that's designed specifically for the program running in the active window. The User menu is available in any window, but the functions vary from window to window. For example, the PAM menu differs from a text processor menu.
- Alpha or Graphics menu. This menu contains one of two sets of commands that depends upon the type of the active window, as described under "Types of Windows" later in this chapter. For an alpha window, the Alpha menu is available. For a graphics window, the Graphics menu is available.

You can toggle between the System and User menus by pressing the (System) key. (The (Shift)(User) keystroke selects the User menu.) These are the main menus that you'll use.

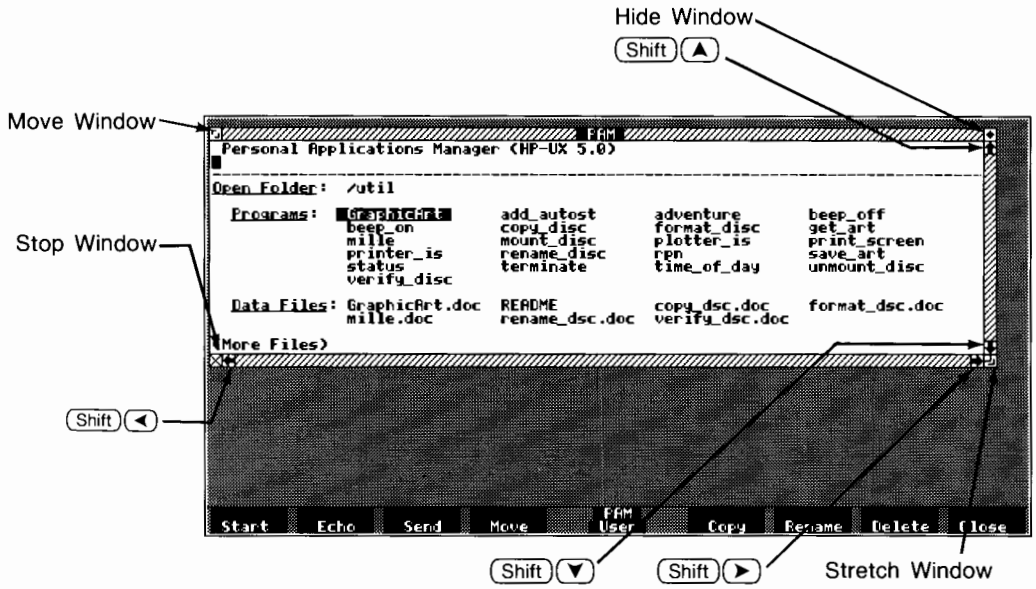
You can activate the Alpha or Graphics menu by pressing (f5) in the System menu. The Alpha and Graphics functions are described under "Types of Windows" later in this chapter.

Window Borders

Normally, the computer shows each window inside a simple outline. However, you can activate an optional *border* that provides easier operation while using the display pointer. Borders are especially useful if you're using a mouse.

To turn the border on or off, press (Shift)(Menu). This toggles the border for only the active window.

While the optional border is active, you can point to certain areas of the border with the display pointer, then press Select to perform the corresponding operation—you can do this easily with a mouse. The following diagram points out the border areas and their corresponding operations.



The "Move" and "Stretch" operations each produce a marker that you can move to a new location. Press Select to choose that location. (Refer to "Moving and Stretching a Window" in chapter 7.)

The "Stop" operation is equivalent to the Stop key. Some programs may not respond to that operation, as discussed in "Saving and Removing Windows" in chapter 7.

If you create a window using an operation in the PAM window, that new window is created with a border if the PAM window has one at that time.

PAM

PAM—the Personal Applications Manager—is the program that you're probably most familiar with by now. PAM manages all aspects of your interaction with the computer, including running application programs. (Actually, PAM is an application program itself.)

Because PAM is your main interface to the computer, it's worth your time to learn more about PAM. In part I, "Fundamentals," you learned how to perform typical operations using the PAM window. Chapter 9, "More About PAM," explains additional capabilities provided by PAM.

File System

As explained in chapter 6, "Organizing Your Information," the file system of your computer is similar to that of an office file cabinet. Each disc is like a drawer in the file cabinet. Each drawer can contain one or more folders, which, in turn, can contain other folders or files.

What Is a File? A file is a collection of information stored on a disc or in the computer's internal memory. Each file has a unique name that consists of 1 to 14 characters. You can use any character of the alphabet or any numeral. You may also use the underbar, the minus, and in certain applications, the period. You should avoid using other characters because PAM attaches a special meaning to certain characters (as discussed in "Using the Command Line" in chapter 9).

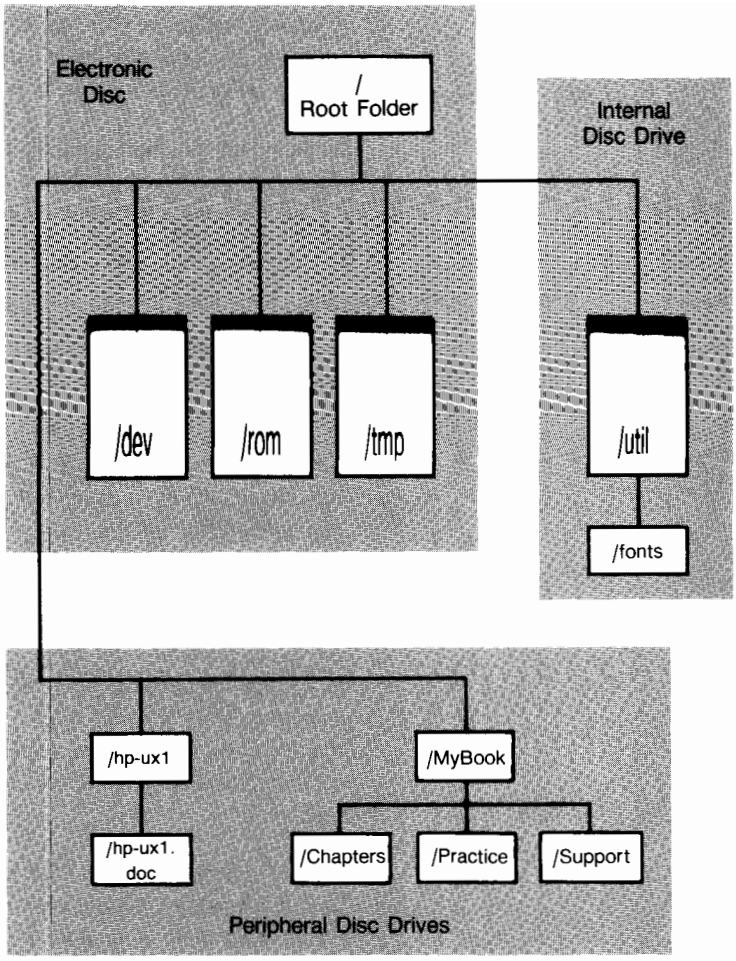
Typical file names include `Junelist`, `Memo-12`, and `Sep.Account`. PAM distinguishes between uppercase and lowercase letters. This means that `test`, `Test`, and `TEST` are three separate file names. If a file name has a period (.) as the first character, the file is a *hidden file*. Hidden files are not listed in the folder area of the PAM display. However, you can still access a hidden file by typing in the complete file name (including the period). Also, you can list the names of hidden files using the `ls` command (refer to `ls` in chapter 11).

There are four types of files in the file system. *Program files* perform specific tasks. Typical types of program files include word processors, disc formatters, and spreadsheet programs. Data created by these files is stored in *data files*. Data files include spreadsheet and document files created by an application program. A third type of file is the *folder*, which organizes files. The fourth type of file is the *device file*. Device files provide access to internal and external physical devices connected to the computer.

What Is a Folder? A folder is a collection of files. You can use folders to help organize your files by grouping them together. For example, if you're creating spreadsheets for three projects, you can make three folders to separate the projects.

The outermost folder on each disc holds all of the other files (and folders) on the disc. When you insert a disc into the built-in disc drive, the outermost folder is included in the file system. (Refer to chapter 10 for information about using external disc drives—they're treated differently from the built-in drive.)

The file system is arranged as a hierarchy of folders. The top level of the file system is called the *root folder*. A typical file system might look like this (the outermost folder names are in color):



File Structure

Accessing Files. The name you use to access a file is called the *path name*. The *full path name* gives the complete name of a file, starting at the root and tracing the hierarchy of the file. For example, to access the file `OpenMe`, you would refer to `/MyBook/Practice/OpenMe`. This tells the computer to look in the folder `MyBook`, then the folder `Practice` to find the file `OpenMe`. This defines the path name for the file `OpenMe`. The first slash character (/) names the root folder; the remaining slashes serve to separate the folder names in the path name.

You can also access a file by using a *relative path name*. A relative path name starts from the current level in the hierarchy and indicates the "downward" path to the file. For example, if `MyBook` is the open folder, the relative path name for accessing `OpenMe` is `Practice/OpenMe`.

Displaying Folder Information. The open folder is displayed in the PAM folder area. This display provides certain information about the files contained in the folder. Files are grouped according to type, such as program files and data files. Folders contain additional information about the files that are in them—refer to `ls` in chapter 11 to learn about displaying this information.

File Security. Every file has three "permissions" (or "modes") that determine which operations can be performed with the file: read permission, write permission, and execute permission. The specific operations allowed by each permission depends upon the file type. Although these permissions are generally set to allow complete access, you can change the permissions using the `chmod` program. (Refer to `chmod` in chapter 11.)

Device Files. Physical devices are accessed through device files. Each device file is associated with an internal or external device. Operations access a device by accessing the corresponding device file. All device files are contained in the `dev` folder, which is in the root folder of the file system.

When you run programs, commands, and utilities, the system automatically accesses certain devices and device files. These default device files correspond to built-in (internal) devices. Peripheral devices are accessed with device files that are automatically created at power on.

Using RAM and the Electronic Disc

The computer dynamically manages the use of its RAM (random-access memory, part of computer memory). It allocates RAM to programs or files only when it is required. When a program has completed execution, the memory it used is returned to the system. This memory then is available for running other programs and for holding files stored in the "electronic disc."

The *electronic disc* is the portion of the file structure that's held in RAM. Essentially, it's the part that isn't on discs and isn't in the `/rom` folder. The root folder is stored on the electronic disc. Therefore, to store files and folders in the electronic disc, you can store them in the root folder. Since RAM is volatile, files stored in the electronic disc—including the root folder—are lost when the computer is switched off. By using the electronic disc (such as the `/` root folder or the `/tmp` folder) to store files or folders in RAM, you make the following operations possible:

- High-speed program loading and file access from an application.
- Copying files between discs using only one disc drive. You can use the root folder for temporary storage while discs are exchanged.
- Accessing disc-based utilities and commands without tying up your disc drives. You can copy the files to the electronic disc so you can use your disc drives for accessing files stored on other discs. (Refer to the next topic.)

You can use the `copy` and `move` to place existing files in the root folder. You can also use the `makefolder` command to create a folder in the root folder. Refer to chapter 11.

Because you can run several programs simultaneously and store files in the electronic disc, it's possible for you to use up all available RAM. In this condition, you wouldn't have enough memory available for running additional programs or performing other needed operations.

If there isn't enough memory to run a program or store a file in the electronic disc, the system displays an error message. You can use any of the following options to increase the amount of RAM available for these tasks. The options are listed in order based on their effect on other programs and files—the first options have the least effect.

- Ending one or more of the application programs that are running.
- Waiting for a program to finish executing before starting another program. When the first program finishes executing, the memory it used becomes available to run other programs.
- Removing a window you don't need. This makes the memory used by the window and its program available for other uses. Each alpha window normally uses about 10K bytes (2K bytes minimum), and graphics windows use about 16K bytes.
- Deleting files you've stored in the root folder (or elsewhere in the electronic disc). Do *not* remove files or folders created by the system.
- Using the `terminate` utility to stop programs.
- Installing one or more memory modules to increase available RAM. (Refer to *Connecting Peripherals to Your Integral PC.*)

Using Disc-Based Commands With One Disc Drive

Several disc-based commands and utilities perform operations on files located on other discs. In order to perform such an operation, the computer must have access to both the disc containing the command or utility (the "program" disc) and the disc that's to be operated upon (the "object" disc). If you have only the built-in disc drive, the computer can't access both discs simultaneously. The way you'll use these commands and utilities depends upon their design:

- Some commands and utilities can start *without* having the "object" disc inserted (mounted). You insert the "program" disc and run the program. When the program shows that it's ready to continue, you can remove the "program" disc and insert the "object" disc. Then you can continue the operation on the "object" disc. The `copy_disc` utility is an example of this type of program.
- Some commands and utilities must have immediate access to the "object" disc, such as one that requires a file name as a parameter when it's executed. You could first copy the command or utility file to the root folder, then insert the "object" disc and run the program from the root folder. (Alternatively, you could copy the command or utility file to the "object" disc, insert that disc, and run the program from that disc. Or you could copy the "object" files to the "program" disc, insert that disc, and perform the operation—although you might be cluttering your "program" disc.) The `ls` and `chmod` commands are examples of this type of program—you could copy them to the root folder when necessary.

Types of Windows

Each application program operates in one of three types of windows: an *alpha* window, a *graphics* window, or a specialized applications window. Each type of window provides features that a program can use to perform its tasks.

For alpha and graphics windows, you can directly access several of their features from the keyboard. These features are described below. (For the specialized applications window, the application program controls your interaction with the window.)

Alpha Windows

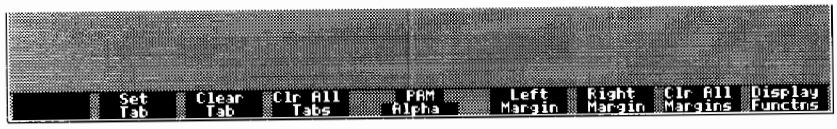
Alpha windows provide features related to processing alphanumeric *characters*. For example, the PAM window uses an alpha window for listing the contents of the open folder and for entering commands.

Alpha Menu. The Alpha menu is activated by pressing (f5) in the System menu while an alpha window is active. The Alpha menu provides functions that help you determine text placement in the window—they're similar to functions used by many computer terminals. The menu is shown and described below.

Caution

These keys provide manual window configuration control that may be in conflict with the controls designed into a particular application. They are rarely needed in normal operation and are provided as an extra convenience for experienced terminal users.

Alpha Menu



- The Set Tab key (f2) sets a tab stop in the current cursor column. (Default tab stops are set every eight columns starting at column 0.)
- The Clear Tab key (f3) clears a tab stop at the current cursor column.
- The Clr All Tabs key (f4) clears all tab stops for the active window.
- The Left Margin key (f5) causes the left margin to be set at the current cursor column. (The default left margin is at column 0.)
- The Right Margin key (f6) sets the right margin at the current cursor column. (The default right margin is at column 79.)
- The Clr All Margins key (f7) sets the left margin at column 0 and the right margin at column 79 (the default margins).
- The Display Functns key (f8) enables or disables display functions mode. In display functions mode, control characters and escape sequences are displayed, rather than executed. An asterisk in the menu label indicates display functions mode is enabled.

Control Sequences. Control characters and escape sequences are sequences of special characters that aren't displayed, but rather perform a function, such as moving the cursor or clearing the display. If you enter control sequences in an alpha window directly from the keyboard, they will have the same effect as if they originated from a program.

Control characters are accessed by simultaneously pressing CTRL and a character key. Escape sequences are entered by pressing ESC followed by several other characters. Normally, control characters and escape sequences are performed and aren't displayed. But if the window is in display functions mode, the control characters and escape sequences are *displayed*, and the functions aren't performed.

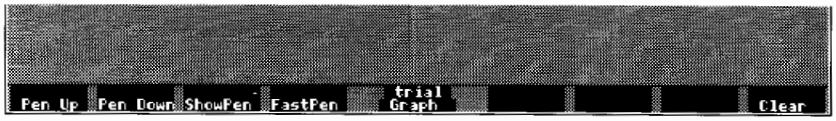
A complete listing of functions performed by control characters and escape sequences is beyond the scope of this manual. If you need descriptions of control characters and escape sequences, contact your local Hewlett-Packard dealer or refer to the *Term0 Reference Manual*, which is part of the HP 82865J Integral PC Programmer's Documentation package.

Graphics Windows

Graphics windows provide features related to processing *graphic elements* by emulating a plotter device. For example, the `plotter_is` utility creates a graphics window.

Graphics Menu. The Graphics menu is activated by pressing (f5) in the System menu while a graphics window is active. The Graphics menu provides functions that control the graphics window and the "pen" in that window. The menu is shown and described below.

Graphics Menu



- The Pen Up key ((f1)) "raises" the pen, allowing you to move the pen without drawing a line. While the pen is up, its image is shown as an outline.
- The Pen Down key ((f2)) "lowers" the pen, allowing you to draw a line by moving the pen. While the pen is down, its image is cross-hatched.
- The Show Pen key ((f3)) turns the pen image on or off. The tip of the pen image corresponds to the plotter pen position. While the pen image is on, an asterisk is displayed in the Show Pen menu label.
- The Fast Pen key ((f4)) changes the relationship between the arrow keys and the plotter pen. Normally, the pen moves one dot each time you press an arrow key. While "fast pen" is active, the pen moves eight dots each time you press an arrow key. While "fast pen" is active, an asterisk is displayed in the Fast Pen menu label.

- The ERROR key (f7) clears error conditions in the graphics window. The menu label is displayed only while an error condition exists.
- The Clear key (f8) clears the graphics window. All data in the graphics window is lost.

Plotting From the Keyboard. The patterns that you can plot in a graphics window directly from the keyboard are limited to lines in the horizontal, vertical, and diagonal directions. You can create a graphics window with the `plotter_is` utility (described in chapter 11). You plot by putting the plotter pen in the down position and then moving it across the window with the arrow keys.

To plot from the keyboard:

1. Select the PAM window and insert the Utilities disc.
2. Type `plotter_is trial` (Return).

   This creates a "hidden" graphics window named `trial`.
3. Select the `trial` window.
4. Activate the System menu, then activate the Graphics menu by pressing Graph (f5).
5. To display the pen, press Show Pen (f3). You can move the pen into view by pressing ➤.

   Before you can draw a line with the plotter pen, you must lower the pen by pressing Pen Down (f2). You can now use the arrow keys to draw lines in the graphics window. The pen is moved like the cursor. Vertical lines are drawn with ▲ and ▼. Horizontal lines are drawn with ◄ and ➤. To draw a diagonal line, press a vertical and a horizontal arrow key simultaneously or alternately. If you hold an arrow key down for several seconds, the repeated key is buffered, and the pen may continue moving for a short distance after you release the arrow key. Pressing Shift with the arrow keys rolls the contents of the graphics window.

Font Options

A *font* is a data file that specifies the visual appearance of all characters. The characters displayed in a window are constructed using the font associated with that window. Each window has a corresponding font that's determined when the window is created. You can change the font that's associated with a window, and you can create and modify fonts according to your needs.

Selecting a Font for a Window

Whenever a window is created, it's assigned a font. In most situations, the window uses the system font, which is stored in the `/rom/6x8` font file.

If you create a window using an operation in the PAM window, that new window takes the same font that the PAM window is using at that time. Again, in most situations, it's the system font.

If you want to change the font for any window, including the PAM window, you can use the `set_font` program (in the `fonts` folder on the Utilities disc). When you run this program, all information in the specified window is redisplayed in the new font.

The Utilities disc provides several fonts in the `fonts` folder that you can use as alternatives to the system font. Some of those fonts are smaller than the system font, and some are larger. In either case, the window size isn't affected by the font—but the amount of information visible at one time *is* affected.

Refer to `set_font` in chapter 11 to find directions for selecting a font at any time. Refer to "Power-On Sequence" later in this chapter for information about setting the font at power-on.

Editing a Font

If you want to edit an existing font file or create a new font file, you should use the `fedit` program (in the `fonts` folder on the Utilities disc). This program enables you to inspect and alter the dot pattern that defines each character in the font. You can start with the system font or any of the fonts in the `fonts` folder on the Utilities disc, or you can start with a blank file and built a font yourself.

Refer to `fedit` in chapter 11 to find directions for modifying or creating a font.

Power-On Sequence

The computer performs a sequence of steps whenever you turn it on. This *power-on sequence* checks the computer's operation, runs an `Autost` program (if present), and creates the PAM window. The following list describes this process.

1. Check RAM and ROM.
2. Create copyright window.
3. Mount all discs present.
4. Is `Autost` file present?
   - (Yes) Change open folder to folder containing `Autost`.
   - (No) Change open folder to first disc (to root if no discs present).
5. Determine language from keyboard type.
6. Get font from open folder (or from `/rom`).
7. Start PAM.
8. Get environment from open folder (or from `/rom`).
9. Process `Autost` file if present in open folder.
10. Display PAM window.

In this sequence, the computer determines the "power-on" folder—the one it opens and searches for special files. The "power-on" folder corresponds to the first disc that contains an `Autost` program (or the first mounted disc if no `Autost` file exists). The computer searches in this order:

1. Built-in disc drive A.
2. Peripheral disc drives in order of HP-IB addresses and unit numbers, if present.

If you want to set up a special condition for your system at power on, you can create the following files in the "power-on" folder:

- A `.environ` file to define the PAM environment.
- A `6x8` file to define the default font.
- A `PAM` file to define an updated PAM program (instead of using `/rom/PAM`) or to define a different power-on program.

Also, you can change the environment and font *after* power on. Refer to "Environment Files" in chapter 9 and "Font Options" earlier in this chapter.

9 More About PAM

Contents

Chapter 9 More About PAM

9 More About PAM

Introduction

PAM—the Personal Applications Manager—is a built-in program that makes it easy for you to interact with the computer. It lets you control various system operations, such as renaming and copying files and starting application programs.

In part I of this manual, you learned about the PAM window and how you can use it to perform some fundamental tasks. This chapter describes more capabilities of PAM. (The main features of PAM are summarized under PAM in chapter 11.)

Using the Command Line

By entering commands into the command line, you can perform operations that aren't provided by the available menus, that involve files that aren't displayed in the open folder, or that require one or more parameters.

To execute commands from the command line, make the PAM window active, enter the command, and press Return. (If the command line is empty, then pressing Return is the same as pressing f1.) If you enter a command containing an error, an error message is displayed. If you want, you can correct errors in the command line—refer to "Editing the Command Line" below.

When using the command line to specify operations, you can use a single period (.) to represent the path name of the open folder. You can use two adjacent periods (..) to specify the folder that contains the open folder.

Command Memory

The last 20 commands entered in the command line are stored in *command memory*. You can display these lines by pressing (Shift)(▲) or (Shift)(▼) to step back or forward through command memory. For example, pressing (Shift)(▲) displays the previous command in command memory. This allows you to execute a previous command again by moving it into the command line, making any necessary modifications, and pressing (Return) again.

Editing the Command Line

You can edit the contents of the command line before executing the command. This may be helpful if you've made a typing mistake or if you want to alter and execute a command from command memory.

You can use the keystrokes in the following list for editing the command line. To execute the revised command, press (Return).

Command Line Editing Keys

Keystroke	Function
(Shift)(◄)	Moves the cursor one character position to the left.
(Shift)(►)	Moves the cursor one character position to the right.
(Shift)(▲)	Displays the previous command in command memory.
(Shift)(▼)	Displays the next command in command memory.
(Back space)	Moves the cursor one space to the left and erases the current cursor character.
(Clear line)	Clears all characters from the current cursor character to the right end of the line.
(Clear display)	Clears the command line and all lines following in command memory.
(Shift)(Clear line)	Clears all characters on the command line.

Command Line Editing Keys (Continued)

Keystroke	Function
(Shift)(Delete line)	Clears the command line.
(Shift)(Delete char)	Deletes the current cursor character.
(Shift)(Insert char)	Toggles insert character mode on and off. While on, the insert indicator appears in the status block of the menu.

Pattern Matching

Most of the operations you specify in the command line involve one or more file names. Many operations require one or more file names as parameters. Frequently, the command name itself is a file name.

When typing a command in the command line, you can use these special characters as *pattern matching* characters: * (asterisk), ? (question mark), and [ ] (brackets). These special characters are used to define a "pattern" of characters that matches one or more file names (or folder names). The following list describes the meaning of each of these characters.

Pattern Matching Characters

Character	Meaning
*	Matches any string of characters (including an empty string).
?	Matches any single character.
[*chars*]	Matches any single character that's listed within the brackets.
[*chars-chars*]	Matches any single character that's listed within the brackets, including the range of characters indicated by the characters before and after the dash. (Characters are listed in order in appendix C.)
[!*chars*]	Matches any single character that's *not* indicated within the brackets.

Each slash (/) in a path name must be explicitly specified as a slash—it's *not* part of the simple file or folder name. The pattern matching characters can match characters only *within* simple file and folder names.

A period (.) at the *beginning* of a file name must be included as a period—it's *not* matched by the * character. A period *within* a file name is treated the same as other characters.

By using the pattern matching characters (sometimes called *metacharacters*), you can avoid typing several related file names or typing a full file name.

Examples. The following command moves all documentation files (files having names with .doc suffixes) from the open folder to the docfolder folder.

```
move *.doc docfolder
```

The next command prints the first, fourth, fifth, and sixth "project" files in /usr/pfolder.

```
print /usr/pfolder/proj[14-6]
```

The next command runs the program whose name starts with HP (such as HPCalc). You can use this form of command if there is only one file name in the open folder that matches the pattern.

```
HP*
```

Quoting

To prevent PAM from attaching special meaning to various special characters, you can enclose those characters in quotation marks. PAM recognizes the following quotation marks: " (double-quote) and ' (single-quote). When one or more characters are enclosed within a set of either single- or double-quote marks, PAM doesn't attempt to interpret them as special characters.

You can also indicate that a single character is *not* a special character—place a \ (backslash) before that character. The character following the backslash is *not* interpreted by PAM as a special character.

Examples. You can use either double-quote marks or single-quote marks to quote a string, but you must use them in matching pairs.

```
echo "some special characters: [ # < > ]"
echo some more special characters: '* & ;'
echo here is a special character \;
```

Quotes can also be quoted, but they must be enclosed within a pair of different quote marks. For example, one or more single quotes can be enclosed within a pair of double quotes.

```
echo 'here are some "double quotes" '
echo "here are some 'single quotes' "
```

Using the Folder Area

The lower part of the PAM window, the folder area, tells you what is in the open folder: the open folder name, folders contained in the open folder, and program and data files in the open folder.

To keep the contents of the folder area current, the system updates the PAM window each time you change your open folder, execute a PAM command that changes the information on the disc containing the open folder, or insert a new disc while the root folder is open. However, it is possible for a program to change the disc without PAM knowing about it. Whenever you wish to update the folder listing, highlight the open folder name and press Reread ((f1)).

If the open folder contains more information than will fit in the folder area, PAM shows as much as fits in the window. To display the next "page" of information, press (Shift)(Next) or select the ‹More Files› field in the window. Similarly, you can use (Shift)(Prev) to see the previous "page" of information, or you can select ‹More Files› to page through the files until you cycle back to the first "page" of files.

The PAM Menu

The PAM User menu enables you to perform frequently-used file operations with a single keystroke. To execute a command, just press the specific function key. The PAM User menu is summarized below.

PAM User Menu Summary

Function Key	Menu Label	Description
f1*	Open	Opens the highlighted folder.
	View	Displays the highlighted data file.
	Start	Executes the highlighted program.
	Reread	Updates the open folder information. (The open folder name must be highlighted.)
f2	Echo	Copies the highlighted file name to the command line.
f3	Send	Sends the highlighted file name to another window.
f4	Move	Used to move the highlighted file.
f5	Copy	Used to copy the highlighted file.
f6	Rename	Used to rename the highlighted file.
f7	Delete	Used to delete the highlighted file.
f8	Close	Closes the open folder.

* The menu label and function of f1 change with the type of file that's highlighted.

Special PAM Keys

Certain keys perform special functions when the PAM window is active. For example, the (Tab) key moves the file indicator across a row of file names. The special PAM keys are summarized in the following list; all other keys retain the functions described in chapter 3.

Special PAM Key Summary

Key	Description
(▲)	Moves the file indicator up a column of file names with "wraparound."
(▼)	Moves the file indicator down a column of file names with "wraparound."
(►)	Moves the file indicator across a row of file names with "wraparound."
(◄)	Moves the file indicator back along a row of file names with "wraparound."
(↖)	Moves the file indicator to the first file in the open folder.
(Shift)(↖)	Moves the file indicator to the last file in the open folder.
(Tab)	Moves the file indicator across a row of file names with "wraparound."
(Shift)(Tab)	Moves the file indicator back along a row of file names with "wraparound."
(Shift)(Next)	Displays the next "page" of file names.
(Shift)(Prev)	Displays the previous "page" of file names.
(Return)	Executes the command in the command line. If the command line is empty, then pressing (Return) is the same as pressing (f1).
(CTRL)(L)	Rereads the disc and updates the open folder information.
(CTRL)(C)	Stops and then restarts PAM. (This is the same as (Shift)(Stop).)

Creating Windows From PAM

Many commands use a separate window for input or output for the operation. For example, the `copy_disc` utility operates in a separate window. Whenever you execute a command that creates a window, PAM follows these conventions:

- PAM first creates a *hidden* window for the command—it appears as a banner at the bottom of the screen. As soon as any information is sent to the window, it appears full size.
- The new window inherits the font, border, and inverted characteristics of the PAM window. (Refer to chapter 8 for information about fonts and window borders.)
- When the command finishes using the window, the window label changes to show the name in parentheses—`( )`. (Usually, PAM removes such a window whenever it needs to create another window. However, you can "save" the window as described in chapter 7.)

Transferring Text

You can transfer file names and character strings from PAM using two commands and two User menu keys. The following list summarizes these capabilities, which are discussed further below.

Command or Key	Function
Echo (f2)	Writes the name of the highlighted file to the command line.
`echo`	Writes a specified character string to another window or file.
Send (f3)	Writes the full path name of the highlighted file to the window you select (followed by carriage return).
`send`	Writes a specified character string to a specified window or to the window you select (followed by carriage return). Can also operate same as Send (f3).

Sending Text

If you want to write the highlighted file's full path name to the current cursor location in another window, you should use the `send` command (with no parameters) or the Send key (f3 in the User menu). If there's more than one other window, you're prompted to select the desired window to receive this information. When the path name is written in the window, it's followed by a carriage return—this is the same as if you had typed the path name in the window and pressed Return.

If you want to write arbitrary text to the current cursor location in another window, you should use the `send` command (with a text parameter). You can either specify the other window in the command or the system will prompt you to select a window. The text is followed by a carriage return—this is the same as if you had typed the text in the window and pressed Return.

After executing `send`, the receiving window becomes the active window. Refer to `send` in chapter 11 for detailed information.

Examples. This command prepares to send the full path name of the highlighted file to another window; the menu labels change so that you can shuffle through the windows to select the desired destination window.

```
send
```

This command sends an "instruction" to the "Chart" program window, where the instruction is executed:

```
send drawit /usr/charts/chart3 #Chart
```

Echoing Text

If you want to write the simple name of the highlighted file to the command line, you should use the Echo key (f2 in the User menu). This is simply a typing aid within the PAM window—no other windows are involved.

If you want to write text to the current cursor location in another window or to a new window, you should use the `echo` command (with or without a text parameter). You can either specify the other window in the command or the system will create an `echo` window. Only the text is written in the window—a carriage return is *not* included.

If you want to write text to a file, you should use the `echo` command (with text and file name parameters). You can create a new file or replace an existing file using `>` with the file name. Or you can append to an existing file using `>>` with the file name. (Refer to "Redirecting Input and Output" below.) If you specify a device file, the text goes to that device (refer to chapter 10, "Installing and Using Peripherals").

After executing `echo`, a receiving window becomes the active window. Refer to `echo` in chapter 11 for detailed information.

Examples. This command writes the string "Testing" to the `check` window:

```
echo Testing #check
```

This command appends the string "End of File" to the end of the file `wicket`:

```
echo End of File >>wicket
```

Redirecting Input and Output

Many programs accept input from the keyboard and send output to the display. You can specify that the input and output of programs be "re-routed" from their normal paths. This *redirection* affects the input and output of only the program it's specified for, and the redirection is cancelled after the program finishes executing.

You specify redirection by including certain symbols in the command that executes the program. (You can use the same redirection symbols with utilities and commands built into PAM when you execute them from the command line.)

Redirecting input and output is a convenient way of selecting the files or devices that a program uses when it's running. You can redirect the displayed output of a program to the active printer by including an output symbol and file name in the command line. This doesn't affect the functioning of the program because the destination of output from the program is changed at the system level.

Input redirection is not used as frequently as output redirection because the keyboard is the default for program input and it is convenient to use. If the keyboard cannot be used for providing program input, then you can specify a file to be used for program input. You designate the file by including an input symbol and file name in the command line when you execute the program.

Every program that is run has three data paths associated with it: standard input, standard output, and error output. Programs use these data paths for communication. Standard output and error output are directed to the window in which the program is running. When this window is the active window, the program can use the keyboard as the standard input path.

The redirection symbols allow you to change one of a program's three data paths while leaving its other data paths unchanged. For example, you can specify a different device for standard output. If you use the device file `/dev/lp` for standard output, the output from the program is printed on the active printer.

Another alternative for redirecting input and output is to designate a window for a program to run in. The program then uses this window for its standard input, standard output, and error output. PAM creates a window if the command line identifies the window name to use for the program.

To redirect the input, output, or error output of a program or to specify a window for it to run in, you enter in the command line the program name (and any required parameters) followed by the redirection information. The information for each redirected path consists of a special symbol and a file name or window name, as required. When you press (Return), the program starts running using the data paths you specified.

The following table lists the redirection symbols, their meanings, and examples.

Redirection Symbols

Syntax	Function	Example
`<`*file*	Read standard input from an existing file.	`program1 <input.data`
`>`*file*	Write standard output to a file.	`program2 >output.data`
`>>`*file*	Append standard output to an existing file.	`sample.prg >>out.data`
`^`*file*	Write error output to a file.	`ABC ^error.listing`
`^^`*file*	Append error output to an existing file.	`pgm.tst ^^error_log`
`#`*window*	Use the specified window for standard input, standard output, and error output.	`newprog #test`

Caution

Information in files or windows is lost when you use certain redirection symbols. If you direct output from a program to an existing file with the > or ^ symbols, the file is emptied and overwritten when the program starts.

If a file you specify with a redirection symbol is not in the open folder, then you must use its full path name to identify it. The system takes the following actions if it can't locate a specified redirection file:

- If a specified input file (< symbol) is not located, the system displays an error message.
- If a specified output file (> or ^ symbol) is not located, the system creates it and uses it for program output.
- If a specified append file (>> or ^^ symbol) is not located, the system creates it and uses it for program output.

Be careful not to specify the same file for standard input and standard output or for standard input and error output. When input and output operations access the same file, the results are unpredictable.

Examples. The following examples show how you can use the redirection symbols to modify the data paths of programs, commands, or utilities. The first example runs the program `prog1`, sends output to the file `data1`, and sends errors to the file `errlist`.

```
prog1 >data1 ^errlist
```

This command runs the program `CHItest` using the file `data1` as input.

```
CHItest <data1
```

The following command adds the current time and date to the end of the file `syslog`.

```
date >>syslog
```

This command displays the file `doc1` in a window named `manual`.

```
view doc1 #manual
```

This last command runs a program named `yodes` and prints only the standard output on the active printer.

```
yodes >/dev/lp
```

Connecting Programs With Pipes

You can connect two or more programs so that the output of one program is used as the input of another program. The data path that joins the programs is called a *pipe*. Pipes allow you to redirect program input and output directly between programs (without redirecting data to and from temporary files).

When programs are connected with pipes, the operating system automatically synchronizes the input and output between the programs. The programs are executed simultaneously. When a pipe is full, the system pauses the program sending data to the pipe. When a pipe is empty, the system pauses the program reading data from the pipe.

Pipes connect the standard output path of one program to the standard input path of another program. The pipes transfer data in only one direction—from program output to program input. The standard input of the first program in the chain comes from its window. The standard output and error output of the last program in the chain go to that program's window.

The vertical bar (`|`) is the "pipe" symbol. To connect two programs with a pipe, simply enter this symbol between the two program names (and any required parameters). For example, the following command specifies the simultaneous execution of three programs joined by pipes.

```
program_A | program_B | program_C
```

Programs connected by pipes operate on data in stages. In the example above, `program_A` might obtain data from a user. This data might be passed by the first pipe to `program_B` where it might be checked for validity and processed. The processed data might then be passed by the second pipe to `program_C` for formatting into a report.

You can use the redirection symbols from the previous section with programs connected by pipes. However, only the data paths that are not joined to the pipes can be changed. If you specify a change to a data path being used with a pipe, then an error occurs. The following redirections are permitted:

- The standard input of the first program can be redirected with the < symbol.
- The standard output of the last program can be redirected with the > symbol or appended to an existing file with the >> symbol.
- The error output for *any* program can be redirected with the ^ symbol or appended to an existing file with the ^^ symbol.

Examples. The following commands show how you can connect programs with pipes and how you can make additional changes to data paths with redirection symbols. The first command connects the programs `test_prog1` and `output_prog` with a pipe.

```
test_prog1 | /usr/output_prog
```

The following command connects the programs `prg.A` and `prg.B` with a pipe and makes additional changes to the standard input and error output paths. For program `prg.A`, the file `data.A` is used as standard input and `err.A` is used for error output. The file `err.B` is used for error output from program `prg.B`.

```
prg.A <data.A ^err.A | prg.B ^err.B
```

This command runs three programs connected with pipes and prints the output of the third program on the active printer.

```
get_it | check_it | process_it >/dev/lp
```

Specifying Sequential Processing

When a program begins, it normally starts executing immediately. You can then run other programs whether or not previous programs have finished executing. However, in some cases it's desirable to prevent additional programs from being run until a specific program has completed. You can accomplish this with *sequential processing*. When you run a program with sequential processing, you can't run additional programs or enter commands from the PAM window until the specific program has finished executing. You can specify sequential processing for any file or command—however, it's normally used only for programs.

If you want to specify sequential processing for an individual command, place a semicolon (`;`) after the last character in the command line. For example, to run the program `backup1` with sequential processing, use the following command:

```
backup1 ;
```

If the program requires parameters in the command line, place the semicolon after the last parameter or redirection symbol, as in the following command:

```
test_program 4 5 >test_output ;
```

While a program is running as a sequential process, PAM places any commands or other keystrokes you enter (while the PAM window is active) in a buffer. There is no response to keyboard activity in the PAM window until after the program has completed. The characters in the buffer for the PAM window are displayed in the command line after the program completes executing.

If other programs are already running when you start a program with sequential processing, all of those programs run simultaneously. While a program is running as a sequential process, you have the options of waiting for the program to finish or using applications already running in other windows.

Examples. The following commands specify sequential processing for programs. The first command runs the program `backup1` as a sequential process in a window called `test_window`.

```
/usr/otherapps/backup1 #test_window ;
```

The next command runs the program `text_formatter` as a sequential process, using file `execfiles/papers/Book1` as a parameter for this program.

```
text_formatter execfiles/papers/Book1 ;
```

This command runs the program `test1_diag`, waits for it to finish, and then runs `test2_diag`.

```
test1_diag ; test2_diag;
```

The last example points out that you can run several programs sequentially using one command. After the first program is run and completed, the next program begins, and so on. All programs run this way are executed in the same window, which has the name of the first program.

Specifying Non–Sequential Processing

You can also run programs *non-sequentially*—each program runs without waiting for previous programs to complete. You specify non-sequential processing by entering an ampersand (&) after the program name.

For example, this command runs the three programs `test1_diag`, `test2_diag`, and `test3_diag` simultaneously.

```
test1_diag & test2_diag & test3_diag
```

To run a single program non-sequentially, the & is optional. (That is, unless specified, programs are processed non-sequentially.) If you specify more than one program in a command line, all of the programs are executed in the same window, which has the name of the first program.

Specifying Operations With Files

You can create three types of files to enhance the operation of the computer: *script* files, *autostart* files, and *environment* files.

- A *script* file is a type of file with execute permission that contains a series of commands like those you enter in the command line. If you execute a script file, you execute all the commands it contains. You can execute a script file at any time.
- An *autostart* file is a file that's used to perform desired operations at the beginning of a session. The file must be named `Autost` and be stored in the outermost folder of a disc for the system to locate it at power on. You can use an autostart file to run an application program without having to enter commands in the PAM window.

- An *environment* file is a data file containing certain variable assignments that determine how the system executes programs and commands. If you want to automatically activate a special environment at power on, the environment file must be named `.environ`. At power on, if the system finds no environment file in the outermost folder of the active disc, it uses the default environment file `/rom/.environ`. Also, you can activate a new environment at any time. The active environment may include a search path for locating files, a home folder that can be easily accessed, and other information about the computer for application programs.

Each of these file types is discussed in greater detail below.

Script Files

A script file contains commands, and you can execute it like a program. By creating a script file, you can build command sequences that meet your particular needs. When you enter the name of the script file into the PAM command line, the commands in the file are executed as if you entered each one individually. Any command that can be entered in the command line can be placed in a script file. You can even execute other script files from a script file—they're used as subprograms or "subscripts."

You can include "comments" in a script file. Each line that starts with ! or # is considered a comment, and it *isn't* executed.

The script file normally runs in a window with the same name as the script file. However, you can redirect the process to a window with another name by using the # redirection symbol.

Output from commands may or may not be displayed when you execute them from a script file, depending upon the source of the output. Output from PAM commands, such as `move` and `copy`, is suppressed. Output from disc-based utilities and other programs is displayed as if you had executed them from the command line. Error messages are displayed for all commands and programs. (Of course, you can redirect the output of individual commands in the script file by using redirection symbols.)

You can create a script file using any text editor—the file is actually created as a data file. If you don't have an text editor application program, you can use either `vi` (on the Commands 2 disc) or `ed` (on the Commands 1 disc). (Refer to appendix F, "Using the `vi` Editor.") However, before you can execute a data file as a script file, you must change its execute permission so it's identified as a program file. You can use the `chmod` command to do this. (Refer to `chmod` in chapter 11.)

Execution. You can execute a script file in two ways:

- Normal execution, in which PAM creates another temporary PAM process to run the commands in the script file. Note that any conditions set up by the script file will apply to only the temporary PAM window—not the main PAM window. For example, if the script file changes the environment or font, that change applies to only the script window and to the commands in the script file—the main PAM window isn't affected. You specify normal execution by entering the script file name as a command.
- "Direct" execution, in which the main PAM program itself executes the commands in the script file. Any conditions set up by the script file *do* apply to the main PAM window, including environment and font changes, for example. You specify direct execution by using the `source` command (`source` *file_name*). Use direct execution when you need to change conditions in the main PAM window.

Script files are normally executed in PAM. Certain users may want the script file to be executed in another "shell" instead. Refer to PAM in chapter 11 for information about how to do this.

Command Processing. Script files normally use sequential processing for all commands. This means that the command on the second line of the script file isn't started until the command on the first line is finished executing. You can run commands simultaneously from a script file by entering an ampersand (&) character after each command that's not to be processed sequentially. Placing an ampersand after a command specifies that the system does not need to wait for it to finish executing before executing the next command in the script file. Using an ampersand after each command in a script file runs all the commands as if they were entered in the command line. Omitting an ampersand after a command in a script file is equivalent to placing a semicolon after the same command when it's entered in the command line.

Script Parameters. If you need to indicate parameters for commands in a script file, you can enter them in the command line when you execute the script file. For example, to execute a script file called `print_it` that requires one file name as a parameter, you can enter the command as

```
print_it lst
```

where `lst` is the file name. Assume for this example that the file `print_it` contains the following commands:

```
copy $1 /
print /$1
delete /$1
```

In the `print_it` script file, `lst` is assigned to the variable `$1`. Then the file is copied to a file with the same name in the root folder. Then this file `/lst` is printed on the active printer. Finally, `lst` is deleted from the root folder. Because sequential processing is the default in script files, the `print` command isn't executed until the file is copied to the root folder, and the file isn't deleted until printing is complete.

The parameters for script files are called "positional" because they're assigned to variables in the script file in the order they're entered in the command line. Parameters must be separated by one or more spaces. The parameter values you enter are passed to the variables `$1`, `$2`, ..., `$9` in the script file. The first parameter is assigned to the variable `$1`, the second parameter is assigned to `$2`, and so on. If you enter fewer than nine parameters, the null string is assigned to each unused variable. The variable `$0` specifies the name of the script file itself.

When you reference all variables at once, or when you need more than nine variables in a script file, you can use the variable `$*`. All the parameters you enter in the command line are passed to the variable `$*` as a long character string. This variable is also useful when you want to give a different interpretation to the parameters, such as character string values.

Autostart Files

The use of an `Autost` file enables you to have a file execute automatically at power on. The file type determines the action that's taken. If the `Autost` file is a data file, it's displayed. If it is a program or a script file, it's executed using "direct execution" (refer to "Script Files" above).

At power on, the system checks each disc drive in order (A, then all external HP-IB drives in increasing address order). The first file named `Autost` that's found in the outermost folder on one of these discs is executed. (If the `Autost` file isn't executable, the action taken depends upon the file type.) The PAM window will appear and accept command input after the `Autost` processing completes. If the system doesn't locate an `Autost` file, the PAM window is displayed.

Any file can become an `Autost` file by simply using the `rename` command to change its name to `Autost`. Names such as `autost` or `AUTOST` aren't recognized because the system differentiates between lowercase and uppercase letters. If the file name isn't `Autost`, the file won't be accessed at power on.

You can create and modify an Autost file using the add_autost utility—you don't have to use a text editor. Refer to add_autost in chapter 11.

Examples. This first Autost file starts the time_of_day program and opens the usr subfolder on the autostart disc. (The time_of_day file must exist on the autostart disc or in another folder in the search path.)

```
time_of_day
./usr
```

This next Autost file sets the built-in printer for compressed printing and displays an alphabetical list of all files and folders in the open folder. (The printer_is and ls files must exist on the autostart disc or in another folder in the search path.)

```
printer_is compressed
ls -a #Files
```

Environment Files

Environment files provide information to the system for defining the *active environment*. The active environment is a collection of settings that the system adds to programs and commands when they're executed. These settings provide information about your system and the files and folders it contains.

The computer system uses an environment file called .environ for defining the active environment at power on. If the system finds an Autost file at that time, the system checks the uppermost folder of the same disc for a file called .environ—if such a file is present, it's used to define the active environment for the current session. If no Autost file is found, the system looks for an .environ file in the outermost folder of the first mounted disc. If an .environ file is not found, the system uses the environment file stored in the computer /rom/.environ.

You can change the active environment during a session by selecting any environment file with the `getenv` command. (Refer to `getenv` in chapter 11.)

The uses of an environment file include the following:

- The system uses the environment file as a source of information when conducting searches. When the name of a program is entered as a command, the `PATH` variable provides a list of folders to search.
- Utilities can determine information about the system environment by reading variables in the environment file. For example, the `time_of_day` utility accesses the variable `TZ` to determine the time zone.
- Commands can use information in the environment file as a supplementary source of information. For example, the `HOME` variable indicates what folder to open when you execute `cd` without a parameter.
- Application programs use the environment file to obtain information about how various features are implemented on the computer. For example, the `LANG` variable indicates the language that the system is localized for.

You normally use a text editor for creating or changing an environment file. If you don't have an text editor application program, you can use either `vi` (on the Commands 2 disc) or `ed` (on the Commands 1 disc). (Refer to appendix F, "Using the `vi` Editor.") You can change the assignment strings of existing variables. You can add different variables to the file as required by your applications. The following form defines each of the variable assignments:

variable=string

In addition,

- Use a colon (:) to separate two parts of a variable assignment. With the PATH variable, colons separate the folder names.
- Use the end-of-line indicator (usually the carriage return character) to end the variable assignment. For example, TERM= sets the terminal type to the null string.
- Don't include spaces within a variable assignment unless they're explicitly desired.

Changing the environment file does not cause a change to be made in the active environment until you activate that environment again. The environment specified by a .environ file is activated when the computer is switched off and on again. You can use the getenv command to change the environment during a session without turning off the computer.

An environment file can contain the following variables.

- The HOME variable sets the home folder. If you execute the cd command without a folder name, this folder is displayed.
- The PATH variable sets the search path for programs and script files. When a program name is entered in the command line, the folders included in the search path are searched in order until the the specified file is found. If this variable isn't specified, the open folder is searched instead.
- The TERM variable provides the terminal type. This can be accessed by application programs to determine whether level 0 or level 1 features are implemented, for example.
- The TERMCAP variable provides information about terminal windows with a terminal capability data base. This data base can be used by applications to identify what capabilities are supported on the computer and how they are accessed.

- The TZ variable provides the time zone information. This can be changed so that the `date` program displays the correct time zone.
- The ACTION variable specifies a program to execute when a data file name is entered as a command. The default action is `view`.
- The SCRSHELL ("script shell") variable specifies the "shell" program that's used to process the script file. If this variable isn't defined (or the shell doesn't exist), then PAM is used. If the variable isn't a full path name, the system searches for the shell using the PATH variable.
- The LANG variable specifies the language of the keyboard. If this variable isn't included in an environment file, the system adds it whenever you activate that environment.

Setting the Home Folder. The home folder is the folder accessed by executing the `cd` command without a parameter. For convenience, you can set the HOME variable to a folder you frequently access. If the folder currently being used as the home folder is not the one you use most often, then the HOME variable in the environment file can be changed to another folder. The folder name assigned to the HOME variable should be an absolute path name.

An example of setting the home folder is changing it from / to a folder called `/usr/execfiles`. The folder you specify must already exist. To change the home folder, access the environment file with a text editor and change the HOME variable to

```
HOME=/usr/execfiles
```

Although the value of the HOME variable has been changed, the home folder used by the system will be / until you set the active environment again at power on or change it with the `getenv` command.

Setting the Search Path. The search path is a list of folders that the system searches in a specific order when it's looking for a program. When you enter a simple program name in the command line, the system checks the folders included in the search path for the specified file. The order of folders in the search path determines the efficiency of the search and, when duplicate file names exist, it determines which file is accessed.

The following conditions affect when the system uses the search path:

- The system does *not* search the open folder unless the search path includes it or unless there is no PATH variable.
- PAM uses the search path only for program (or script) file names entered as commands. Files used as *parameters* are not located with the search path—they must be defined explicitly.
- If you enter a program name as a *path name*, the system doesn't use the search path.

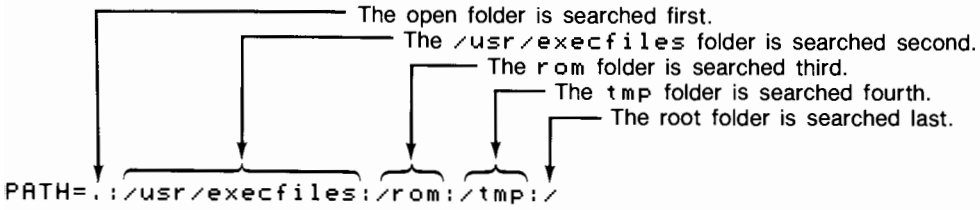
The system searches for a file starting in the first folder in the search path. If the file is not found, the system searches the next folder. When a match is found, the search ceases and the file is accessed. If no match is found, the system displays an error message.

The path names of the folders in the search path may be full path names or relative path names. The path names are separated with colons, as follows:

PATH=*folder_name* : *folder_name* ...

The following drawing shows the order that folders are searched when a command is entered in the command line. If the search is unsuccessful after searching the last folder in the search path, the system displays an error message.

Sample Search Path



If you use the root folder frequently for storing programs and files, you can change the ordering of the folders in the search path to improve search efficiency. The following assignment changes the sample search path so that the root folder, represented with /, is searched first instead of last:

```
PATH=/:.:/usr/execfiles:/rom:/tmp
```

Another change you can make to a search path is adding another folder to it. If you use a folder named /usr/solutions for storing frequently executed programs, then you can include it in the search path. This would allow you to execute programs stored in /usr/solutions from any open folder without specifying a path name. The following variable assignment adds the /usr/solutions folder to the sample search path:

```
PATH=.:/usr/execfiles:/rom:/tmp:/:/usr/
 solutions
```

If a folder is specified for the PATH variable that is not currently included in the file system, then it is ignored. An error message is displayed only when programs entered in the command line cannot be located.

The open folder should normally be included in the search path so that you can easily access its files. The open folder is specified by . (dot) in the PATH variable. You can also indicate a "local" subfolder by specifying a path such as ./usr in the PATH variable.

Setting the Action. When you select a file, the computer takes certain actions according to the file type. The `ACTION` variable in the environment file can change the action taken by the system when data files are selected. If the `ACTION` variable is not present, data files are displayed when they're accessed. When the variable `ACTION` is included in the environment file and is set equal to a program name, then selecting a data file causes that program to begin executing.

Note 

If you change the default action for accessing data files by including the `ACTION` variable in your environment file, you can still use the `view` command to display data files.

Assigning a program to the `ACTION` variable enables application programs to be automatically executed whenever you enter the name of a data file in the command line. The system passes the name of the selected data file to the program as the first positional parameter—entering the name of a data file becomes equivalent to entering the program name followed by the data file name in the command line.

To identify a program as the default action, set the `ACTION` variable equal to a program file. For example, to specify that the program `MemoMaker` be executed as the default action, add the following line to the environment file:

```
ACTION=MemoMaker
```

If you use a simple name to identify the program, then make sure the folder that it's stored in is included in search path.

Selecting Another Environment

You can change the active environment at any time by using the `getenv` command. The `getenv` command replaces the active environment with the environment defined by the specified file.

To select a new environment, enter `getenv` followed by the name of the new environment file. For example, to change the active environment to the settings specified in a file named `sysenv`, enter the command:

```
getenv sysenv
```

The `getenv` command is convenient for updating the environment after editing the environment file without restarting the computer.

Whenever you create a new window from PAM, it inherits the current PAM environment. By changing the PAM environment, it's possible to create several windows that have different environments according to their individual requirements.

If you change the environment by executing `getenv` from a script file using normal execution, the new environment applies to only the commands executed from the script file. After the script file finishes, the environment reverts to its previous definition. If you execute the script file using "direct" execution, the new environment remains in effect. (Refer to "Script Files" in this chapter.)

10 Installing and Using Peripherals

Contents

Chapter 10 Installing and Using Peripherals

10 Installing and Using Peripherals

Introduction

Although you don't need peripherals to operate the computer, they are useful in many situations. For example, you might want to add a disc drive for additional mass storage, use a letter-quality printer to obtain printed copies of documents, or use a plotter to make copies of graphs you created on the computer.

This chapter provides information for connecting peripheral devices to the computer and using those devices. In this chapter you'll learn how to use the built-in HP-IB interface and a serial interface module. (If you have a different type of interface, refer to "Using Special Interfaces" at the end of this chapter.)

Steps to Using Peripherals

Before you can access peripherals, you must first connect them to the computer with an HP-IB cable or with a serial interface module and the proper cable. After a peripheral is properly connected, the computer prepares itself to use the peripheral. Whenever you turn on the computer, it automatically scans its ports to determine the types of peripheral connections.

Note

Connect and turn on all peripherals *before* turning on the computer. This ensures that the computer finds all of the peripheral connections. Otherwise, the computer may not be able to successfully access a peripheral.

The following paragraphs give an overview of how to connect and use peripheral devices. *Connecting Peripherals to Your Integral PC* gives detailed information about connecting specific peripherals into your system—it's your primary source for set-up information.

To use peripherals with the computer, they must be set up according to the following steps:

1. **Connect the peripherals to the computer.** You can connect peripherals to the built-in HP-IB interface, or to another interface that plugs into one of the I/O ports on the back of the computer. Refer to the *Connecting Peripherals to Your Integral PC* for detailed information about connecting peripherals. The following discussion describes how to use peripherals connected to the built-in HP-IB interface and the serial interface card.
2. **Access the peripherals.** Peripherals are typically accessed like files—you read from them and write to them. Or you can assign them certain roles in your system, such as the system printer or plotter.

Example: You have an HP 9133H combination fixed disc/microflexible disc drive that you want to connect to the computer and use. The disc drive contains an internal fixed disc and a slot for a microflexible disc. The disc drive has an HP-IB interface that you can connect to the computer. To connect the disc drive:

1. Check the HP-IB address of the drive. The disc drive has a preset HP-IB address of 0. If no other HP-IB peripherals have this address, you can go on to the next step. Otherwise, you might need to change the address setting of the disc drive. Refer to *Connecting Peripherals to Your Integral PC* for more information.
2. Using a right-angle HP-IB cable, connect the disc drive to the computer. Install the right-angle connector at the computer.

3. Insert a formatted disc into the peripheral disc drive and turn on the drive.
4. Turn on the computer and open the root folder. The outermost folders of the fixed disc and microflexible disc are displayed in the contents. You can now access files on the disc drive.

Example: You have an HP LaserJet Printer that you want to connect to the computer and use. The printer uses a serial interface. You need an HP 82919A Serial Interface and an HP 82947D serial interface cable. To use the LaserJet printer:

1. Connect the LaserJet printer to the computer using the serial cable and the serial interface. (Refer to the user's guide for the interface and *Connecting Peripherals to Your Integral PC* for information about installing and connecting the interface and printer.)
2. Turn on the LaserJet printer. The ready light will blink for about a minute until it is completely warmed up. Then the ready light will be steadily on and the status should be "00", indicating that the printer is ready to use.
3. Turn on the computer. In the command line, type

   ```
   printer_is laserjet cul00
   ```

   and press (Return). (This assumes that the printer is connected to the first serial interface on the computer.)
4. Use the `print` command to send information to the printer.

The dev Folder

The computer maintains several *device files*, each of which is associated with a specific device, indicates the type of device, and uniquely identifies that device. Internal devices, such as the keyboard and the beeper, are accessed with device files. Peripheral devices are also accessed with device files.

The `dev` folder contains all the device files supplied with the computer plus those created at power on because interfaces are installed. The computer uses device files for accessing devices with input and output operations. Reading from a device file enables you to input information from a device. Writing to a device file causes data to be output on a device.

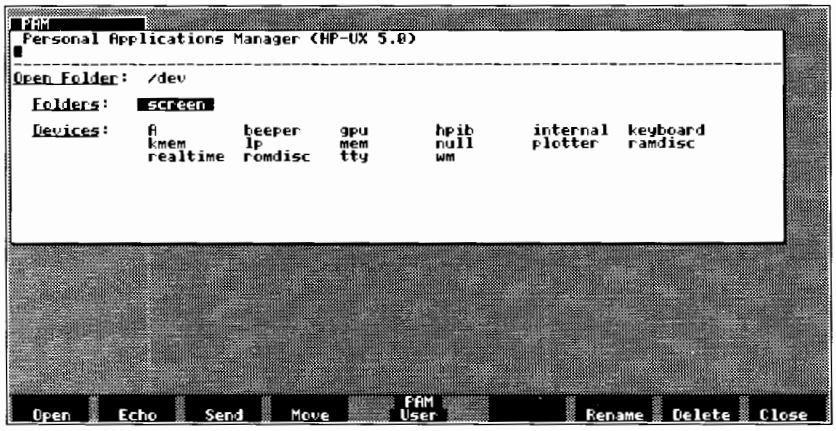
You can use several of the device files in `dev` as parameters for commands and utilities. Device files that you might use as parameters include the following:

- The `A` and `internal` files each identify the built-in disc drive as a parameter with utilities for disc maintenance and other disc operations. The built-in drive is accessed by default when a disc drive is not specified.
- The `lp` file can be used for sending output to the built-in printer or to certain peripheral printers. The printer accessed by this file is called the *active printer*. If there is no active printer, output to this file is lost.
- The `plotter` file is used for accessing plotter windows and compatible peripheral plotters. The plotter accessed by this file is called the *active plotter*.
- The `null` file is a device that accepts and ignores all data sent to it.
- The `D000`, `D001`, ... files, if present, identify individual disc drive units connected to the built-in HP-IB interface.
- The `cul00`, `cul01`, ... files, if present, identify devices connected to each plug-in serial interface. (Note that the third character is the letter "l", *not* the number "1".)

Caution

Don't write directly to a disc drive file such as `A`, `D000`, `D001`, ... If you do, you may corrupt the data on the disc. Use the disc's folder name instead, as described under "Using Peripheral Disc Drives" below.

The following illustration shows devices files included with each computer. (Your computer might show more device files than shown here.)



The built-in HP-IB interface is specially configured for "system" devices, such as HP-IB mass storage devices, printers, and plotters. Although the `hpib` device file is associated with devices connected to the built-in HP-IB interface, these devices are *not* accessed by this device file name. Mass storage devices are accessed using separate device files created for them (refer to "Peripheral Disc Drives" below). Certain utilities identify other devices connected to the built-in HP-IB interface by their HP-IB addresses. For example, a device at address 2 is specified simply as `2`.

A device connected to a serial interface is identified by the device file the system creates for that serial interface. If the device is connected to the first serial interface in the com-

puter, its file name is `cu100`. If it is connected to the second serial interface, the file is `cu101`, and so on up to `cu109`. (Refer to the search order described next.) This is the device name you would use, for example, in a `printer_is` or a `plotter_is` command.

Whenever you turn on the computer, it searches for interfaces in all ports in the order listed below. Note that up to two bus expanders may be connected to the computer.

1. Port A.
2. Port B.
3. Ports A1–A5, if a bus expander is present in port A.
4. Ports B1–B5, if a bus expander is present in port B.

Using Peripheral Disc Drives

You might want to use peripheral HP-IB disc drives to provide additional mass storage or to make it easier to access several discs. When using peripheral disc drives, you must be careful to follow the correct procedures for inserting and removing discs so that the computer always knows which discs are available to it.

In contrast, the system automatically keeps track of discs you install in the built-in disc drive. For this reason, the built-in drive provides a location for you to swap discs most easily. Peripheral disc drives are more suitable for discs that normally remain installed.

The computer can accomodate as many as eight peripheral drives. Also, the system can contain no more than eight file systems at one time (refer to "Accessing Disc Files" below). This means that the number of volumes on one fixed disc may be limited for large systems, for example.

Disc Drive Names

For certain commands and utilities, you must use device names to identify peripheral disc drives. A *drive name* is a device name that identifies an individual drive unit and volume within a disc drive. Drive names are actually the names of device files created by the system in the `dev` folder when the computer is turned on or when you execute `scan_discs`. A drive name has the following form, which uniquely identifies a peripheral drive:

`D` *volume_number HP-IB_address unit_number*

There are actually no spaces between the four characters in the drive name. The three digits following the letter `D` are determined by the system when it creates the device file for an HP-IB disc drive.

- The first digit is the volume number of the disc. This is 0 for most flexible discs and most hard discs. Some fixed discs can be partitioned into several volumes: 0, 1, ...
- The second digit is the HP-IB address of the disc drive. (For this computer, the address must be within the range 0 through 7.) Disc drives have a preset address of 0. Some hard disc drives that include a flexible disc drive use two separate HP-IB addresses, so the system will create two device files for accessing these disc drives.
- The third digit is the unit number. This is printed on the disc drive unit, usually beside the slot where the disc is inserted. (It is usually 0 or 1.) The system creates separate device files for units 0 and 1.

When determining what device file to enter as the device name for a command or utility, you can open the `dev` folder to display the current device files. If a device file for a peripheral drive is not present in the `dev` folder, then it cannot be accessed. If this occurs, make sure the HP-IB address of the peripheral drive is set correctly and that the drive is properly connected to the computer.

Example. The drive name `D000` identifies unit 0 of a peripheral disc drive with an HP-IB address of 0 and a volume number of 0. The drive name for unit 1 in the same disc drive would be `D001`. The drive name `D170` identifies unit 0 of a drive at address 7; the volume number of the drive is 1.

If your system includes a hard disc drive, often one device file is created for accessing it. However, some hard disc drives can have more than one drive name assigned because you can set them to have more than one *volume*. A volume is simply a portion of the hard disc memory that is treated as a separate disc.

Example. Assume that you have an HP 9133H combination fixed disc drive/microflexible disc drive connected to your computer. (No other peripheral drives are connected.) The HP-IB address of the disc drive is 4. The drive has a switch on it that enables you to specify the number of volumes on the fixed disc. (Refer to the owner's manual.) If you select three volumes, then you have the following drive names:

`D041` for the microflexible disc drive (unit 1), volume 0.
`D040` for the fixed disc drive (unit 0), volume 0.
`D140` for the fixed disc drive (unit 0), volume 1.
`D240` for the fixed disc drive (unit 0), volume 2.

Examples: The following commands use device name parameters to specify operations on discs in peripheral disc drives. The first command checks the file on the disc in unit 0 of a peripheral disc drive.

```
verify_disc D000
```

This command changes the disc name of a disc drive.

```
rename_disc zenon D010
```

The next command copies the file system on the disc in the primary drive to the disc in unit 0 of the peripheral disc drive.

```
copy_disc A D000
```

This command copies the file system from the disc in unit 0 of a peripheral disc drive unit to the disc in unit 1 of the same drive.

```
copy_disc D000 D001
```

Formatting Discs

Before you can copy files to either a flexible disc or a hard disc, you must first *initialize* and *format* the disc. During initialization, "track" and "sector" marks are written on the disc, and the disc is tested for bad tracks and sectors. During formatting, a file system is created on the disc. The initialization process may take a long time, especially for a fixed disc (up to a half hour). But a disc has to be initialized only once—thereafter, you need only format the disc if you want to "erase" it. (You must re-initialize as fixed disc if you want to change the number or sizes of its volumes.)

The procedure for initializing and formatting a disc is described in chapter 5, "Protecting Your Information." This procedure uses the `format_disc` utility. While using the procedure, simply specify the appropriate drive name using the conventions described under "Disc Drive Names" above.

If you intend to use a peripheral disc drive to format a disc, you must ensure that the drive is accessible to the computer. Do this in one of these ways:

- If you turn on the disc drive (and allow it to perform its power-on sequence) *before* you turn on the computer, the drive is automatically accessible.
- If you turn on the disc drive *after* you turn on the computer, you must run the `scan_discs` utility to make the drive accessible.

When you format a disc (or a volume on a multi-volume fixed disc), it receives a volume label, which is also the name of the outermost folder of the disc. Whenever this disc is mounted, this folder is included in the root folder, and all of its files become part of the file system.

For a multi-volume fixed disc, you might want to choose the number of volumes and name them according to a common HP-UX hierarchy, such as that listed below. (Refer to the owner's manual for information about how to set the number of volumes on the disc.) This will make your file structure similar to that of other systems, and it will make adding the optional HP-UX C language disc set or the HP-UX Tools disc set easier.

Volume (Folder)	**Use**
`/usr`	General-purpose volume, typically includes subfolders for utility programs, temporary files, manual pages, "include" files, etc.
`/bin`	Utility programs and commands.
`/etc`	System data files and maintenance utility programs.
`/lib`	Object libraries.

Inserting and Removing a Disc

The installation of a disc into a peripheral drive consists of three stages: creating a device file for the disc drive, inserting the disc into the drive, and *mounting* the disc's file structure into the computer's file structure so that you can access the disc files. The action required to do this—and the ease of doing this—depends upon the situation. (Procedures are described later.)

- If you insert a disc in the drive and turn on the disc drive *before* you turn on the computer, the device file is automatically created in the `dev` folder and the disc is automatically mounted. No other action is required. *This is the easiest way to mount a disc.*
- If you turn on the disc drive *after* you turn on the computer, no device file is created. You must run the `scan_discs` utility to do this. If a disc is already inserted in the drive when you run this utility, the disc is automatically mounted at that time.
- If you insert a disc *after* a device file is created for the drive, the disc is *not* mounted. You must run the `mount_disc` utility to mount the disc.
- For a fixed disc drive, if you turn on the disc drive (and allow it to perform its power-up sequence) *before* you turn on the computer, the device files for all volumes are automatically created in the `dev` folder and they're automatically mounted—no other action is required. If you turn on the disc drive *after* you turn on the computer, you must run the `scan_discs` utility.

If you want to remove a disc from a peripheral disc drive, you must first run the `unmount_disc` utility. This prevents loss of data on the disc. You can then mount another disc if you want.

Caution

Before inserting or removing a disc in a peripheral disc drive while the system is turned on, you must inform the computer of your intention. You can do this using the `mount_disc` and `unmount_disc` utilities. If you don't follow the correct procedure for inserting or removing discs from peripheral disc drives, you might lose some or all of the data on a disc.

The device file created for a disc drive is the "drive name" that you must specify for certain operations. The conventions for names are explained under "Disc Drive Names" above. A sample name might be `D021`, representing the drive at HP-IB address 2, unit 1.

To insert a disc before the computer is on:

1. Make sure the computer, the external disc drive, and all other peripherals are turned off.
2. Insert your disc (or discs) in the external drive and switch on the drive.
3. After the disc drives and all peripherals have completed their power-up sequences, switch on the computer and wait for the computer to display folder information. The computer automatically runs the `scan_discs` utility (described in chapter 11), which checks for new disc drives and creates a device file in the `dev` folder. When the computer is turned on, it automatically mounts all discs in all connected, powered-up disc drives.

To install a new drive while the computer is on: Run the `scan_discs` utility to register the new drive and mount the discs in the drive. Execute the command `scan_discs`. (Refer to chapter 11 for details.)

To exchange discs while the computer is on:

1. Before removing a disc from an external drive, run the `unmount_disc` utility to remove the disc file system from the root folder (/). Execute the command `unmount_disc` *drive_name*. (Refer to chapter 11 for details.)
2. Remove the disc from the drive.
3. Insert another disc into the drive.
4. Run the `mount_disc` utility to put the disc file system into the root folder. Execute the command `mount_disc` *drive_name*. (Refer to chapter 11 for details.)

Remember that you use `scan_discs` when you add a disc drive to the system, and you use `mount_disc` and `unmount_disc` for inserting and removing discs from drives that have names in the `dev` folder.

Accessing Disc Files

Disc files are accessed through folders in the system file structure. When a disc is mounted, its uppermost folder is placed in the root folder—the uppermost folder of a disc is the same as the volume label of that disc. When a disc is unmounted, its uppermost folder is removed from the root folder, and the disc becomes inaccessible to the computer.

The system can accomodate up to eight mounted "file systems" simultaneously. Each microflexible disc (including a disc in the built-in drive) contains one file system. Each volume of a fixed disc contains one file system. The computer provides two "built-in" file systems that are automatically included: the

ROM disc (the `/rom` folder) and the electronic disc (the RAM part of the `/` root folder, which contains subfolders that you create in the root folder plus the `/dev` and `/tmp` folders). You can check the root folder at any time to find out how many file systems you have at that time.

If you want to have more than eight file systems available, you can unmount some discs before mounting others. If you have several disc drives, you may be able to do this without physically removing the discs from the drives.

To perform operations that involve files and folders on a disc in a peripheral drive, perform them in the same manner as operations with files on a disc in the built-in drive. For certain operations, such as initializing or formatting a disc with the `format_disc` utility, you must specify the *drive* name, rather than the *folder* name.

Caution

Do not use double-sided discs with a single-sided peripheral disc drive. The additional files and system data stored on the reverse side of a double-sided disc cannot be accessed by a single-sided drive. The absence of certain data about the file system may cause unpredictable results when files are accessed. You can use single-sided discs in either type of drive, but the amount of data that can be stored on them is much less than can be stored on a double-sided disc.

Using Peripheral Printers

You can use peripheral printers if you have special applications that the internal printer isn't suited for. A printer is usually connected to the computer with the built-in HP-IB interface or with an optional serial interface card. You use the `printer_is` utility to change the printer used by the system—the printer designated as the *active printer*. You can run this utility several times during a session to select different peripheral printers (or to select the built-in printer again). You can even select and print with another printer while a previous printer is still printing. The `printer_is` utility is stored on the Utilities disc provided with your computer.

To specify a peripheral printer as the active printer, execute the `printer_is` command with the printer's HP-IB address or its RS-232 device name as a parameter. Other optional parameters for this utility are described under `printer_is` in chapter 11.

You can access the active printer by using the `/dev/lp` device file. It doesn't matter which printer you've designated as the active printer—the `/dev/lp` file always corresponds to the active printer.

You can automatically assign the active printer each time you turn on your computer. Include a `printer_is` command in an autostart file. Refer to "Autostart Files" in chapter 9.

To change and use the active printer:

1. Ensure that the computer and all peripherals are connected and turned on.
2. Run the `printer_is` utility to select the peripheral printer (or built-in printer) as the active printer. Execute a command like one of these:

   `printer_is` *HP-IB_address*
   `printer_is` *RS-232_device_name*
   `printer_is internal` (for built-in printer)

   This enables the `/dev/lp` file to access the printer. (Refer to chapter 11 for details about optional parameters.)
3. Use the `print` command (or redirect output to `/dev/lp`) to print on the active printer.

An HP-IB printer is designated using its HP-IB address, such as `1` or `13`. A serial (RS-232) printer is designated using the device file of its serial interface, such as `cul00` or `cul04`. Each serial interface is assigned a device file according to its location in order in the I/O ports, starting with `cul00`. (Refer to "The `dev` Folder" above.) In addition, the `printer_is` command permits optional parameters that set serial communication for a serial printer. The examples below illustrate how to specify a peripheral printer as the active printer.

To select the built-in printer as the active printer again, execute the command

```
printer_is internal
```

Whenever you execute `print` or `stopprint`, the system uses the active printer. Whenever you print the screen contents using (Shift)(Print) or the `print_screen` utility, the system uses the active printer—although not all printers can process HP-formatted raster graphics data. You may want to make the built-in printer the active printer while printing screen contents.

Note

If you want to use a specific printer with a program, select that printer as the active printer *before* running the program. A program can normally use only the printer that is active at the time the program starts.

Examples. The following commands cause different printers to be selected as the active printer. Files to be printed need to be submitted to a specific printer while it is the active printer.

This command selects an HP-IB printer with an address of 2 as the active printer.

```
printer_is 2
```

If serial interfaces were connected in ports A and B, the next command selects the printer connected to port B as the active printer. (Note that the first interface—in port A—is associated with device file `cu100`.)

```
printer_is cu101
```

This command selects the printer connected to the first serial interface module as the active printer, and selects 1200 baud rate and even parity.

```
printer_is cu100 1200 even
```

The following command selects the built-in printer as the active printer.

```
printer_is internal
```

Using Peripheral Plotters

You access peripheral plotters by using device files, just as you access other peripherals. Before you can use a plotter, you must run the `plotter_is` utility to specify an *active plotter* device. Note that the system does not assign a default active plotter. After you specify an active plotter, you can access it with the device file `plotter`.

You can automatically assign the active plotter each time you turn on your computer. Include a `plotter_is` command in an autostart file. Refer to "Autostart Files" in chapter 9.

To use a peripheral plotter:

1. Ensure that the computer and all peripherals are connected and turned on.
2. Run the `plotter_is` utility to select the peripheral plotter as the active plotter. Execute a command like one of these:

   `plotter_is` *HP-IB_address*
   `plotter_is` *RS-232_device_name*

   This enables the `/dev/plotter` file to access the plotter. (Refer to chapter 11 for details about optional parameters.)
3. If you use a file containing plotter instructions, make sure it's included in the file system. Send output from the file containing plotter instructions to `/dev/plotter`.

After you run the `plotter_is` utility to change the active plotter, you use the device file `plotter` for accessing the active plotter. Refer to chapter 9 for information about redirecting output.

An HP-IB plotter is designated using its HP-IB address, such as `1` or `13`. A serial (RS-232) plotter is designated using the device file of its serial interface, such as `cul00` or `cul04`. Each serial interface is assigned a device file according to its location in order in the I/O ports, starting with `cul00`. (Refer to "The `dev` Folder" above.) In addition, a the `plotter_is` command permits optional parameters that set serial communication for a serial plotter. The examples below illustrate how to specify a peripheral plotter as the active plotter.

If the device name you enter doesn't match a file in the `dev` folder, then the computer creates a graphics window with the name you enter as a parameter. If a plotter window with the specified name already exists, then this window is selected as the active plotter.

Examples. The following commands cause different plotters to be selected as the active plotter. HP-IB plotters have a preset address of 5. The following command selects an HP-IB plotter with an address of 5 as the active plotter.

```
plotter_is 5
```

If serial interfaces were connected in ports A and B, the next command selects the plotter connected to port B as the active plotter. (Note that port A is associated with device file `cul00`.)

```
plotter_is cul01
```

The next command selects a plotter connected to the first serial interface module as the active plotter, and selects 1200 baud rate and even parity.

```
plotter_is cul00 1200 even
```

Using Special Interfaces

You've seen that the system automatically creates device files for devices connected to the built-in HP-IB interface or plug-in serial interfaces. If you want to use any other interface, you'll need to provide a special *driver program* that makes connected devices accessible through the file system. Other interfaces include plug-in HP-IB interfaces and current-loop interfaces, for example.

Note

If you have a *plug-in* HP-IB interface, you should still use the *built-in* HP-IB interface for all mass storage devices. The computer's operating system is designed to perform mass storage operations with only the built-in interface.

The procedure for installing a driver usually involves two files: a utility program that installs the driver, and a data file that contains the code for the driver. The utility creates a device file in the `dev` folder that corresponds to the interface. You can then use that device name in subsequent operations.

1. **Install the interface in the computer.** Refer to your interface owner's manual for details.
2. **Connect and turn on the peripheral device.** Refer to your interface owner's manual or the manual for the peripheral device.
3. **Turn on the computer.**
4. **Install the driver.** Do this by running the installation utility for the interface. The utility is usually provided on a disc.
5. **Access the peripheral.** Use the device file created in the `dev` folder to specify the device.

Each time you turn on the computer, you need to install the drivers for all special interfaces you want to use. You may be able to do this most conveniently using an autostart script file, which runs automatically when you turn on the computer. Refer to "Autostart Files" in chapter 9.

Installation utilities and drivers for certain Hewlett-Packard interfaces are available from Hewlett-Packard.

Example. If you have an HP 82915A 300/1200 BPS Modem, you can use it with your computer by performing this procedure:

1. Install the modem following the instructions in the modem owner's manual.
2. Connect the modem to the phone line following the instructions in the modem owner's manual.
3. Turn on the computer.
4. Install the modem driver by inserting the System disc and running `load_modem`. This creates the modem device file `modem00`.
5. Use the `modem00` device file to specify the modem. For example, you could redirect output to `/dev/modem00`.

If you're using the Datacomm program, you don't have to perform step 4—the Datacomm program automatically installs the modem driver. If you have more than one modem, the `load_modem` program creates device files for all of the modems.

11 Command and Program Dictionary

Contents

Chapter 11

Command and Program Dictionary

11 Command and Program Dictionary

This chapter describes the built-in PAM commands, the friendly utility programs, and several of the application programs. It is arranged in alphabetical order with each program described on its own reference page.

Accessing Commands

Built-In PAM Commands. The built-in PAM commands are available whenever you are in the PAM window.*

Disc-Based Commands. Disc-based commands, utilities, and application programs described in this chapter can be found on the following discs provided with your computer:

- Utilities disc.
- Commands 1 disc.
- Commands 2 disc.
- System disc.
- Diagnostic disc.

To access these commands, you have three choices:

1. Mount the disc containing the command in a disc drive.
2. Copy the command to the root folder.
3. Copy the command to another disc that's mounted.

* The PAM commands are also available from other shells by using the `PAM -c` command to invoke PAM. Refer to the `PAM` entry for more information.

For example, to use the `ls` command to obtain a directory listing of a disc, copy the `ls` command to the root folder, insert your target disc, then start the `ls` command, giving the name of the desired directory.

`/ls` *options directory_name*

Note that some commands such as `ls` or `chmod` need to have all the files specified in the command line present when the command is invoked; other utilities, `format_disc` for example, will prompt you for the appropriate disc when it's needed.

Command Syntax

In general, commands consist of a command name and a series of optional parameters. The command name must be at the beginning of the PAM command line. The optional parameters follow the command name and are separated by spaces. The general syntax is:

`command_name` *optional_parameters ...*

Words shown in `dot matrix` are literals—they are to be entered just as they appear.

Italic words represent optional parameters.

Ellipses (...) are used to show that the previous parameter may be repeated.

The underscore character (_) must be entered when it is part of a command name. When shown as part of the optional parameters it is used to simply join two words together to form a descriptive identifier. The user must substitute an appropriate option for the identifier. For example, the following command name is to be entered exactly as shown:

```
format_disc
```

In the following option, the user must substitute an appropriate file name for the optional parameter:

```
view file_name
```

Case Sensitivity

PAM is sensitive to differences between uppercase and lowercase letters. For example, PAM will assume that a file named "RUNME" is different from a file named "runme." Be sure that you type commands in the PAM command line with the same case and spacing as that shown in the examples.

Parameters

Most commands or utilities have a number of required or optional *parameters*. The form or syntax for each of the *parameters* used by the commands and programs in this chapter is given in the following table.

Command and Program Parameters

Parameter	Description	Examples
drive_name	One of two forms: For peripheral disc drives, use the letter D followed by three digits, indicating *volume_number*, *HP-IB_address*, and *unit_number*. Refer to chapter 10, "Installing and Using Peripherals." For the built-in disc drive, use the letter A (or `internal`).	`unmount_disc` *drive_name* `umount_disc D140`
disc_name	One to six characters, which may include letters, digits, and the following symbols after the initial character: period (.), hyphen (-), and underscore (_).*	`format_disc` *disc_name* `format_disc usr`
file_name†	Complete file names consist of 1 to 14 characters, which may include letters, digits, and the following symbols after the initial character: period (.), hyphen (-), and underscore (_).* For commands that take more than one file name argument, file names may be generated using the *, ? and [...] pattern matching characters. (A *file_name* beginning with a period will not be visible in the PAM window.)	`rename` *file_name* `rename OldFile`

* Some programs allow a wider range of special symbols in names than the period (.), hyphen (-), and underscore (_). However, you should avoid using the following symbols in names because they are reserved for other uses: slash (/), backslash (\), vertical bar (|), greater-than symbol (>), less-than symbol (<), semicolon (;), number sign (#), circumflex (^), ampersand (&), asterisk (*), apostrophe ('), double quotation mark ("), equals sign (=), question mark (?), opening parenthesis ((), closing parenthesis ()), opening bracket ([), and closing bracket (]).

† Wherever a *file_name* appears in this chapter, you may instead use a relative or absolute *path_name*.

Command and Program Parameters (continued)

Parameter	Description	Examples
folder_name	Same syntax as that for a *file_name*. (A *folder_name* beginning with a period will not be visible in the PAM window.)	`makefolder` *folder_name* `makefolder Brand-New`
HP-IB_address	One or two digits, from `0` through `30`, that uniquely identifies a device connected to the internal HP-IB interface.*	`printer_is` *HP-IB_address* `printer_is 2`
program_number	An integer greater than 1 that specifies a program loaded in system memory. (Use either the `ps` or `status` command to obtain a list of program numbers.)	`terminate` *program_number* `terminate 19`
RS-232_device_name	The device name that uniquely identifies a serial interface connected to the computer. Consists of the three letters `cul` and a two digit string from `00` through `09`.	`printer_is` *RS-232_device_name* `printer_is cul00`
window_name	Same syntax as that for a *file_name*.* Precede with a number sign (#) *only* when indicated in the descriptions.	`plotter_is` *window_name* `plotter_is PieChart`

* Cannot be used with plug-in HP-IB interfaces.

Command and Program Parameters (continued)

Parameter	Description	Examples

pattern_matching_character	One of the following pattern matching characters: * (asterisk) matches any string of characters except the period and forward slash; ? question mark matches any single character except the period and forward slash; [...] (a string of characters enclosed in square brackets) matches any single character within the brackets.	`ls` *pattern_matching_character*
		`ls DataFile.???`
		`delete *.bak`

While most commands and utilities require that you specify the parameters when you invoke the program, the following programs from the Utilities disc enable you to make parameter selections interactively from the program window (once the program has started running). Thus typing parameters in the PAM command line is unnecessary.

- `copy_disc`.
- `format_disc`.
- `rename_disc`.
- `verify_disc`.

These programs also have on-line help files. For help with these utilities, enter the command with the `-help` option. For example:

```
copy_disc -help.
```

Overview of Commands

The following tables list the commands and programs provided with your Integral PC. Most users will find the commands and utilities they need described in this chapter. The remainder of this chapter provides information on how to use the built-in PAM commands and the basic utilities that you will use in everyday operation. Information on how to use the HP-UX commands and game programs is provided in `/doc` folders on the discs containing the respective commands.

Built-In PAM Commands*

Command	Description
`cd`	Change directory (change the open folder).
`close`†	Close the open folder.
`copy`†	Copy a file.
`delete`†	Delete a file or folder.
`echo`	Echo text to a window or a file.
`getenv`	Get environment (set the current PAM environment).
`makefolder`	Make a new folder.
`move`†	Move a file or a folder to another folder.
`print`	Print a file.
`rename`†	Rename a file or a folder.
`reread`†	Reread the open folder.
`scan_discs`	Scan for new disc drives.
`send`†	Send a path name or string to a window.
`source`	Run a script file directly.
`stopprint`	Stop printing the current file.
`view`†	View the contents of a file.

* These commands are included in the HP-UX operating system ROM and form part of the Personal Applications Manager.

† These commands can be accessed using the function keys on the PAM User menu or the System menu.

HP-UX Commands (Commands 1 and 2 Discs)

Command	Description
bfs	Big file scanner.
cat	Concatenate, copy, and print files.
chgrp	Change file group ID.
chmod	Change file mode (permissions).
chown	Change file owner ID.
chroot	Change root directory for command.
cmp	Compare two files.
cp, ln, mv	Copy, link or move files.
csh	A shell (command interpreter) with C-like syntax.*
date	Display and set the date.
du	Summarize disc usage.
echo	Echo (print) string.
ed	Line-oriented text editor.
fold	Fold long lines for finite width output device.
grep, egrep, fgrep	Search an ASCII file for a pattern.
hostname	Set or print name of current host system.
hp	Terminal filter.
ls, l, ll, lsf, lsr, lsx	List contents of directories.
mkdir	Make a directory.
mknod	Create special, fifo, files.
more, page	File perusal filter for crt viewing.
mvdir	Move a directory.
nice	Run a command at a low priority.
pr	Printer filter.
ps	Report process status.
pwd	Print name of working directory.
rm, rmdir	Remove files or directories.
sh	Shell program.

HP-UX Commands (continued)

Command	Description
stty	Set the options for a terminal port.
tail	Deliver the last part of a file.
tee	Tee pipe fitting.
touch	Update access/modification/change times of file.
tr	Translate characters.
uname	Print name of current HP-UX version.
vi, view	Visual (screen-oriented) editor.*
wc	Word, line and character count.
whereis	Locate source, binary, and/or manual for program.
which	Locate a program file including aliases and paths.
whoami	Print effective current user id.

* The HP-UX `view` command is a read-only version of `vi`, it is not the same as the PAM `view` command.

Utilities and Games (Utilities Disc)

Utility	Description
add_autost	Create or add to an Autost file.
adventure	Adventure game.
beep_off	Disable the internal beeper.
beep_on	Reenable the internal beeper.
copy_disc	Copy the contents of one disc to another disc.
fedit	Interactive font editor.
format_disc	Format a disc for use.
get_art	Retrieve a picture for GraphicArt program.
GraphicArt	Interactive graphics program.
mille	Mille Bournes game.
mount_disc	Mount a flexible disc file system in a peripheral disc drive.
plotter_is	Set the current plotter device.
print_screen	Print a screen image.
printer_is	Set the current printer device.
rename_disc	Change the name of a disc.
rpn	Reverse Polish Notation calculator.
save_art	Store a picture from GraphicArt program.
set_font	Set the character font for a window.
status	Report status information.
terminate	Terminate the execution of a program.
time_of_day	Display/set time and date.
unmount_disc	Unmount a disc from the file system.
verify_disc	Verify the integrity of a file system.

add_autost

Name

add_autost—create or add to an Autost file

Syntax

add_autost *options ...* "*command*"

Location

Utilities disc

Description

The add_autost utility is used to add a command line to your Autost file. The command line must be enclosed in quotes. If you don't have an Autost file in the current directory, it will create one.

add_autost Options

Syntax	Description
top	Add command line to beginning of Autost file.
bottom	Add command line to end of Autost file.
remove *unique_string*	Remove the first command line that contains the specified string.

When no options are used, the command line will replace a line with the same first word, if present, otherwise it will be added to the end of the file.

After you have created or modified an Autost file in the current directory, use the move or copy commands to create an Autost file on your "boot" disc.

Examples

```
add_autost "printer_is cul00 9600 no-parity xon-xoff"
add_autost top "echo 'Starting Autost Program'"
```

...add_autost

Limitation

The `add_autost` utility will only search for, or use, an `Autost` file if it is in the current directory. If you only have one disc drive and you want to add an `Autost` file to another disc, copy the `add_autost` program to the root directory, insert your disc in the built-in disc drive, then use the full path name to invoke to `add_autost` program.

See Also

Chapter 9

Autost

Name

Autost—Autostart file

Syntax

```
Autost
```

Location

The Autost file must be in the top-most directory on a disc if it is to be recognized by the system. Examples may be found on the Tutor disc.

Description

If Autost is a script file, PAM will automatically execute it when the computer is switched on—the effect is the same as executing source Autost (refer to source in this chapter). If it is not a script file, PAM will take the appropriate action—the action taken is the same as if the user had opened the folder containing the Autost file, highlighted the file, then pressed (Return).

When the power is turned on, PAM will search for a file named Autost in the top-most directories of each disc that is mounted. The search starts with the built-in disc drive, then the peripheral disc drives (in assending order of their HP-IB addresses). It will stop and execute the first—and only the first—Autost file it encounters.

Note that the only difference between an Autost file and any other executable file is simply the name. (The name must be spelled with an initial capital and the remaining letters in lowercase.)

PAM will again search for an Autost file in the following cases:

- If PAM is stopped with the (Shift)(Stop) or (CTRL)(C) keystroke.
- If the computer is reset with the (Shift)(Reset) keystroke.
- If you start another version of PAM. The new version of PAM will search the *current open folder* for the Autost file.

...Autost

Most Autost files are *script files* that cause the execution of other program files. They can be used to automatically start an application program, execute a printer_is command, or start another shell program.

Here are one-line script files:

Sample Autost Script Files

Contents of the File	Description
basic	Causes the file named basic to begin executing in a window named Autost. The PAM window will not appear in the display until basic finishes executing.
basic #BASIC	Causes the file named basic to begin executing in a window named BASIC. The PAM window will not appear in the display until basic finishes executing.
printer_is 1	Assigns the current active printer in PAM to the printer with HP-IB address 1.

Note

An ampersand (&) in an Autost script file enables the PAM window to appear in the display. Without the ampersand, the PAM window will be suppressed until the Autost file runs to completion.

Example

```
Autost
```

Limitation

The Autost file must be in the top-most directory on a disc. Only the first Autost file found will be executed.

See Also

PAM
Chapter 9

beep_off

Name

beep_off—disable the internal beeper

Syntax

```
beep_off
beep_off -help
```

Location

Utilities disc

Description

Beep_off disables the internal beeper so that programs will not be able to generate sounds from the computer.

A beep_off condition remains in effect until:

- You execute a beep_on command.
- You switch the computer off and back on.

Example

```
beep_off
```

See Also

beep_on

beep_on

Name

beep_on—enable the internal beeper

Syntax

```
beep_on
beep_on -help
```

Location

Utilities disc

Description

Beep_on causes the internal beeper to become accessible so that programs can cause the beeper to sound.

You need to use beep_on only after executing the beep_off program.

Example

```
beep_on
```

See Also

beep_off

Name

`cd`—change directory (change the open folder)

Syntax

```
cd
cd ..
cd folder_name
```

Location

Built into PAM

Description

The `cd` command causes a specific folder to be opened and its contents listed in the PAM window. The *folder_name* parameter identifies what folder to open. If the folder is not contained in the current open folder (that is, if you can't see the *folder_name* in the PAM `Folders` area), then use a full path name to identify the folder.

Note

With the exception of the first form of the command above (`cd` by itself), the two letters `cd` need not appear in a change directory command.

Examples. Entering `MyFolder` in the PAM command line is the same as entering `cd MyFolder`. Entering `..` is the same as entering `cd ..` to change directories.

The `cd` command is included in PAM for the convenience of users familiar with the Bourne and csh shells.

Any folder currently included in the file system can be opened with the `cd` command and displayed by PAM unless its *read* or *execute* permission has been removed (refer to `chmod` in this chapter). If the folder specified cannot be located, PAM will display a message and keep the current folder open.

c

Entering `cd ..` (with two periods as the parameter) causes the current open folder to be closed and causes PAM to display the contents of the outer folder. Using `cd ..` is the equivalent of pressing the Close function key (f8).

If `cd` by itself is entered in the PAM command line, then the system attempts to open the *home folder*. The home folder is part of the active environment, which is set at power-on or when the user executes the `getenv` PAM command. The absolute path name of the home folder is assigned to the variable `HOME` in the environment file. If the home folder is not defined or cannot be located, then the root folder of the file system is displayed. (To change the `HOME` variable in your environment file, use one of the editors provided with your system to edit a copy of the default environment file, `/rom/.environ`, save the file, then use the `getenv` command to load your new environment file.)

Examples

```
cd CurrentFiles
cd /usr/CurrentFiles
cd ..
cd ../CurrentFiles
```

Limitation

The `cd` command will not use the `PATH` variable in your environment to search for directories. This means that except for directory names in the current working directory, you will need to specify the absolute or relative path names for the directory.

See Also

`chmod`

`getenv`
Chapter 6

Name

`close`—close the open folder

Syntax

`close`

Location

Built into PAM

Description

Entering the *close* command is the same as pressing Close (f8). The command causes PAM to close the current open folder and to display the files contained in the outer folder.

If you enter a `close` command when the open folder is /, the open folder will not change.

Example

```
close
```

See Also

`cd`
Chapter 6

chmod

C

Name

chmod—change the mode (read/write/execute permissions) of a file or folder

Syntax

chmod *permission_code file_name ...*
chmod *permission_code folder_name ...*

Location

Commands 2 disc

Description

The chmod program changes any of the *permissions* associated with files or folders. These include *read, write,* and *execute* permissions.

You may include any number of *file_names* or *folder_names* in the command line. The same change will be made to each file or folder.

The *permission_code* determines what changes are made to the permissions of the files or folders. You can use chmod to:

- Remove *read* permissions from a file so that the file cannot be edited or viewed.
- Remove *write* permissions from a file to protect the file from accidental deletion or alteration.
- Restore *read* or *write* permissions to a file when you want to modify the file.
- Add *execute* permissions to a file so that the file can be used as a *script file* or executable program.

The *permission_code* consists of two connected symbols and takes the following form:

operator permission_name

The *operator* and *permission_name* symbols that you will use most often are defined in the following tables.

`chmod` Operators

Symbol	Description
+	Add the specified permission to the file.
-	Remove the specified permission from the file.
=	Set the specified permission and remove all other permissions.

`chmod` Permissions

Symbol	Description
r	Read permission. Enable users to view, list, print, and rename files. Enable users to open folders.
w	Write permission. Enable users to modify, copy, move, and delete files. Enable users to change the contents of folders.
x	Execute permission. Enable users to execute a file. Enable users to access the contents of folders.

For example, if you want to change the permission of a file so that it is read-only, use the = operator.

```
chmod =r file_name
```

The above example sets the read permission and removes the other permission for the file. To add execute permission to a file such as a script file, use the + operator and the x permission name:

```
chmod +x file_name
```

This will add execute permission.

The read/write/execute permissions for folders work as follows:

- If you have only one disc drive and you want to perform this operation on another disc, you can copy the command to the root folder. Refer to chapter 8 for more information.
- If you want to allow a user to execute or access a file within a folder but not to *open* the folder itself, then remove the folder *read* permision.
- If you want to secure all the files in a folder from deletion, name changes, or permissison changes, then remove the *write* permission from the folder.
- If you want to deny access to a folder altogether, then remove the *execute* permission from the folder. The execute permission is also called the search permission because it enables users access to the names of the files contained in the directory.
- You can always rename a folder if the folder that contains the desired folder has *write* permission.

Notes

- File permissions are checked with each action performed upon a file (such as renaming a file). Folder permissions are checked with each action performed upon a folder (such as opening a folder or deleting a file from a folder).
- Normally, data files are created with both read and write permissions, but not execute permission. Use `chmod -r` or `chmod -w` when you want to protect a data file from access by others.
- Normally, `chmod` does its work silently. It creates a window only when you've entered an incorrect form of the command.
- Use the `-ld` options with the `ls` program when you want to check the permissions of a folder.

- If you want to transfer files to multi-user HP-UX systems, you may want to set other permission on your files. Refer to the `chmod.doc` file (provided in the `hp-ux2.doc` folder on the Commands 2 disc) for complete information.

Examples

```
chmod +x Your_Script
chmod =r *
chmod +r+w Memo99
chmod -w /usr/DataLog01 /doc/Back_Up_File
```

Limitations

- The `chmod` command doesn't signal PAM when changing a data file to an executable file or *vice versa*. You can use the PAM Reread key (f1) or the CTRL L keystroke to update the PAM window after executing `chmod`.
- Embedded blanks are not allowed in the *permission_code*. For example, `+w` is a valid *permission_code*; `+ w` is not.
- The description of `chmod` on these pages is not complete. Please refer to the documentation listed below for more information.

See Also

`ls`
`/hp-ux2/hp-ux2.doc/chmod.doc` on the Commands 2 disc

copy

Name

`copy`—copy a file

Syntax

`copy` *file_name*
`copy` *source_file_name destination_file_name*
`copy` *source_file_name destination_folder_name*
`copy` *source_file_name ... destination_folder_name*

Location

Built into PAM

Description

The `copy` command enables you to make extra copies of individual files.

If you specify only one parameter after the command name, then the action is the same as highlighting a file name in the PAM window and pressing Copy (f5). PAM will issue a `Please wait...` message while copying the file into the `/tmp` electronic disc folder. Afterwards, PAM will prompt you for the destination name with the message—
`Now complete the copy with 'to file name'.`—and display the word `to` in the command line. After supplying the destination file name or folder name, press Return. PAM will copy the temporary file from `/tmp` to the destination and afterwards delete the temporary file from the electronic disc.

To interrupt the copy operation, erase the word `to` in the command line with the Back space key or Shift Delete line keystroke. The copy operation may be completed by typing the word `to` followed by the name of the destination folder.

When you only have one microflexible disc drive, use the first form of the command to copy a file across discs using the internal disc drive. Insert the second disc in the internal disc drive when prompted to `complete the copy`, and specify the name for the duplicate file after the word `to`. (If you simply press Return, the name of the destination file on the second disc will have the same name as the source file.)

If you specify both source and destination, as in the second form of the command, then PAM will complete the copy operation without further prompting. Unless you specify an absolute or relative path name for the destination file, PAM places the duplicate file in the open folder.

Use the third form of the `copy` command when the destination folder for the file is different from the source folder. A copy of the file by the same name will be made in the destination folder.

The fourth form of the `copy` command provides an easy way to copy a number of files to another folder. List the names of the desired files after the `copy` command; each will be copied to the destination folder. Note that if any one of the copy operations fails, then PAM will display an error message and cancel the rest of the copy operation at that point. Finally, to copy a file from one disc to another when both discs are currently in the system, use path names to identify source and destination discs.

The message line will show `Done with 'copy'` when PAM finishes copying the file or files.

Normally, you shouldn't use the name of an existing file for the *destination_file_name*. If you do specify a file that already exists and if it has write permission (the default), then PAM overwrites the file with information from the source file, causing the loss of information originally in the file.

Note that the `copy_disc` program provides an easy way to copy an entire disc to another disc.

Examples

```
copy MasterFile
copy MasterFile BackupFile
copy Memo99 /etc
copy DataFile01 DataFile02 DataFile03 /etc
```

Limitations

- Entering `copy` by itself causes an error message.
- The `copy` command only allows you to copy individual files; you cannot use the command to copy a disc or a folder.
- If a file or folder permission has been changed with the `chmod` command, then a copy operation may not be successful. The following permissions are necessary for copying files:
  - **a.** A source file must have *read* permission.
  - **b.** If a source file is not in the open folder, then the folder for the file must have both *read* and *execute* permission.
  - **c.** The destination folder must have *write* permission.
  - **d.** If the destination folder is not the open folder, then that folder must have both *write* and *execute* permission.

See Also

`chmod`
`copy_disc`
Chapter 5
Chapter 6 of the Tutor

Name

`copy_disc`—copy the contents of one disc to another disc

Syntax

```
copy_disc
copy_disc drive_name
copy_disc source_drive_name destination_drive_name
copy_disc -help
```

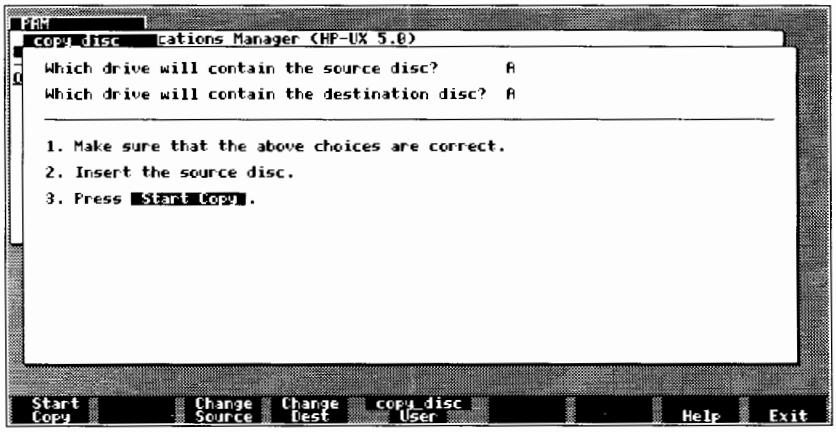
Location

Utilities disc

Description

The `copy_disc` program does a complete byte-by-byte copy of the data from one disc (the *source* disc) to another disc (the *destination* disc). The result will be a complete copy of all the directories, regular files and hidden files on the destination disc. The information about the size of the disc will also be copied so the destination disc will have the same number of bytes available as the source disc.

If you start `copy_disc` without specifying any parameters, you'll create the following program window:



The `copy_disc` window shows that the built-in disc drive will contain both the source and the destination discs.

The following function keys control the `copy_disc` program:

Function Keys for the `copy_disc` Utility

Function key	Description
(f1) Start Copy	Causes the program to start the actual copying from the source disc to the destination disc. The copy operation can be canceled with Exit ((f8)).
Again	Causes the program to return to the initial screen so you can copy another disc.
(f3) Change Source	Enables you to enter a different *drive_name* for the source disc, such as `D000`. Pressing the key moves the cursor to the `source disc` field and clears that field. Type in the new source and press (Return). To restore the field to its previous value, backspace over the field and press (Return) while the field is blank.
(f4) Change Dest	Enables you to enter a different *drive_name* for the destination disc, such as `D001`. Pressing the key moves the cursor to the `destination disc` field and clears that field. Type in the new destination and press (Return). To restore the field to its original value, simply press (Return) while the field is blank.
(f7) Help	Creates a "help" window, separate from the `copy_disc` window, in which you can read usage information about the program.
(f8) Exit	Ends the program, cancels the current copy operation (if in progress), and eliminates the `copy_disc` program window.

If you specify just one *drive_name* in the command line, then that name will appear as both source and destination in the `copy_disc` window.

Notes

- You must format a new disc using the `format_disc` utility before you can copy information to it.
- `copy_disc` will name the destination disc with the same name as the source disc. To rename the new disc, use the `rename_disc` utility.
- `copy_disc` will prompt you to switch discs if the source and destination disc drives are the same unit. (If you have two microflexible disc drives, you can avoid the disc swaps by using both drives.) With only one drive, the program will read as much of the source disc into memory as possible, and then ask you to insert the destination disc to complete the write.

Note

To reduce the number of disc "swaps," eliminate as many programs and windows as possible to free-up memory before running the `copy_disc` program.

- After completing a copy operation, `copy_disc` will display the total number of bytes copied. *The program will continue executing.* To copy another disc, press Again (f1), then follow steps 1 through 3 in the `copy_disc` window. To end the program, press Exit (f8).
- If you press Exit in the middle of a copy operation, the source disc will not be affected in any way, but the destination disc will be left in an undefined state and will probably be unusable.

Disc performance information. There are times when a file-by-file copy to another disc is preferable to a `copy_disc` operation. The reason is that after you have used a disc for a long time and have copied, moved, and deleted a number of files on that disc, the remaining files will be interspersed with "gaps" on the physical medium. The result is slower disc access times and decreased performance. The effect varies, depending on the number of copies, moves, and deletions the disc has undergone.

When you use `copy_disc`, the destination disc inherits the exact file organization of the source disc, including any gaps that exist on the source disc. If instead you use a combination of PAM `makefolder` and `copy` commands (perhaps in a PAM script file) to copy the files from one disc to another disc, the new disc will have its files recorded in an optimum amount of space.

To summarize, `copy_disc` provides an easy way to duplicate discs; a file-by-file disc copy involves more effort, but may result in enhanced performance for the destination disc.

Examples

```
copy_disc
copy_disc D000
copy_disc D000 D001
```

Limitations

- The discs must be the same size and type (for example, both single-sided or both double-sided).
- The Back space key is the only editing key that you can use to edit any of the fields in the `copy_disc` window.

See Also

`format_disc`
Chapter 5
Chapter 5 of the Tutor

delete

Name

`delete`—delete a file or folder

D

Syntax

`delete` *file_name* ...
`delete` *folder_name* ...

Location

Built into PAM

Description

The `delete` command enables you to delete individual files and empty folders.

Caution

Once you delete a file, it is gone forever! There is no way to restore a deleted file. Be certain that you've finished using a file before deleting it.

Include the names of files and empty folders after the `delete` command and then press (Return). The PAM window will be updated to show that a deletion has occurred.

Note that Delete ((f7)) is a typing aid for the `delete` command. Pressing Delete causes the word `delete` to appear in the PAM command line, followed by the name that is highlighted in the PAM window.

To protect a file from accidental deletion, remove its *write* permission using the `chmod` command.

Examples

```
delete OldFile
delete *
delete Memo10 Program22, DataFile44
delete EmptyFolder
```


Limitations

- Entering `delete` by itself causes an error message.
- You must delete all the files in a folder before you can delete the folder itself. Also, the folder must have *read*, *write*, and *execute* permission before it can be deleted.
- A folder must have *write* and *execute* permission before any of the files in it can be deleted.
- You can't delete a file if its *write* permission has been removed.
- The `delete` command cannot be used to delete an entire disc, the open folder, the root folder, or the `/rom` folder.
- If you include multiple file or folder names after a `delete` command and if any one of the deletions fails, then PAM will display an error message and cancel the delete operation at that point.

See Also

`chmod`
Chapter 6
Chapter 6 of the Tutor

diagnostic

Name

`diagnostic`—run the diagnostic test

D

Syntax

```
diagnostic
```

Location

Diagnostic disc

Description

The `diagnostic` program performs a confidence-level check of computer hardware, including:

- Read-only memory (ROM), in which the HP-UX operating system resides.
- Random access memory (RAM), both internal and plug-in, in which your programs run.
- The computer keyboard, including individual keys, connectors, and cable.
- The circuitry leading to the display.
- The internal speaker.
- The built-in printer.
- The real time clock, for providing time and date functions.
- The built-in disc drive.
- The built-in Hewlett-Packard Interface Bus (HP-IB), for controlling peripherals.
- Optional plug-in interfaces.

Caution

The `diagnostic` program clears the contents of computer memory. Do not run the test if you are in the middle of work that you haven't yet saved.

To run the diagnostic program:

1. Insert the Diagnostic disc in the internal disc drive.
2. Highlight and start the `diagnostic` program.
3. Follow the instructions provided by the program.

Example

```
diagnostic
```

Limitation

The `diagnostic` program clears computer memory so that it is not possible to run the program concurrently with other programs.

See Also

Appendix A

echo

Name

echo—echo text to a window or a file

Syntax

```
echo
echo pattern_matching_character
echo text
echo text #window_name
echo text > file_name
echo text >> file_name
```

E
F
G

Location

Built into PAM

Description

The echo command enables you to write a string of characters either to a window or to a file.

For example, entering `echo Done! #PAM` at the end of a PAM shell script file causes PAM to display the message `Done!` in the PAM window.

If the window specified after the # symbol already exists, then PAM writes the *text* beginning at the current cursor location in that window.

The redirection symbol > causes PAM to create the specified file (or to overwrite an existing file) and to enter the *text* beginning at column 1, line 1 of that file. For example, entering:

```
echo Data Error 099 >ErrorLog
```

This example causes PAM to create a file named `ErrorLog` in the open folder and to write the characters `Data Error 099` to that file. (Afterwards, press CTRL L to cause PAM to update the PAM window with the name of the new data file.)

The redirection symbol >> causes PAM to *append* the *text* to a file.

Don't confuse the `echo` command with the Echo key (f2). The command is used for writing strings to windows and to files; the key is used just as a typing aid for the PAM command line.

The main usefulness of `echo` is in PAM script files to display messages as the script files are executing. For example, after you've copied a file using the `copy` command in a script file, you can include a command line, such as `echo Done with copy!` to report on the disc copy completion. You can also use echo with pattern matching characters to print the names of files in the open folder. For example, `echo .*` will create a list of all the hidden files in the open folder.

If you enter an `echo` command without specifying a *window_name* or a *file_name*, then PAM will create an window named `echo`.

Use single quotation marks (' ') or double quotation marks (" ") to enclose any *text* that includes special characters, such as a vertical bar (|). Quoting special characters means that PAM won't use them for anything but textual information. You may nest a pair of single quotation marks within a pair of double quotation marks in the same line, and *vice versa*.

Examples

```
echo "Please wait..." #Backup
echo *
echo "Done with the 'copy' command" >> logfile
```

Limitation

If `echo` directs output to a file, then that file must have *write* permission. If the file is not in the open folder, then the folder itself must have *write* and *execute* permission.

See Also

`chmod`
Chapter 9

fedit

Name

`fedit`—font editor

Syntax

`fedit`
`fedit` *file_name*

Location

E
F
G

Utilities disc in `/util/fonts` folder

Description

The `fedit` program enables you to create and modify *fonts*, or character sets, for the computer. You can open an existing character set (or *font file*) for editing, and then use the cursor keys (or mouse) and the function keys to *change* individual characters and to *add* new characters to the font. Afterwards, you can permanently save the character set in a new font file and use that font file for the character set of any alpha window you choose. In this way you can create your own special symbols, alphabets and graphic elements.

Using the font editor, work on one character at a time by selecting individual dots in the *cell* for that character. You can easily view the results as you make changes.

For example, if you wanted to change the style of the character "A", do the following:

1. Invoke the font editor by first opening the `/font` folder on the Utilities disc, then highlighting and starting `fedit` (this will load the default character set from ROM).
2. Using the cursor keys, move the display pointer to the character `A` in the character set matrix in the `fedit` window. Press (Select) to select the character.
3. Using the display pointer, point at the individual dots in the expanded view of the character `A` and turn them on and off by pressing the select button on the mouse or the (Select) key.
4. Examine the new character `A` by typing (Shift)(A) from the keyboard.
5. When satisfied with your changes, save the changes in a new file with the ? Files and the Store keys.
6. Load the new font into a window with the `set_font` program.

If you enter the `fedit` command by itself, a copy of the system font (or character set) will be loaded for editing. (Note that you can't *change* the system font itself because it is in ROM.) If you enter the `fedit` command followed by a *file_name*, then that font file will be loaded for editing.

For more information on how to use `fedit`, type:

```
fedit -help
```

Example

```
fedit block12x16
```

Limitations

- Using the *fedit* program is easy, but creating a completely new character set can be a lengthy process, because each of the 256 characters must be individually edited.
- The maximum size for a new font is 16 dots wide by 16 dots high. (The size of the display itself is 512 dots wide by 256 dots high.)

See Also

`set_font`
`/util/fonts/fedit.doc` on the Utilities disc

format_disc

Name

`format_disc`—format a disc for use

Syntax

```
format_disc
format_disc disc_name
format_disc disc_name drive_name interleave_factor
format_disc -help
```

E
F
G

Location

Utilities disc

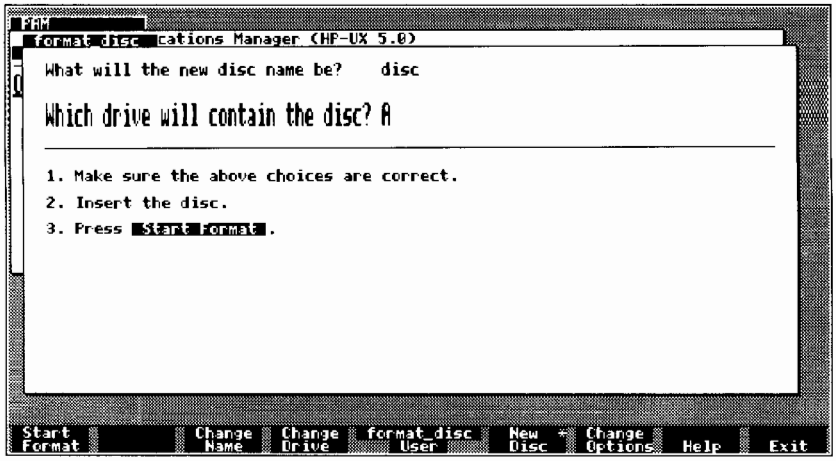
Description

The `format_disc` prepares new discs for storing information.

Caution

The `format_disc` program destroys any information that already exists on a disc. Do not use the `format_disc` utility unless you're certain that nothing of value resides on the disc that you intend to format.

If you start `format_disc` without specifying any parameters, you'll create the following program window:



The `format_disc` window shows that the program will format the disc in the built-in disc drive and will name it `disc`.

The following function keys control the `format_disc` program:

Function Keys for `format_disc`

Function key	Description
(f1) Start Format	Causes the actual formatting to begin. Make sure that the `drive name` field is correct before pressing this key!
Again	Displays the initial screen so you can format another disc.
Continue	Continues the current format operation.
Done	Done with changes to the options. Returns to main screen.

Function Keys for `format_disc`

Function key	Description
(f3) Change Name	Enables you to specify a different *disc_name* for the disc that you will format. (The default name is simply `disc`.) Pressing the key moves the cursor to the `disc name` field and clears that field. Type in the desired name (up to six characters) and press (Return). To restore the field to its previous value, backspace over the field and press (Return) when the field is blank.
Change Interlve	This option allows to specify a disc sector interleave factor for new discs. (This is used to optimize disc performance.) If this factor is set to the default setting of 0, the `format_disc` utility will set the interleave factor based on the type of disc drive. You may reset this number to any positive integer, based on your experience or the manual that comes with your disc drive. The setting of 0 is recommended for most users.
(f4) Change Drive	Enables you to specify a different *drive_name* for the disc drive that will do the formatting. Pressing the key moves the cursor to the `disc drive` field and clears that field. Type in the name of the disc drive, such as `D000`, and press (Return). To restore the field to its previous value, backspace over the field and press (Return) when the field is blank.
(f5) New Disc	Enables you to choose between: ■ Initializing and then formatting a disc (the default), or ■ Only formatting a disc (this saves time if your disc has already been initialized).
(f6) Change Options	This will cause a new screen and menu labels to appear and enables you to change the interleave and swapping space options.
(f7) Help	Creates a "help" window, separate from the `format_disc` window, in which you can read usage information about the program.
(f8) Exit	Ends the program, cancels the current format operation (if in progress), and eliminates the `format_disc` program window.

Caution

Remember to insert the correct disc before starting the formatting operation. Otherwise, you'll format the wrong disc!

Notes

- If you cancel a disc formatting operation, the disc will be left in an undefined state. The disc will need to be reformatted before it can be used again.
- If the new disc option has been selected, and the disc has information on it, the format disc operation will pause to allow you to check the disc. This is to prevent accidental formatting of valuble discs. Press Continue (f1) to format the disc.
- If you specify the *disc_name* or the *drive_name* when entering the command, then the fields in the `format_disc` window will be set to those values. Note that you must include *both* parameters to specify an external disc drive, for example:

```
format_disc bin D020
```

- After completing a format operation, `format_disc` will display the total number of bytes available on the disc medium (not including the swapping area). *The program will continue executing.* To format another disc, follow press Again (f1) then follow steps 1 through 3 in the `format_disc` window. To end the program, press Exit (f8).

`format_disc` operates in two phases:

- The first phase *initializes* the disc medium. For a fixed disc, this phase may take a 2 to 3 minutes/M-byte of disc space. Track and sector marks are written on the disc media. At the same time, the disc is tested for bad sectors and tracks; if necessary, track "sparing" is performed, to prevent these bad areas on the disc from being used.
- The second phase *formats* the disc medium. This phase is much shorter. A *file system* with the specified disc name, interleave factor and swapping area is created on the disc.

You don't always have to initialize a disc medium before formatting it. Once you initialize a new disc, you never need to initialize it again. Use New Disc (f5) to skip the lengthy initializing phase if you simply want to erase the contents of a used disc for a fresh start.

Initially, the New Disc * key label appears with an asterisk (*), indicating that `format_disc` is ready both to initialize and format a disc. Pressed once, the New Disc function key will cause `format_disc` to format the disc *only*—the asterisk (*) in the key label will turn off. Pressed again, the asterisk will disappear (New Disc) and the `format_disc` utility will both initialize and format the disc.

The `format_disc` program calculates the size of a disc medium (measured in 1,024-byte blocks) and the number of files that the disc can hold. The numbers will be reported at the end of the formatting phase.

Examples

```
format_disc
format_disc /etc
format_disc /etc /dev/D030 7 4000
```

Limitations

- The (Back space) key is the only editing key that you can use to edit any of the fields in the `format_disc` window.
- The initializing phase of a `format_disc` operation will slow down system performance. Try not to use the computer for running other programs when initializing discs.

Caution

Do not eject a disc during a disc formatting operation (while the colored busy light is lit). Such action will not only interrupt the operation, but will cause the entire computer to stop functioning until you switch the power off and back on. To cancel a disc formatting operation, press Exit ((f8)).

Name

getenv—get environment (set the current PAM environment)

Syntax

getenv *file_name*

Location

Built into PAM

Description

The getenv command enables you to change the *environment* of PAM and of programs started from the PAM window. The *file_name* parameter identifies an *environment file*—a special text file that consists of environment *variables* and *settings*. The PAM environment is set with the variables in this file. The PATH variable, for example, defines the search path as a list of folders.

PAM Environment Variables

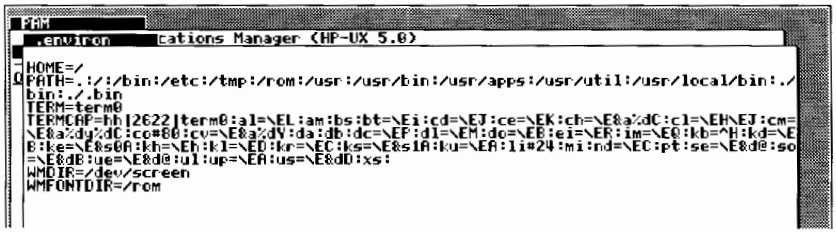
Variable Name	Description
HOME	Sets the folder that will be opened when a cd command is executed without parameters.
PATH	Sets the folders that PAM will search when you enter the simple name of a program in the PAM command line.
TERM	Identifies the terminal type to be used for alpha windows.
SCRSHELL	Identifies the shell to be used by PAM in running shell scripts. The value of the variable may be an absolute, relative or simple path name of a shell. If the value is a simple path name, PAM will search for the shell using the PATH variable. If the shell is not found, the script is processed by PAM.
LANG	Identifies the language being used. PAM will add this variable if it does not already exist when PAM is initialized or the getenv command is executed.
WMDIR	Identifies the directory containing the window manager wm.

PAM Environment Variables (Continued)

Variable Name	Description
TERMCAP	Identifies terminal control sequences that applications can use to control alpha windows.
TZ	Identifies the time zone for the computer. The format is *xxxnzzz*, where *xxx* is a three-character abbreviation for the local time zone, *n* is the difference in hours from GMT, and *zzz* is a three-character abbreviation for the local daylight-savings time zone, if any. *N* may range from 12 through −13 (use positive values for areas *west* of Greenwich and negative values for areas *east* of Greenwich). In Oregon, for example, the complete time zone string reads: TZ=PST8PDT.

Default Enviornment File. When the power is turned on the system will automatically load an environment file. The file will be from one of two sources: /rom/.environ, which is built into ROM; or a .environ file you have created in the "power-on" open folder. The open folder will be the top-most directory on the disc containing your Autost file, or, if you don't have an Autost file, then on the top-most directory on the first disc found in the file system.

The /rom/.environ file consists of the following settings:



Changing the Environment File. The environment file is an ordinary data file. It can be changed with vi, ed, or with most other general purpose text editors. The most common reason for changing the default environment is to reassign the PATH variable—that is, to set the list of folders that PAM will use to find programs as well as the *order* that PAM will search them, and to add a time zone variable.

To create a new environment file, you can copy /rom/.environ to a new disc file (for example, /disc/.environ) and then use a text editor to modify the new environment file.

You may need to use a text editor that allows you to enter lines *longer than 80 characters* since in most cases both the PATH and TERMCAP variables can have more than 80 characters. Both the vi and ed editors will allow lines up to 511 characters.

It shouldn't ever be necessary to change the TERM or the TERMCAP variable. Also, the LANG and WMDIR variables will be needed only by programmers.

Example

```
getenv NewEnviron
```

Limitations

- You cannot set individual environment variables directly from the PAM command line. You must instead use a text editor to create and modify a complete environment file and afterwards set *all* environment variables with the getenv command.
- An environment file must have *read* permission. The folder that contains the environment file must have *execute* permission.

See Also

cd
chmod
Chapter 9

ls

Name

ls—list the contents of folders

Syntax

ls *options file_name ...*
ls *options pattern_matching_character ...*
ls *options folder_name ...*

Location

Commands 1 disc

Description

Use ls to list a directory or more information about a file or a folder than is shown in the PAM window.

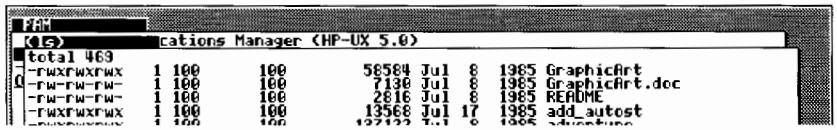
For each *file_name,* ls lists the name and other information specified by the *options.* For each *folder_name,* ls lists the contents of that folder, according to the *options.* By default, ls sorts the output alphabetically. When no names are supplied, ls lists the contents of the open folder.

The more common *options* are:

ls Options

Symbol	Description
-l	List files using a long format, showing the permissions (read, write, and execute), size (in bytes), and time of last modification.
-t	Sort by time of last modification (latest first), instead of by name.
-a	List all entries, including "hidden" files (those whose names begin with a period).
-F	Cause directories to be marked with a trailing slash (/) and executable files to be marked with a trailng asterisk (*).
-R	List the contents of folders contained within other folders (list recursively).
-d	List information (permissions, times, etc.) about folders.

The important information to be gained from running `ls` are the *sizes* of files (in bytes), the modification *times*, and the file *permissions*. File and folder permissions are listed as follows when you specify the `-l` option.



File and Folder Permissions*

Symbol	Description
r	The file is readable.
w	The file is writable.
x	The file is executable. (For a folder, "execute" permission means permission to search the folder for a specified file.)
-	The indicated permission is *not* granted.

* For a complete listing, refer to the on-line manual page for `ls` on the Commands 1 disc.

After you apply `chmod` to a file or a folder, you can observe the effects with the `ls -l` option. The file permissions will be set or cleared according to the options specified in `chmod`.

When you apply the `-l` option by itself to a folder, the `ls` program will report the total size of everything *in* the folder (measured in 1K-byte blocks), before listing the contents of the folder.

Examples

```
ls -al
ls -l File1 Folder1
ls -lR
ls -t | print
ls -F /Disc1
ls -ld /usr /etc
ls -l Accounting.???
```

Limitations

- If you have only one disc drive and you want to perform a directory listing of another disc, you must copy the `ls` command to the root or to the target disc. For more information, refer to chapter 8.
- The output device is assumed to be 80 columns wide.
- The description of `ls` on these pages is not complete. Please refer to the "doc" file listed below for more information.

See Also

`chmod`
`/hp-ux1/hp-ux1.doc/ls.doc` on the Commands 1 disc

Name

`makefolder`—make a new folder

`makefolder` *folder_name ...*

Location

Built into PAM

Description

The `makefolder` command enables you to create new folders for storing files and other folders.

Each *folder_name* you enter as a parameter results in a new folder, created by default in the open folder. To create a folder in a different place, specify the path name for the new folder, such as `/tmp/bin`.

New folders will be created with *read*, *write*, and *execute* permissions. Use the `chmod` program to change folder permissions if you want to secure your information.

PAM will respond, `Cannot create '`*folder_name*`'.`, if you've specified the name of a file or or folder that already exists in the same place. (However, you may have duplicate names in separate folders.)

...makefolder

Examples

```
makefolder /etc
makefolder man1 man2 man3 man4 man5 man6 man7 man8
makefolder /usr/bin ../games lost+found
```

Limitations

- If you include multiple folder names after a `makefolder` command and if any one of the folders can't be created—for one reason or another—then PAM will display an error message and cancel the rest of the operation at that point.
- The folder that is to contain a new folder must have *execute* and *write* permissions.

See Also

`chmod`
Chapter 6

Name

`mount_disc`—mount a file system on flexible disc in an external disc drive

Syntax

`mount_disc` *drive_name*
`mount_disc -help`

Location

Utilities disc

Description

The `mount_disc` program is used to add the files and folders contained on a flexible disc to the root file system so that the computer can access it. The process is called *mounting a disc.*

You do *not* need to use `mount_disc`:

- If the disc is already inserted in the disc drive when the computer is turned on.
- When you change a disc in the *built-in* disc drive.

You *do* need to use `mount_disc` any time that you want to insert a flexible disc in an external disc drive after turning on the computer.

To mount a flexible disc in an external disc drive:

1. Run the `unmount_disc` program for the external disc drive. For example, to unmount the disc in `/dev/D000`, you could insert the Utilities disc in the built-in disc drive and enter `/util/unmount_disc D000` in the PAM command line.

Note

Skip this step and step 2 if you've powered up the computer before inserting a disc in the external disc drive.

2. Remove the flexible disc from the external disc drive.
3. Insert the replacement disc.
4. Execute the `mount_disc` program, specifying the same disc drive.

The disc will be mounted at the root of your file system, so that the disc name will appear in the root (/) folder.

Example

M

```
mount_disc D000
```

Limitation

You can mount only one disc at a time.

See Also

`scan_discs`
`unmount_disc`
Chapter 10

Name

`move`—move a file or a folder to another folder

Syntax

`move` *file_name*

`move` *folder_name*

`move` *source_file_name destination_file_name*

`move` *source_file_name ... destination_folder_name*
`move` *source_folder_name destination_folder_name*

Location

Built into PAM

Description

The `move` command enables you to:

- Move individual files from one folder to another.
- Move an individual folder and all of its contents to another folder on the same disc.

You can think of the `move` command as a combination of two other PAM commands—`copy` and `delete`. That is, a move operation is equivalent to first copying the source file to a destination and then deleting the source file.

If you specify both source and destination names after a `move` command, then PAM will complete the move operation immediately.

If you specify only one parameter after the command, then the action is the same as highlighting a name in the PAM window and pressing Move (f4).

That is, PAM will prompt you for the destination name with the message—`Now complete the move with 'to file name'.`—and display the word `to` in the command line. After supplying the destination, press (Return).

If you specify a *folder_name* as the destination, then the source file will be put in that folder under the same name as the original file. If you specify a *file_name* as the destination, as in the third form of the command, then the source file will overwrite an existing file by that name.

You can include the names of as many files as you'd like after the `move` command. All will be moved to the destination folder (which should appear as the last name in the command line).

The message line will show `Done with 'move'` when PAM finishes moving the files.

To cancel a move operation at the point when PAM prompts you to `complete the move`, erase the word `to` in the command line with the (Back space) or (Shift)(Delete line) keys. (However, see Limitations below.)

Moving a *folder* means that the folder and all the files and folders contained within that folder are moved to the destination folder. The organization of the files and folders *within* the source folder remains intact. You are not allowed to move a folder to another disc, including the electronic disc (`/` and `/tmp`), but you can move a folder to any other folder on the same disc.

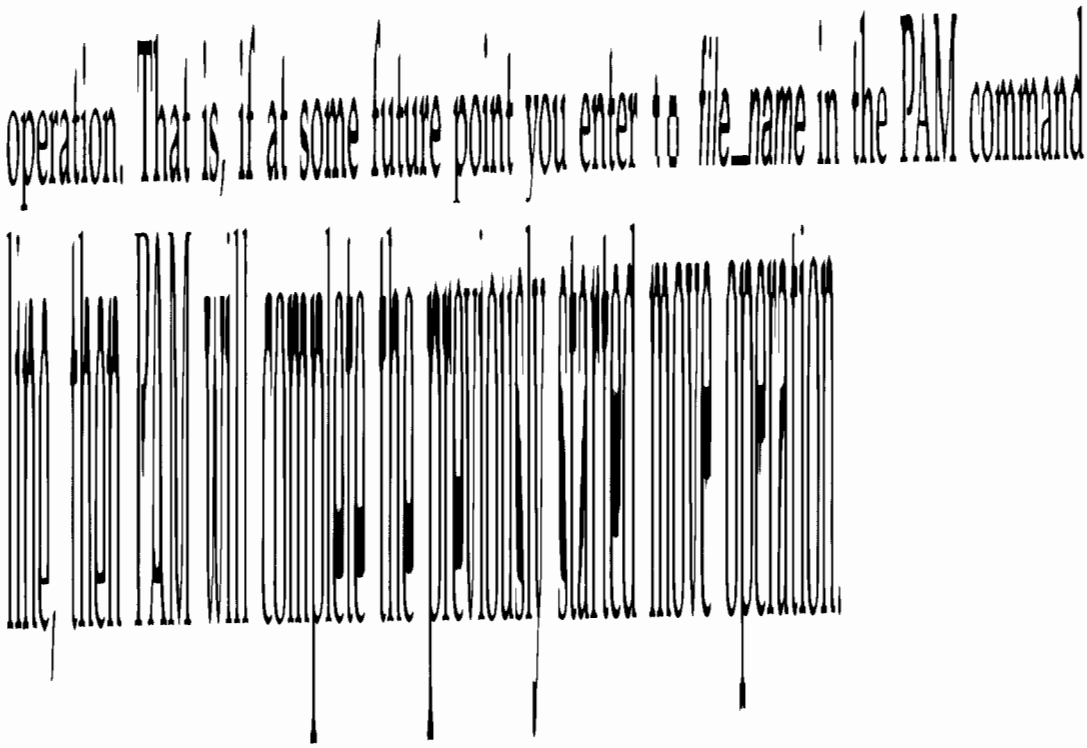
Note that if the destination folder doesn't exist, then the last form of the command is equivalent to renaming the source folder.

Examples

```
move Memo1332 MemoFolder
move Memo44 Memo45
move test.c /usr/src/OldFiles
move File1 File2 File3 NewFolder
move ThisFolder ThatFolder
move ThisFolder ..
```

Limitations

- Entering `move` by itself causes an error message.
- You can cancel a move operation by erasing the word `to` in the PAM command line; however, the move will remain pending until you've started another move, copy, or rename operation. That is, if at some future point you enter `to file_name` in the PAM command line, then PAM will complete the previously started move operation.



- The `move` command cannot be used to move a folder across discs. You can, however, move individual files across discs.
- The `move` command will not allow you to move the open folder.
- If you include multiple file names after a `move` command and if any one of the move operations fails, then PAM will display an error message and cancel the rest of the `move` command at that point.

See Also

`copy`
Chapter 6

Name

PAM—Personal Applications Manager, a visual shell

Syntax

PAM
PAM -c *command*

Location

/rom folder

Description

PAM is a display-oriented command interpreter that provides many traditional HP-UX shell facilities. Unlike traditional shells, the PAM shell maintains a display of the open folder and operates in a window of its own.

Display. There are two parts to the PAM display: the command area and the folder (directory) area. The command (upper) area is the area in which the user inputs commands and receives messages. The command area has two lines. The top line is used to display PAM error and prompting messages. Commands are entered in the lower line. PAM maintains a buffer of 20 command lines, but only one line at a time is visible. The command lines in the buffer can be accessed with the shifted up- and down-arrow keys, then edited using the editing keys.

The folder (lower) area is the area in which PAM displays the files in the folder that is currently open (that is, the current directory). A file name is highlighted in the folder area; this highlighted file name is used as the parameter for PAM commands that are invoked using the PAM menu. The highlighted file name can be changed using the Tab, Shift Tab, and the arrow keys.

Commands. A command is a sequence of words separated by blanks. The first word is the name of the command and the following words are arguments to the command. Two or more commands separated by "|" form a pipeline. The standard output of a command in a pipeline is connected to the standard input of the following command. A command (or pipeline) can be followed by a semicolon ";" to specify that PAM is to wait for the completion of the command before running another commnad. A command (or pipeline) can be followed by a

ampersand "&" to specify that PAM should invoke the command and then, without waiting for the command completion, run other commands. For interactive command input, non-sequential processing is the default; for input of commands from a script, sequential processing is the default.

The action that PAM takes when a file name is entered as a command depends on the type of file.

- Program (executable) files are run as programs or shell scripts.
- Folders (directory files) are opened. This is equivalent to executing a `cd` *folder_name*.
- Data (non-executable) files are displayed using the `view` command, unless there is another action specified by the `ACTION` variable in your environment file.

Input and Ouput. The keyboard and display are the standard input and output. The standard input, output and error output of a command can be redirected using the syntax shown below.

Syntax	Description
< *file_name*	Use the file name as the standard input of the command.
> *file_name*	Use the file name as the standard output of the command.
>> *file_name*	Append the standard output of the command to the end of the named file.
^ *file_name*	Use the file name as the standard error output of the commmand.
^^ *file_name*	Append the standard error output of the command to the end of the named file.
# *window_name*	Use the named window as the standard input, output and error output of the command. If the window doesn't exist, then a window is created. Specific redirection of the input or output will override the redirection done with "#".

Pattern Matching. You can indirectly specify file and folder names using pattern-matching characters: `*`, `?`, `[`, `]`, `-`, and `!`. An `*` matches any string of characters (including no characters). A `?` matches any single character. A pair `[` and `]` encloses a set of characters and matches any single character in the set. For a set within a pair of brackets, a `-` indicates a range of characters, and a leading `!` indicates the set of characters *not* specified within the brackets. Each slash (/) and each leading dot must be matched explicitly. For example, `*` represents all file names, `h*` represents all file names starting with "h", and `?[0-9]` represents all file names of two characters with the last character a numeral.

Quoting. Characters can be quoted on the PAM command line to prevent PAM from processing special characters (<, >, #, |, ;, &). A pair of single or double quotes can be used to quote the string of characters that they enclose. The backslash (\) character quotes only the following character. You cannot quote non-printing control characters in the PAM command line.

PAM Script Procedures A script procedure is an executable file containing command lines and comments. A comment is a line that begins with an exclaimation mark (!); comments are ignored by PAM. (Note, however, that a line beginning with an exclamation mark is not a valid comment line in other shell languages such as the Bourne shell.) The command lines in the script file are executed in sequence (unless non-sequential execution is explicitely specified using the "&" character).

Script parameters that are specified when a script is run can be accessed by script commands using the notation "$1" for the first parameter, "$2" for the second parameter, and so on. All parameters can be accessed at once using "$*". The name of the script can be accessed using "$0".

Script Shell Selection. The SCRSHELL environment variable can be used to identify the shell to be used by PAM in running scripts. If the SCRSHELL variable is defined then the specified shell is used in running all script files started from PAM. If the SCRSHELL variable is not defined then the shell used to run a script can be specified in the first line of the script. To specify a shell script, the first line of a script must use the following form:

#!*shell_name*

PAM will use the PATH variable to find the specified shell if a path name is not provided. If a specified shell is not found, PAM will run the script in a sub-PAM process. (Refer to source in this chapter for information on how to run the script directy.)

P
R

Autostart Files. Autost is the name of a special script or file that is automatically run when PAM is first started. This file must be in the top-most directory on a disc. PAM will search for a file named Autost on the internal disc drive first, then the peripheral disc drives (in order of their HP-IB addresses). PAM will not process any input until the Autost file has completed. If the Autost file is an executable file, PAM will execute it. If it is a directory, PAM will open it. If it is a data file, PAM will take the action specified in the ACTION environment variable, if present, otherwise it will view the file.

Environment. The PAM environment is setup when PAM is run and can be reset at any time using the command `getenv`. The environment variables are read from a file and are not checked or sorted for syntax by PAM. PAM passes the current environment to commands that it starts. PAM itself uses the `ACTION`, `HOME`, `PATH`, and `SCRSHELL` variables. It also will add the `LANG` variable if it does not already exist. (Refer to `getenv` in this chapter for more information.)

PAM Menu. The PAM User menu is available whenever the PAM window is the active window. The command associated with a menu item is run whenever the item is selected by pressing the corresponding function key. The parameter for the command, if it takes one, is the highlighted file name in the folder area of the display.

Built-In Commands. Built-in commands are executed directly by PAM. These commands are summarized at the beginning of this chapter. For more information about a specific command, refer to the corresponding page in this chapter.

Terminating PAM. When PAM receives an interupt signal (CTRL C), it will clean up and terminate. Once it has terminated, the system will automatically start a new PAM program.

Invocation. PAM is normally invoked during the power-on process. PAM can also be invoked from the command line or from a sub-shell using the following form:

```
PAM
```

To execute a single PAM command from a sub-shell (such as the Bourne shell), use the following form:

```
PAM -c command paramemters
```

The specified command (with parameters) will be executed by a sub-PAM process and then sub-PAM will stop. Any commands that affect the environment will only affect the shell that invoked the PAM command.

Example

```
PAM
PAM -c makefolder letters
```

Limitation

The PAM shell does not have a built-in programming language and will not execute shell procedures from other shells that are written in a shell programming language.

Files

```
/rom/PAM
/rom/.environ
```

See Also

Chapter 9

Name

`plotter_is`—set the current plotter device

Syntax

`plotter_is` *window_name*
`plotter_is` *HP-IB_address*
`plotter_is` *RS-232_device_name optional_parameters*
`plotter_is -help`

Location

Utilities disc

Description

The `plotter_is` program defines the standard plotter device (named `/dev/plotter`) that subsequent programs will use to plot graphics data. That is, once you've run `plotter_is` and afterwards started a graphics program, that program will direct any `/dev/plotter` output to the specified window or external plotter.

The plotter definition will remain in effect for the duration of the graphics program. Running `plotter_is` again will affect only programs started afterwards.

The *window_name* parameter specifies a new or existing *graphics* window (as opposed to an alpha window).

Note

Specifying a new *window_name* causes the `plotter_is` program to create a graphics window by that name. To access the graphics menu for that window, select the System menu and press Graph (System menu key (f5)). The window will also show itself when a program first directs its output to the window.

Note that most graphics applications create their own windows so that you don't have to. The `plotter_is` program is most useful to direct program output to an external plotter.

...plotter_is

You may include one or more of the following parameters (in any order) when specifying *optional_parameters* for a serial plotter.

Parameter	Values
baud rate	`110`, `300`, `600`, `1200`, `2400`, `4800`, or `9600`
parity	`odd`, `even`, or `no-parity`
pacing	`xon-xoff`, `hardware-handshake`, or `no-pacing`

If you omit the *optional_parameters*, the plotter will be set to `9600` baud, `no-parity`, and `no-pacing`.

Examples

```
plotter_is SalesFigures
plotter_is 5
plotter_is cul01 1200 even xon-xoff
```

See Also

Chapter 10

Name

`print`—print a file

Syntax

`print` *file_name* ...

Location

Built into PAM

Description

The `print` command causes files to be printed on the current printer device (defined as `/dev/lp`). Enter the names of the files to be printed as parameters.

If the printer is currently busy or multiple files are specified for printing, then the files to be printed are put in a list of waiting files. When the printer finishes printing the current file, it moves to the next top of form and begins printing another file in the list of waiting files. To cancel a print operation that's already started, use the PAM `stopprint` command or select the `print` window and press (Stop).

Printing of the file is canceled if the system cannot access the disc the file is stored on. You can copy files to be printed to the root folder manually before executing the `print` command if all files to be printed cannot be included in the file system at the same time.

You can also "pipe" the output of one program into the `print` command using the vertical bar (|) symbol in the PAM command line. For example, entering:

```
/hp-ux1/ls -al | print
```

This causes the `ls` program on the `/hp-ux1` disc to direct its output to the current printer device via the `print` command.

Examples

```
print Memo99
print /rom/.environ
print File1 File2 File3
ls -al | print
```

Limitations

- Entering `print` by itself causes an error message.
- If the disc containing the file currently being printed is removed from the disc drive, then printing of that file only continues until that portion of the file already in system memory (called the *print buffer*) is completed.
- When an out-of-paper condition occurs during a `print` operation to the built-in printer, the printer will continue to receive data until the 1024-character print buffer of the internal printer is full. To resume printing, load paper into the printer and press the top-of-form button (the button with the grey square). When programs other than PAM experiences an out-of-paper condition while printing to the internal printer, program execution will be suspended when the print buffer fills up. Other program will not be affected unless they also attempt print operations during the out-of-paper condition.
- If a `print` command fails to produces printed output after you've cancelled an earlier print command, check the `/tmp` folder for the files `plock` and `lp0`. The `print` command creates these temporary files to prevent simultaneous print operations. If these files are present, delete them.

See Also

`printer_is`
`stopprint`
Chapter 4

Name

print_screen—print a screen image

Syntax

print_screen

print_screen *options*

print_screen -help

Location

Utilities disc

Description

Use the print_screen utility to send an image of the screen to the current printer or a file. The (Shift)(Print) key combination will produce the same output as the print_screen utility without any options. With options, the print_screen command can print a single window, expand the output, print only a portion of the screen, turn the screen sideways, adjust the dot density, and format the output. The options are summarized in the table below.

print_screen **Options**

Syntax	Description
expanded	Expand the screen image 2× in each direction.
centered	Center screen image vertically on the page.
sideways	Turn printed screen image sideways on the page.
laserjet	Set options for the HP 2686 LaserJet printer.
inverted	Invert the color of the printer output.
form_feed	Send a form feed character after printing the screen.
unframed	Print screen without a border. Without this option a line one dot wide will be added around the screen to form a border.
window *name*	Specifies the name of the window to print. Prints only the window specified.

`print_screen` **Options (Continued)**

Syntax	Description
`density` *number*	Specity density for a LaserJet or similar printer. Acceptable densities are 75, 100, 150, and 300 dots/inch. After printing, the density is reset to 75 dots/inch.
`width` *bytes*	Specify the width of the graphics line on the printer (in bytes). Common widths are: 80 bytes for the ThinkJet printer; 90 bytes for HP 293X series; 75, 100, or 300 bytes for LaserJet printers. The default width is 80 bytes.
`part` *x y w h*	Print only a portion of the screen. This option describes a rectangular portion of the screen. The upper left corner of the rectangle is located at *x,y* (with respect to the upper left corner of the entire screen); the width and height are given by *w* and *h*. The values are in dots or lines.
`narrow`	Make the print narrow on the ThinkJet printer.
`height` *dot_rows*	Indicate the graphic height of the paper, in rows of dots.
`indent` *bytes*	Indent print from left margin. Each byte equals eight dots.
`pad_top` *dot_rows*	Skip *dot_rows* at top of page before printing screen.
`lines_bottom` *number*	Skip *number* of blank lines after printing screen.

Examples

```
print_screen
print_screen expanded sideways
print_screen laserjet window PAM form_feed
print_screen expanded part 0 100 512 155
```

Limitation

If the output is wider than the width of the paper, the right edge of the output will be truncated and the frame on the left side will be eliminated.

See Also

Chapter 4

Name

printer_is—set the current printer device

Syntax

printer_is *product_name RS-232_device_name parameters options*
printer_is *HP-IB_Address options*
printer_is internal *options*

printer_is *options*
printer_is -help
printer_is

Location

Utilities disc

Description

The printer_is program enables you to access external HP-IB and serial printers.

The printer_is program is used to define the standard printer device (/dev/lp) that subsequent programs will use to print text, and to set-up commands to the printer. Once you've run printer_is, subsequent programs will direct any printer output to the specified printer. Specifying internal causes the printer_is program to restore the internal printer as the current printer.

Product Name. Specifying the product name of a Hewlett-Packard printer will set the serial interface to settings shown in the following table.

P
R

Product Name	Serial Interface Setting
laserjet *or* 2686	9600, xon-xoff, no-parity
thinkjet *or* 2225	9600, xon-xoff, no-parity
2932 *or* 2634	2400, xon-xoff, no-parity
2602 *or* 2603	1200, xon-xoff, odd

If a *RS-232_device_name* is specified, one or more of the following parameters can be specified:

Serial Interface Parameters

Parameter	Values
baud rate	110, 300, 600, 1200, 2400, 4800, or 9600
parity	odd, even, or no-parity
pacing	xon-xoff, hardware-handshake, or no-pacing

If you include an *RS-232_device_name*, but omit the *parameters*, the printer will be set to 9600 baud, no-parity, and no-pacing.

The *options* are described in the following table.

`print` **Options**

Option	Sends This String to the Printer
page_length *nn*	\e&l*nn*P
text_length *nn*	\e&l*nn*F
line_spacing *n*	\e&l*n*D
perf_skip	\e&l1L
no_perf_skip	\e&l0L
wrap	\e&l0C
no_wrap	\e&l1C
reset	\eE
normal_pitch	\e&k0S
expanded	\e&k1S
compressed	\e&k2S
expanded-compressed	\e&k3S

`print` **Options (Continued)**

Option	Sends This String to the Printer
`bidirectional`	`\e&k1W`
`no_bidirectional`	`\e&k0W`
`self_test`	`\ez`
`string "`*`character_string`*`"`	*character_string*

Non-printing characters can be sent to the printer by using the `string` option (described above) and one or more of the following two-character strings:

Non-Printing Characters

String	Character
`\\`	Backslash
`\'`	Single Quote
`\t`	Horizontal Tab
`\r`	Carriage Return
`\e`	Escape
`\f`	Form Feed
`\`*nnn*	Sends character code *nnn* (octal)
`\n`	New Line

Examples

```
printer_is
printer_is 0
printer_is page_length 66 string "Page Length now 66 lines\f"
printer_is laserjet cul00 string "\f"
printer_is cul00 1200 even xon-xoff
```

See Also

`print`

rename

Name

`rename`—rename a file or folder

Syntax

`rename` *file_name*
`rename` *folder_name*
`rename` *source_file_name destination_file_name*
`rename` *source_folder_name destination_folder_name*

Location

Built into PAM

Description

The `rename` command enables you to change the name of an individual file or a folder.

If you specify only one parameter after the command, the action is the same as highlighting a name in the PAM window and pressing Rename ((f6)).

That is, PAM will prompt you for the destination name with the message:

```
Now complete the rename with 'to file name'.
```

and then display the word `to` in the command line. After supplying the destination file name or folder name, press (Return).

If you specify both the source and destination names, then PAM will complete the renaming operation without further prompting.

The message line will show `Done with 'rename'` when PAM finishes renaming the file or folder.

Note that PAM will assume that the renaming is to take place within the *open folder*. If you wish to rename a file or folder contained in another folder, then you must specify a path name for *both* source and destination names.

To cancel a rename operation at the point when PAM prompts you to `complete the rename`, erase the word `to` in the command line with the Back space or Shift Delete line keys. (However, see "Limitations" below.)

Note that to rename a disc you need to use the `rename_disc` utility from the Utilities disc.

Examples

```
rename OldFile
rename OldFile NewFile
rename /usr/doc/OldFile ../NewFile
```

Limitations

- Entering `rename` by itself causes an error message.
- You can cancel a rename operation by erasing the word `to` in the PAM command line; however, the renaming will remain pending until you've started another rename, copy, or move operation. That is, if at some future point you enter `to` *file_name* in the PAM command line, then PAM will complete the previously started rename operation.
- The `rename` command cannot be used to rename a disc, the open folder, the root folder, or the `/rom` folder.
- If you include multiple file or folder names after a `rename` command and if any one of the rename operations fails, then PAM will display an error message and cancel the rest of the `rename` command at that point.

See Also

`rename_disc`
Chapter 6

rename_disc

Name

rename_disc—change the name of a disc

Syntax

```
rename_disc
rename_disc disc_name
rename_disc disc_name drive_name
rename_disc -help
```

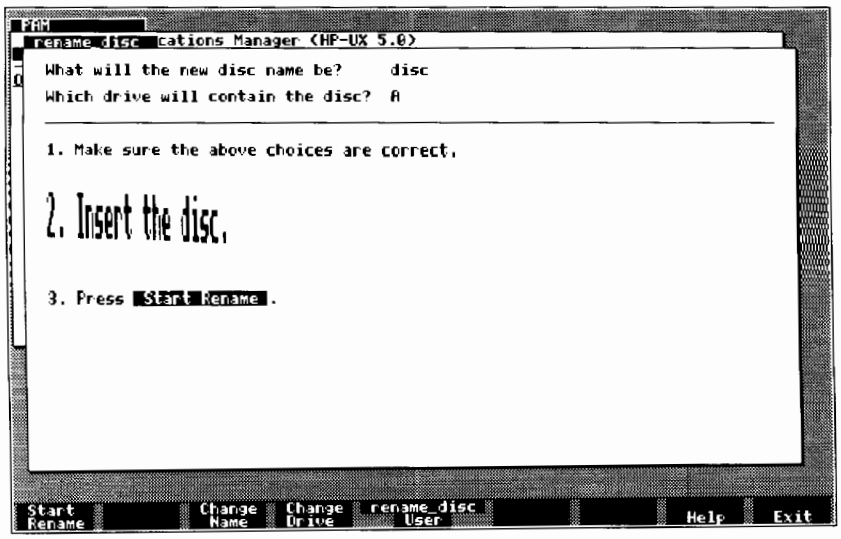
Location

Utilities disc

Description

The rename_disc utility enables you to change the name of a flexible disc or a single volume of a hard disc.

If you start rename_disc without specifying any parameters, you'll create the following program window:



The rename_disc window shows that the program is ready to change the name of the disc in the internal drive to disc.

The following function keys control the `rename_disc` program:

Function Keys for `rename_disc`

Function key	Description
(f1) Start Rename	Causes the actual renaming to occur.
Again	Returns to initial screen and menu.
	Enables you to specify a different *disc_name* for the disc. (The default name is simply `disc`.) Pressing the key moves the cursor to the `new disc name` field and clears that field. Type in the desired name (up to six characters) and press (Return). To restore the field to its previous value, backspace over the field and press (Return) when the field is blank.
(f4) Change Drive	Enables you to specify a different *drive_name* for the disc drive that will do the renaming. Pressing the key moves the cursor to the `disc drive` field and clears that field. Type in the name of the disc drive, such as `D000`, and press (Return). To restore the field to its previous value, backspace over the field and press (Return) when the field is blank.
(f7) Help	Creates a "help" window, separate from the `rename_disc` window, in which you can read usage information about the program.
(f8) Exit	Ends the program, cancels the current rename operation (if in progress), and eliminates the `rename_disc` program window.

...rename_disc

Notes

- If you specify the *disc_name* or the *drive_name* when entering the command, then the fields in the `rename_disc` window will be set to those values. (Note that you must include *both* parameters to specify an external disc drive.)
- After completing the rename operation, you can rename another disc. Press Again (f1), then follow steps 1 through 3 in the `rename_disc` window. To end the program, press Exit (f8).

Examples

```
rename_disc
rename_disc /usr
rename_disc /usr D000
```

Limitation

The Back space key is the only editing key that you can use to edit any of the fields in the `rename_disc` window.

Name

reread—reread the open folder

Syntax

```
reread
```

Location

Built into PAM

Description

The reread command causes PAM to check the contents of the current open folder and to update the PAM window with that information.

Entering reread in the PAM command line is the same as highlighting the name of the open folder and pressing Reread (f1).

Note also that pressing CTRL L is the same as entering the reread command.

Use reread when you've made changes to one or more files in the open folder and you want to ensure that the list of files and folders in the PAM window is properly updated.

For example, if you've changed a text file to a PAM script file using the chmod command, you need to cause PAM to update the window in order to verify that the *data* file has become a *program* file.

Example

```
reread
```

See Also

chmod
Chapter 9

rpn

Name

rpn—A calculator using Reverse Polish Notation

Syntax

```
rpn
```

Location

Utilities disc

Description

The rpn program serves as a personal use calculator. The rpn window is small enough so that you can keep the calculator around in the display while running other applications.

The calculator uses the same form of logic as other Hewlett-Packard calculators—Reverse Polish Notation. To calculate results, you enter first one number, press (Return), enter the next number, and then press the one of the four function keys (such as (−) for subtraction).

Functions available from the keyboard include addition ((+)), subtraction ((−)), multiplication ((*)), division ((/)), and percent ((%)). Use the numeric keypad at the righthand side of the keyboard for easy digit entry.

For example, to calculate $23.8 \times 13.6 \times 10^{-6}$, do the following:

1. Type 23.8.
2. Press (Return) to enter the first operand.
3. Type 13.6.
4. Press (E) for *exponent*.
5. Type 6, then press chs ((f1)).
6. Press (*) to multiply.

The answer will be displayed in the X-register of the calculator. If you're familiar with Hewlett-Packard calculators, you'll recognize the visual display of the X, Y, Z, T, and L (last X) registers in the calculator window.

The function key labels for the calculator indicate other calculator functions, such as chs (f1), to change the sign of the number in the X-register.

To change the number of digits shown, use the f key followed by a number. For example, to display four decimal places, press f 4. (The "f" stands for "fixed point.")

The send key operates similarly to the PAM Send key. Its purpose is to send results from the X-register to other program windows. For example, you can send a calculated result to a spreadsheet window.

Example


```
rpn
```

Limitations

- The calculator provides only one storage register, accessed with function keys f5 and f6.
- The calculator is not programmable.

See Also

send
The owner's manual for any Hewlett-Packard calculator

scan_discs

Name

`scan_discs`—scan for new disc drives

Syntax

```
scan_discs
```

Location

`/rom` folder

Description

The `scan_discs` program enables you to add new disc drives to the computer after you've already powered up the computer.

Normally, *before* switching on the computer, you should switch on all peripheral disc drives (in any order), and allow them to complete their self-tests.

Then when you switch on the computer, the `scan_discs` program is run automatically to find the peripheral disc drives connected by the HP-IB interface to the computer. If you're using one or more peripheral disc drives, you may notice:

- The busy lights of the disc drives will turn on and off, perhaps several times.
- The PAM window will be updated one or more times.

The reason for this activity is that `scan_discs` is locating the drives and creating names for the drives in the `/dev` folder so that you can access the drives later on. Within a few moments, when `scan_disc` finishes executing, PAM *will* be able to find all of your discs.

If you want to add a disc drive to the system after you've turned the computer on, simply enter `scan_discs` in the PAM command line. Although the program does its work silently, you may note its progress by observing the disc drive lights.

Afterwards, you can check the new drive name by examining the `/dev` folder and check the *disc name* of the disc in the drive by examining the root folder.

Example

```
scan_discs
```

Limitations

- If you have a number of external disc drives connected to the computer, then `scan_discs` may take a long time to finish executing.
- `scan_discs` is useful only for adding external disc *drives* (the hardware) to the system. Its purpose is *not* to add individual *discs* to the system (the software). If you want to do the latter after you've already turned the power on, you should use the `mount_disc` program from the Utilities disc.

See Also

`Autost`
`getenv`
`mount_disc`
`unmount_disc`
Chapter 10

send

Name

send—send a path name or string to a window

Syntax

send
send *string_of_characters window_name*

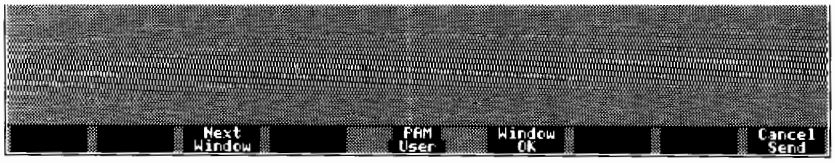
Location

Built into PAM

Description

The send command enables you to send a string of characters or the complete path name of a file from the PAM window to another window. The window may be specified by entering a pound sign followed by the name of the window

The send command may be invoked by entering the command in the PAM command line or by pressing the send (f3) function key. Invoking the send command without specifying a string will send the complete path name of the *highlighted file name* in the PAM window to the specified window. The complete path name is followed by a carriage return. The result is the same as if you had typed the complete path name in the destination window yourself and pressed Return. Invoking the command without a window name will cause the window immediately behind the PAM window to appear with the menu shown below.



If you have more than one other window, you must select a window to complete the send command. Press Window OK (f5) or shuffle windows by pressing Next Window (f3). To cancel the send operation, and return to the PAM window, press Cancel Send (f8). If you only have one other window, PAM will complete the send operation to that window.

The `send` command is helpful when you're running an application that requests a file name. You can also send a character string to the current cursor position in another window.

For example, when MemoMaker prompts you for a file name for editing, you can quickly select the PAM window, highlight the name of the file that you want to edit, and `send` the file name to the MemoMaker window. Without further prompting, MemoMaker will open the specified file.

If there are no windows besides PAM showing in the display, then the `send` command is ignored.

Example

```
send
send Error in shell script. #ErrorWindow
send #MemoMaker
```

set_font

Name

`set_font`—set the character font in a window

Syntax

`set_font` *font_file_name window_name*

Location

Utilities disc in `/util/fonts` folder

Description

The `set_font` program enables you to change the font (or character set) of a window. Running the program causes the specified font file to be used for the character set in the specified window. With the `fedit` program, `set_font` enables you to create and access customized character sets.

A number of ready-made fonts are included on the Utilities disc, in the `/util/fonts` folder.

To view another font, try this:

1. Insert the Utilities disc in the built-in disc drive and open the `/util/fonts` folder.
2. Enter the `echo` command in the PAM command line to create a window named `test`. Type: `echo #test` (Return).
3. Save the `test` window with Save (key (f4) in the System menu).
4. Load a new font into the `test` window. Type in the PAM command line: `set_font block12x16 test` (Return). (Note the omission of the # symbol.)
5. When the `set_font` window reports that the specified font has been set, select the `test` window and type away!

Some application programs, such as HP Calculator, have their own fonts and load them into system memory when first started. A font will stay in system memory until no window is using the font any longer.

Several *system fonts* reside permanently in ROM. One of the fonts is loaded during the power-up procedure for the computer. To have your own font loaded during the power-on process, simply rename your font `6X8` in the top-most folder on a "boot" disc. Your "boot" disc usually contains your `Autost` and `.environ` files. If you don't have a disc containing these files, the disc containing your font file must be the first disc found on your system.

Note

To establish a new font for subsequent windows, change the font in the *PAM* window. Each new program started from PAM will inherit the current PAM font (unless the program loads its own font).

Examples

```
set_font cour9x15 test
set_font cour9x15 PAM
```

Limitation

The maximum size of any font is 16 dots by 16 dots per character cell.

See Also

`fedit`

source

Name

`source`—run a script file directly

Syntax

`source` *file_name*

Location

Built into PAM

Description

The `source` command is used to directly execute the commands in script files by PAM. PAM will not create a sub-PAM or shell to execute the script file. This can be used to execute script files that contain commands that affect the PAM environment, such as `printer_is` or `cd`.

Examples

```
source Autost
```

See Also

Chapter 9

Name

`status`—report status information

Syntax

```
status
status -help
```

Location

Utilities disc

Description

The `status` command provides information on:

- The amount of unused memory (in kilobytes).
- The names, identification numbers, window names, and running times (in elapsed *minutes* and *seconds*) of all programs currently executing.
- The names, drive names, and available storage (in kilobytes) of all discs currently mounted in the system.

Use `status` when you want to check:

- How much memory a particular program requires. First run `status` before running the other program. Then run `status` again while the other program is running. The difference in reported system memory is the amount used the the particular program.
- How much space you have left on a particular disc.
- The program number of a particular program. (This may be used with the `terminate` utility to stop a program.)
- How many programs you have running (if you're not certain that all program windows are visible).

...status

Example

```
status
```

Limitations

- The `status` program can take a long time to complete, particularly if you are using a variety of disc devices in your system.
- If you have only one disc drive and you want to check the status of another disc, you must either copy `status` to the target disc or to the root folder. For more information, refer to chapter 8.

See Also

`terminate`

Name

stopprint—stop printing the current file

Syntax

```
stopprint
```

Location

Built into PAM

Description

The stopprint command causes the current printer device to stop printing a file. Use stopprint after you've issued a PAM print command in order to cancel the print operation.

Example

```
stopprint
```

Limitations

- The stopprint command affects all the files currently queued-up for a single print command. If you've started print a number of times, then you must issue a series of stopprint commands if you want to cancel all the print commands.
- The internal printer maintains a 1K *print buffer* in which characters are stored before they're printed. After entering a stopprint command, you'll need to wait until the printer buffer is emptied (of up to 1,024 characters) before the printer actually stops printing.
- Affects only the active printer—if another printer is printing, you can either select the corresponding window and press (Stop), or else you can use printer_is to make that printer the active printer and then execute stopprint.

See Also

print
Chapter 4

terminate

Name

`terminate`—terminate the execution of a program

Syntax

`terminate` *program_number*
`terminate -help`

Location

Utilities disc

Description

The `terminate` utility will kill the program specified by the *program_number*. All resources used by the specified program, including memory space, will be reclaimed by the system.

Note that `terminate` may not eliminate the *window* in which the specified program runs. However, the window of the terminated program will be eliminated the next time the computer begins executing another program.

Normally, you won't need to use the `terminate` program—you can eliminate virtually any program using the Exit function key for that program, or else the (Stop) key on the keyboard (to issue a quit signal), or else the (Shift)(Stop) keystroke (to issue a kill signal).

In those situations when you want to use `terminate`, run the `status` program first to determine the *program_number* of the desired program. Then include the *program_number* after typing `terminate` in the PAM command line.

Examples

```
terminate 32
```

See Also

`status`

Name

`time_of_day`—display/set time and date

Syntax

```
time_of_day
time_of_day hour:minute
time_of_day hour:minute month/day/year
time_of_day -help
```

Location

Utilities disc

Description

By itself, `time_of_day` causes a small digital clock window to appear in the upper right corner of the display. The clock will be updated every minute. Only one clock can be started.

If you specify the *hour*:*minute* parameters, the clock will be set to that time—the date will not be affected. Options:

- *hour*:*minute*—sets the clock according to 24-hour notation.
- *hour*:minute `am`—sets the clock to *am* time.
- *hour*:*minute* `pm`—sets the clock to *pm* time.

To set the *date*, include the *month*/*day*/*year* parameters after the *hour*:*minute* parameters.

Note the time displayed in the clock window is dependent on the `TZ` (time zone) variable in the current PAM environment. If no environment file has been read since you've powered up the computer, then the time zone variable will default to Eastern Standard/Eastern Daylight Savings time. To find out what the current time zone setting is, run the `date` command from the Utilities disc.

The reading in the clock window will automatically convert to and from local standard and daylight time during the year.

Examples

```
time_of_day
time_of_day 9:30
time_of_day 3:30 pm
time_of_day 15:30  10/18/1985
time_of_day 3:30 pm 10/18/1985
```

See Also

getenv
/hp-ux1/hpux1.doc/date.doc on the Commands 1 disc

Name

unmount_disc—unmount a disc from the file system

Syntax

unmount_disc *drive_name*
unmount_disc -help

Location

Utilities disc

Description

The unmount_disc program enables you to remove (or "unmount") flexible discs from external disc drives.

Caution

Do not remove a flexible disc from an external disc drive without first executing the unmount_disc program. Otherwise, the integrity of the data on the flexible disc may be destroyed.

Running the unmount_disc program ensures that all information shared between system memory and the disc is up-to-date and that no program is currently using any of the files on the disc.

...unmount_disc

Notes on `unmount_disc`.

- You *don't* have to use `unmount_disc` to remove a disc from the internal disc drive.
- You won't be allowed to unmount a disc that has an *open file* on it. For example, if you're editing a file that resides on the flexible disc and try to unmount the disc, the `unmount_disc` program will report that the disc can't be properly removed while the file is still open. If you want to edit a file, then write to the same disc, be sure the disc has write-permission before you open the file for editing. If you need to unmount an disc with an open file, save the file in `/tmp` then close the file, or stop the program that is using the file, and try again.
- You won't be allowed to unmount a disc if the open folder is on that disc. In such a case, open a folder on another disc and then try again.

After you've unmounted one disc and inserted another flexible disc in its place, you should run the `mount_disc` program to add the new disc to your file system.

Examples

```
unmount_disc D000
unmount_disc D001
```

Limitation

You can unmount only one disc at a time.

See Also

`mount_disc`

`scan_discs`

Name

verify_disc—verify the information on a disc

Syntax

verify_disc

```
verify_disc drive_name
verify_disc -help
```

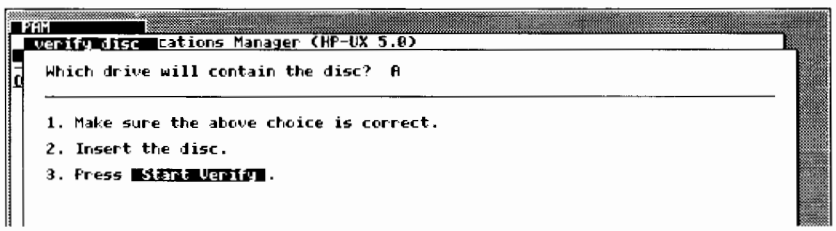
Location

Utilities disc

Description

The verify_disc program checks the integrity of the file system on a specified disc, attempts to locate misplaced files, and displays statistics on the number of files and folders on the disc as well as the amount of free space on the disc.

If you start verify_disc without specifying any parameters, you'll create the following program window:



Note that the verify_disc window is ready to verify the disc in the built-in disc drive.

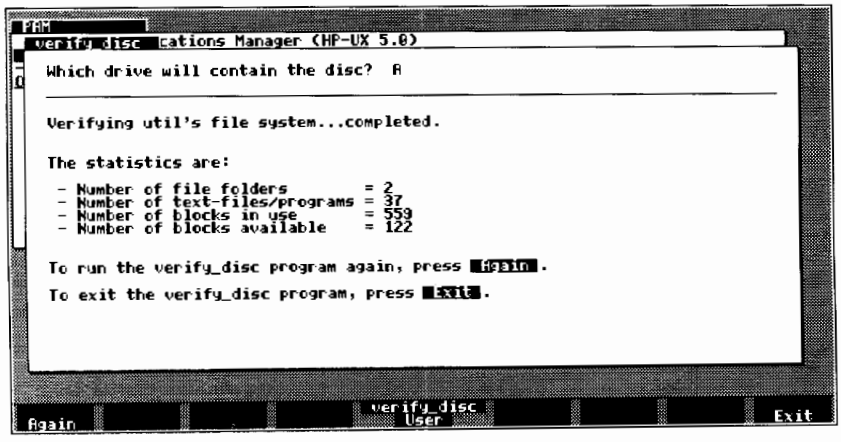
If you specify a *drive_name* before running the program, then the `drive` field will be set to that value when the program starts.

The following function keys control the `verify_disc` program:

Function Keys for `verify_disc`

Function key	Description
(f1) Start Verify	Causes the program to examine the disc in the specified drive.
Again	Returns to the initial screen and menu.
(f4) Change Drive	Enables you to specify a different *drive_name* where the disc is inserted. Pressing the key moves the cursor to the `disc drive` field and clears that field. Type in the name of the disc drive, such as `D000`, and press (Return). To restore the field to its previous value, backspace over the field and press (Return) when the field is blank.
(f7) Help	Creates a "help" window, separate from the `verify_disc` window, in which you can read usage information about the program.
(f8) Exit	Ends the program, cancels the current verify operation (if in progress), and eliminates the `verify_disc` program window.

Here is a typical sample of output after `verify_disc` has completed verifying a disc in the internal disc drive.



The amount of used and available disc space is measured in *blocks*, equal to 1,024 bytes.

After completing a verify operation, you can examine another disc. Press Again (f1) then follow steps 1 through 3 in the `verify_disc` window. To end the program, press Exit (f8).

If, for some reason, `verify_disc` finds a file that is not correctly associated with any folder on the disc, then the program will attempt to recover the file. The `verify_disc` utility will:

- *Rename* the file with a numeric code.
- *Move* the file to the outermost folder on the disc.

Afterwards, you can determine the identities of any "repaired" files and move the files back to their original folders.

Examples

```
verify_disc
verify_disc D000
```

Limitations

- The Back space key is the only editing key that you can use to edit any of the fields in the `verify_disc` window.
- The execution time of `verify_disc` can be lengthy, especially for hard discs.

view

Name

`view`—view the contents of a file

Syntax

`view` *file_name ...*
`view` *file_name ...* *#window_name*

Location

Built into PAM

Description

The `view` command enables you to view the contents of text files. The effect is identical to highlighting the *file_name* of the desired file and pressing View (f1).

The view window is associated with two function keys:

Function Keys for the View Window

Function Key	Description
f1 More	Causes PAM to display the next windowful of text.
f8 Quit	Causes PAM to eliminate the view window.

Note that pressing *space bar* has the same effect as pressing More (f1). Also, pressing the Q key (for lowercase q) has the same effect as pressing Quit (f8).

The view window will remain in the display until you eliminate it with Quit, Q, or Stop.

Using `view` is a convenient way of displaying the contents of *script files*. Script files appear in the PAM window as *program files*, but actually consist of lines of text.

You can specify a *window_name* parameter after the `view` command name in order to name the resulting `view` window.

You can also "pipe" the output of one program into the `view` command using the vertical bar (|) symbol in the PAM command line. For example, entering `ls -al | view` causes the `ls` program on the Commands 1 disc to direct its output to a `view` window.

Note that you can view other types of files, including folders, but the displayed information will probably appear as a jumble of characters.

Examples

```
view Memo-3-31
view One Two ../Three
view ThisFile #ThatWindow
ls -al | view
```

Limitation

Entering `view` by itself causes an error message.

See Also

Chapter 4
Chapter 4 of the Tutor

A User's Information

When Things Go Wrong

If your Integral PC does not respond as expected when you turn on the power, read the following information for help. If your computer fails *after* you have turned on the power, refer to "Running the Diagnostic Program" below.

Try This First

Is the power on? Are the voltage switch and fuse correct? If the power does not appear to be on, verify that the voltage setting is correct on the computer. The *Setup Guide* shows you how to verify that your computer's voltage setting matches the voltage of your ac power source (by observing the type of power cord that fits your power outlet). The voltage setting must match the voltage of your ac power source. (Types of power cords are also shown under "Changing the Line Voltage" later in this appendix.)

Warning

To avoid damage to the computer, check the voltage selector switch before applying power.

To avoid electrical shock to yourself and damage to the computer, use only the power cord specified by Hewlett-Packard for your area. Any replacement cord must have the same polarity as the original.

Your computer must have the correct power cord, voltage setting, and fuse for the ac power source in your area. If the power cord and voltage setting are correct, you can assume that the fuse is also correct.

You may need to change the setting of the voltage switch or change the fuse if:

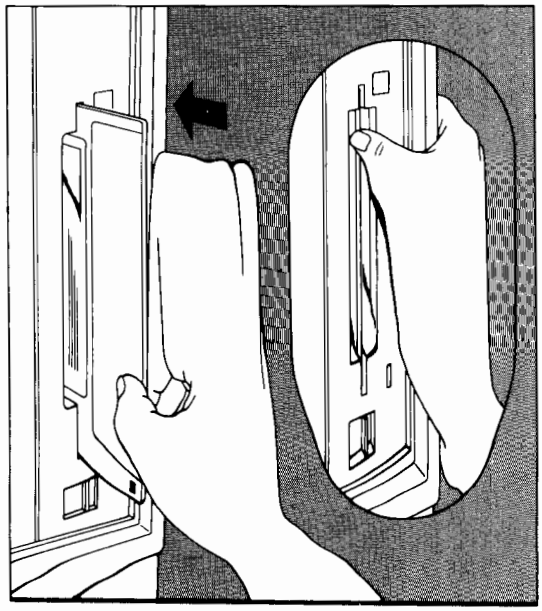
- The voltage switch isn't set to the voltage of your ac power source. In this case, you must change the setting of the voltage switch and check that the correct fuse is installed.
- The power cord with your computer isn't correct for your area. In this case, you must check the setting of the voltage switch and check that the correct fuse is installed.
- You travel with your computer to a country with a different power system. If you need to use a different power cord, you may also need to change the voltage setting and the fuse.

"Changing the Line Voltage" and "Changing the Fuse" later in this appendix describe the procedures for changing the setting of the voltage switch and for checking and changing the fuse.

Are you unable to insert the Tutor disc? The computer is shipped with a dummy disc inserted to protect each disc drive. Make sure you've ejected the dummy disc before you try to insert the tutor disc. You should save the dummy disc with the packing carton in case you need to ship your computer—the disc protects the drive in case the computer is dropped.

Does the Tutor program fail to start? When you insert the Tutor disc and then turn on the computer, the computer reads the Tutor program from the disc (which takes a minute or two) and then starts the Tutor program. If the Tutor program didn't start automatically, make sure that:

- You inserted the correct disc.
- You inserted the disc *before* you turned on the computer.
- You inserted the Tutor disc correctly and completely—the label should to the left and the metal shutter should face inward. (Check the orientation of the disc in the illustration below. Push the disc in until it clicks into place.)



- The computer has power. (Make sure that there is power to your power outlet; that the power cord is connected to both the power outlet and the computer; that the power button is depressed.)

Does the keyboard (or mouse) have no effect? If pressing keys on the keyboard seems to have no effect, or if rolling the mouse doesn't move the display pointer, follow these steps:

1. Switch off the computer (the button should be flush with the case).

Caution 

Do not connect or remove the keyboard or mouse cables when the power is on. Doing so may damage the peripheral device.

---

2. Make sure the keyboard (and mouse) are plugged into the HP-HIL connectors at the front of the computer. (If you have other HP-HIL peripherals, refer to *Connecting Peripherals to Your Integral PC* for instructions about where to connect them.)
3. Switch on the computer (the button should be recessed).

Remember that the computer scans for devices on the HP-HIL loop only at the time you switch on the power. If you connect a keyboard or mouse when the computer is already on, the peripheral will not be recognized and will have no effect.

Other problems? The computer tests itself each time you switch it on. If it finds a problem, it will display an error message in the copyright window that appears when you turn on the computer. If this happens, note the message and look it up under "Power-On Messages" in appendix E. Refer to the *Support Guide* for information about obtaining service. If the computer appears operational but won't run the Tutor program, you can test the computer hardware by running the diagnostic tests included on the Diagnostic disc. Instructions for using the Diagnostic disc are included under "Running the Diagnostic Program" below.

If the computer passes the diagnostic tests but still won't run the Tutor program, refer to the *Support Guide* for information about getting help.

Verifying That Your Printer Works

Occasionally, you may suspect that your computer is not communicating with your printer. When this happens, try one of the following procedures. Before you try to verify that an external printer is working, check to be sure that it's properly connected to your computer and that it has passed its own self-test. If you suspect that the problem is in your computer, refer to "Running the Diagnostic Program" below.

To verify that your built-in printer works:

1. Make sure that the computer is off (the power button is flush with the case).
2. Hold down the (FF) button or the (LF) button on top of the computer.
3. Switch on the computer. (Don't release the (FF) or (LF) button yet.)

4. After a few seconds, release the FF or LF button.

   The computer tests the printer's electronics and prints one page (66 lines) of a programmed set of examples.

If the printer fails to print the test page, check for problems as follows.

- Make sure you've correctly loaded paper in the printer.
- Make sure that nothing is blocking the movement of the print head carriage.
- Check the print head cartridge. (Refer to "Improving Print Quality" later in this appendix.)

Then repeat the four steps above. If you've correctly loaded paper and the carriage is free to move, but the printer still fails to print the test page, the printer failed the test. In this case, refer to the *Support Guide* for information about obtaining service for your computer.

To verify that your external printer works:

1. Connect your printer. (Refer to *Connecting Peripherals to Your Integral PC.*)
2. Turn on the printer, then turn on the computer.
3. Use the `printer_is` utility to configure your printer. (Refer to chapter 10 and to *Connecting Peripherals to Your Integral PC.*)
4. Print a data file. For example, you can do this by inserting the Utilities disc and typing
   `print README` Return.

If nothing happens, check the following:

- Did you configure the printer using the `printer_is` utility? (Refer to chapter 10 in this manual or to "Configuring Your System" in *Connecting Peripherals to Your Integral PC.*)
- Is your printer connected properly?
- If your printer uses an HP-IB interface, is its address switch set to a unique number?

- Did your printer pass its self-test?
- Did you turn on the printer before you turned on the computer?
- Did the contents of the Utilities disc appear when you inserted the disc? If they didn't, type `/util` (Return), then try printing the `README` file.

If your printer has a raster graphics capability, you can also check it by pressing (Shift)(Print) to print a copy of the computer display. If your printer still does not work, refer to the troubleshooting section in the owner's manual for your printer.

Verifying That Your Plotter Works

Occasionally, you may suspect that your computer is not communicating with your external plotter. Before you try to verify that your system is working, check to be sure that your plotter is properly connected to your computer and that it has passed its own self-test. If you suspect that the problem is in your computer, refer to "Running the Diagnostic Program" below.

To verify that your external plotter works: Use HP-UX Technical BASIC or an application program that uses the plotter to send data (HP-GL commands) to your plotter. If your plotter does not respond properly, check the following:

- Did you configure the plotter using the `plotter_is` utility? (Refer to chapter 10 in this manual or to "Configuring Your System" in *Connecting Peripherals to Your Integral PC.*)
- Is your plotter connected properly?
- If your plotter uses an HP-IB interface, is its address switch set to a unique number?
- Did your plotter pass its self-test?
- Did you turn on the plotter before you turned on the computer?

If your plotter still does not work, refer to the troubleshooting section in the owner's manual for your plotter.

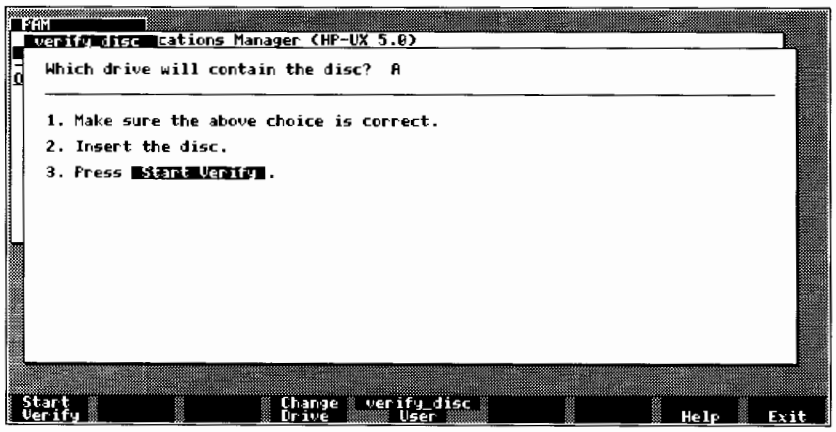
Verifying Your File System

You can use the `verify_disc` utility to verify the integrity of a mass-storage file system. You can verify any flexible or fixed disc that has been formattted.

To verify a disc:

1. Insert the Utilities disc.
2. Start the `verify_disc` utility. Do this by pressing the cursor keys to highlight the `verify_disc` program, and then pressing Start (f1). (Refer to chapter 2 for details about starting a program.)

   The following window will be displayed:



3. If you're *not* going to use the built-in disc drive, press Change Drive (f4). Then enter the device name of the desired disc drive. (Refer to "Disc Drive Names" in chapter 10 for information about drive names.)
4. Press Start Verify (f1). After checking the file system, the `verify_disc` program will display the message `Verify completed.` and statistics about the file system. If the program finds a file that is not correctly associated with any folder on the disc, the program will attempt to recover the file.
5. For more details, press Help (f8) or refer to `verify_disc` in chapter 11.
6. Press Exit (f8).

Running the Diagnostic Program

Every time you turn on the computer, a self-test automatically checks RAM and ROM. The diagnostic program provides an additional method for testing the operation of your computer.

Note

The diagnostic program clears the contents of computer memory (including the electronic disc). Do not run the tests if you are in the middle of work you haven't yet saved.

The diagnostic program can be used to test the following components:

- Keyboard.
- Display.
- Speaker.
- Random-Access Memory (RAM).
- Real-Time Clock.
- Internal Disc Drive.
- Built-In HP-IB Interface.
- I/O Interfaces and Memory Modules.
- Built-In Printer.

If you suspect one or more of the components listed above has failed, run the diagnostic program. This will give you a quick "confidence-level" check of the system. If you still suspect that your computer has failed, but the diagnostic program has not detected an error, record the circumstances under which the failure occurred and contact your authorized HP service representative listed in the *Support Guide*.

To run the diagnostic program:

1. Switch off the computer and peripherals.

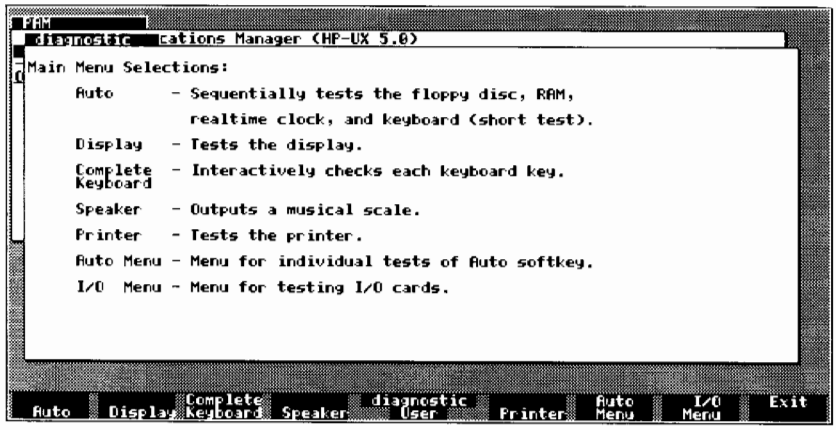
Caution 

If you proceed, you will lose all other programs currently executing on your computer as well as all data stored on the electronic disc. Be sure to stop all programs and copy important data currently on the electronic disc to the built-in disc or another disc before continuing.

---

2. Disconnect your peripherals and remove all interfaces and memory modules. Don't disconnect the keyboard. (After you have tested your computer, you can reinstall and test the I/O modules as described below.)
3. Insert the Diagnostic disc in the built-in disc drive.
4. Turn the computer on.
5. Press Start (f1) to start the `diagnostic` program.
6. Press Cont (f1) to proceed with the test.

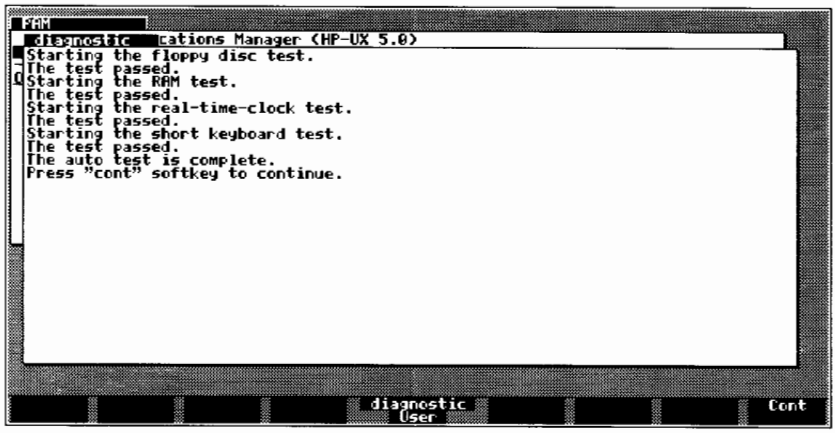
   The display shows:



7. Press Auto (f1) to select the automatic test.

The diagnostic program tests the computer RAM, clock, keyboard, and built-in disc drive. These tests require no participation on your part. As each test is completed, the program displays the result.

If all tests are completed without problems, the display shows:

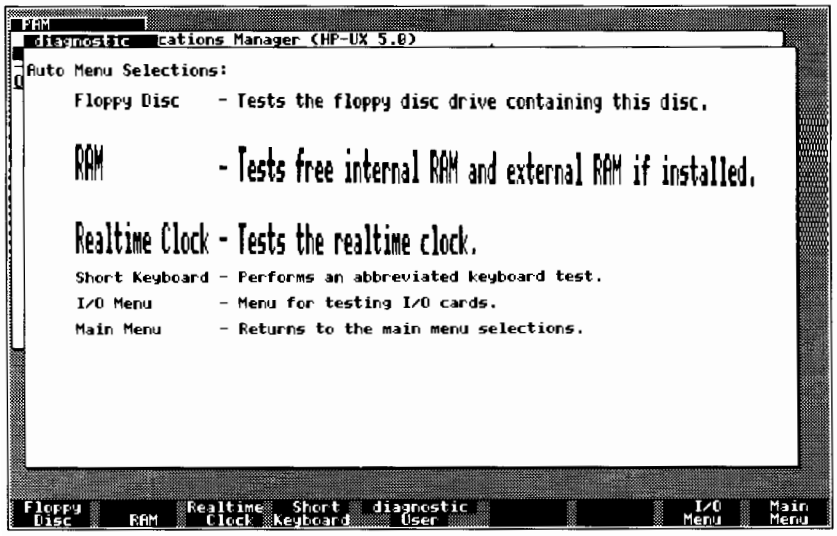


If any test discovers a problem with your computer, an error message specifying the problem is displayed. In this case, write down the error message. Then refer to the *Support Guide* for information about obtaining service. When you send your computer for servicing, be sure to include the results of the diagnostic test.

8. Press Cont (f8) to return to the main menu.

9. If you want to rerun just one of the tests from the Auto test (built-in disc test, RAM test, clock test, or keyboard test), use the Auto menu:

a. Press Auto Menu (f6). The display shows:



b. Press the function key for the desired test. For example, press RAM (f2) to run the RAM test.

c. Note the test results.

d. Press Main Menu (f8) to return to the main menu.

10. If you think there is a problem with the display, keyboard, or speaker, press the function key on the main menu for the desired test. These tests require your participation. Follow the instructions that are displayed at the start of each test. If your display isn't working, try checking the power cord of the computer.

If any test indicates a problem with the computer, first repeat the test to confirm the problem. If you can confirm the problem, note the symptoms. Then refer to the *Support Guide* for information about obtaining service. When you send your computer for servicing, be sure to include your note of the symptoms.

11. Press Exit (f8) to end the diagnostic program. Remove the Diagnostic disc, then turn off the computer.

To test a plug-in memory module: After testing the computer as described above, you can test memory modules one at a time.

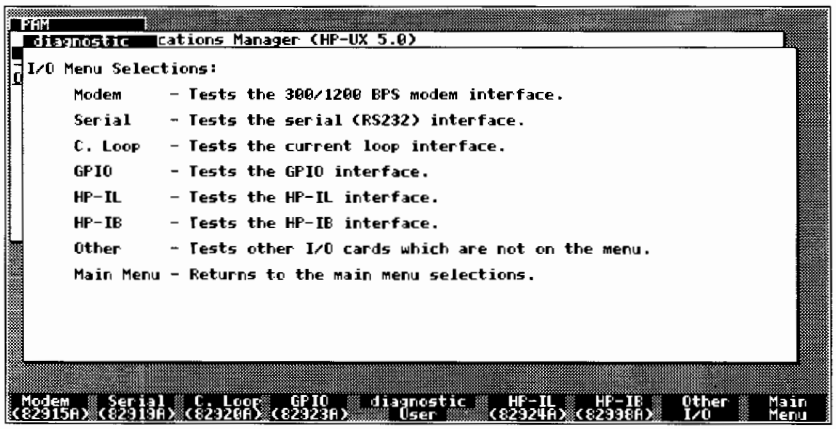
1. Remove the Diagnostic disc and turn off the computer.
2. Install the memory module in a computer port.
3. Turn on the computer.
4. Insert the Commands 1 disc.
5. Start the `status` program by pressing the cursor keys to highlight the `status` program and then pressing Start (f1). (Refer to chapter 2 for more information about starting a program.)
6. Note the amount of free memory displayed by the `status` program. The free memory for *only the computer* is 264K bytes if memory has been cleared before starting (such as by the diagnostic program). If you've installed a memory module, the amount of free memory should be increased above 264K by about the module size. If the free memory is only 264K while the memory module is installed, either the module is bad or the computer port is bad.
7. Press Stop to end the `status` program.
8. Remove the Utilities disc.
9. Insert the Diagnostics disc in the built-in disc drive.
10. Press Start (f1) to start the `diagnostic` test.
11. Press Cont (f1) to proceed with the test.
12. Press Auto Menu (f6).

14. Note the test results.
15. Press Main Menu (f8) to return to the main menu.

To isolate a problem, try another I/O port with the same module, or the same I/O port with a different module.

To test a plug-in interface: After testing the computer as described above, you can test plug-in interfaces one at a time.

1. Remove the Diagnostic disc and turn off the computer.
2. Install the interface in a computer port.
3. Prepare the interface as follows:
   - For an HP 82924A HP-IL Interface, connect an HP-IL cable between the IN and OUT connectors to form a loop.
   - For an HP 82998A HP-IB Interface, connect an HP-IB cable between the plug-in HP-IB interface and the built-in HP-IB interface.
   - For all other interfaces, disconnect all cables.
4. Insert the Diagnostics disc in the built-in disc drive.
5. Turn on the computer.
6. Press Start (f1) to start the `diagnostic` test.
7. Press Cont (f1) to proceed with the test.
8. Press I/O Menu (f7). The display shows:



9. Press the function key for the desired test.
10. Note the test results.
11. Press Main Menu (f8) to return to the main menu.

To isolate a problem, try another I/O port with the same interface, or the same I/O port with a different interface; also try a different cable.

Maintaining the Computer

This section describes the procedures you'll need for maintaining your computer:

- Cleaning the computer.
- Traveling with the computer.
- Changing the fuse.
- Changing the line voltage.
- Changing the print head.
- Improving the print quality.
- Caring for the built-in disc drive.
- Clock accuracy.
- Serial number.

Cleaning Your Computer

You should clean the computer regularly using a damp lint-free cloth. Be sure *not* to use anything abrasive on the display. Use of scouring powders, paper towels, or special cleansing chemicals may permanently damage the electroluminescent panel.

Traveling With Your Computer

Traveling by air? If you take your computer on an airplane, insert the dummy (cardboard) disc in the built-in drive. Remove the print head and carry it separately in a plastic bag. The reduced pressure in an airplane can cause the print head to leak.

Do not check the computer as baggage! If you can't carry the computer with you in the plane, you must pack it for shipping.

Shipping the computer? Remember to insert the dummy (cardboard) disc in the drive and be sure to use the original packing carton for your computer. If you've lost the original packing carton, contact your Hewlett-Packard dealer.

Traveling internationally? You may need a different power cord to use your computer in another country. Be sure to use only the power cord specified by Hewlett-Packard. If you change the power cord, refer to the next section for information about the changing the voltage switch and fuse.

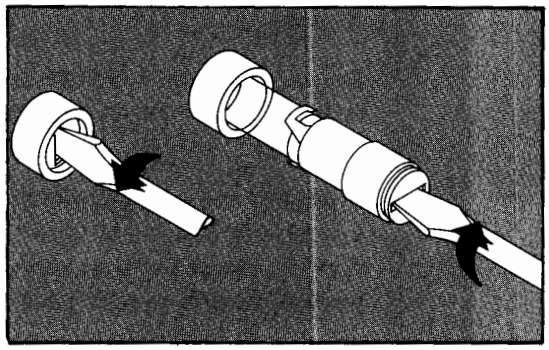
Changing the Fuse

The fuse *must* be selected according to the voltage of the power source. (The voltage selector switch must also be set accordingly.) If the fuse "blows" or if you change the voltage switch setting, you should change the fuse.

To change the fuse:

1. Turn off the power and unplug your computer at the wall outlet.
2. Turn the computer around so that you are facing the rear panel.

3. Use a flat-bladed screwdriver to remove the existing fuse.



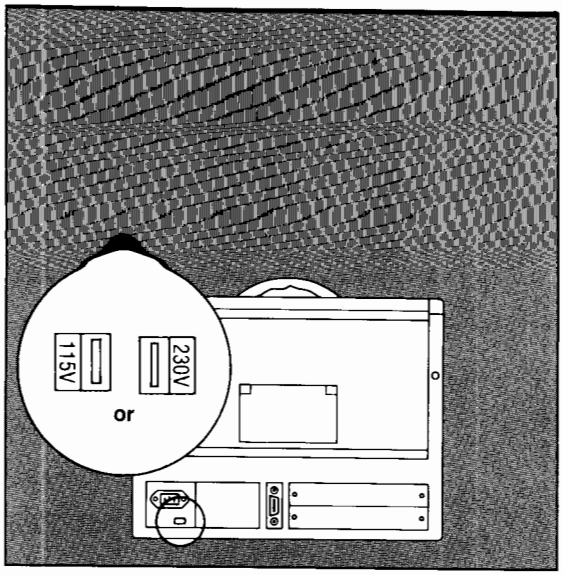
4. Insert a new fuse in the fuse holder. Use one of the following fuses:
   - 1500 mA (for 115V operation).
   - T800 mA (for 230V operation).
5. Insert the fuse and fuse holder into the receptacle. Then use a flat-blade screwdriver to turn the fuse cap clockwise until it locks in place.

Changing the Line Voltage

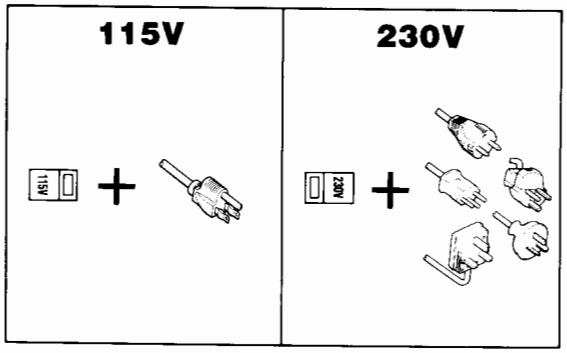
Whenever you intend to change the voltage of the power source for the computer, you *must* change the voltage selector switch accordingly. (You must change the fuse also, as described above.)

To change the line voltage:

1. Turn the power off.
2. Turn the computer around so that you are facing the rear panel.
3. Use a flat-blade screwdriver to slide the voltage select switch to the right for 115V, or to the left for 230V operation.



4. Select the correct type of grounded line cord and plug for your ac line voltage and wall outlet.



Changing the Print Head

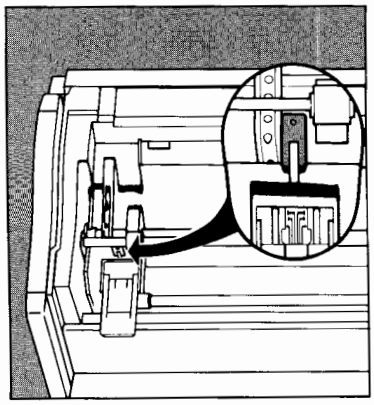
The first symptom of an insufficient ink supply is usually poor print quality for dense images, such as boldface characters, while print quality for normal characters is still good. If the ink bladder inside the print head cartridge is substantially collapsed, you need to change the print head. The *Setup Guide* contains some helpful illustrations.

Warning

The ink in the print head cartridge contains diethylene glycol, which is harmful if swallowed. Keep new and used cartridges out of the reach of children.

To change the print head:

1. Open the printer lid.
2. Close the printer arm.
3. Open the print head cartridge latch.
4. Remove the old cartridge. (Grasp it by its top ridge and pull up.)
5. Open the printer arm.
6. Note the location of the ink absorber and its placement in the ink absorber holder.

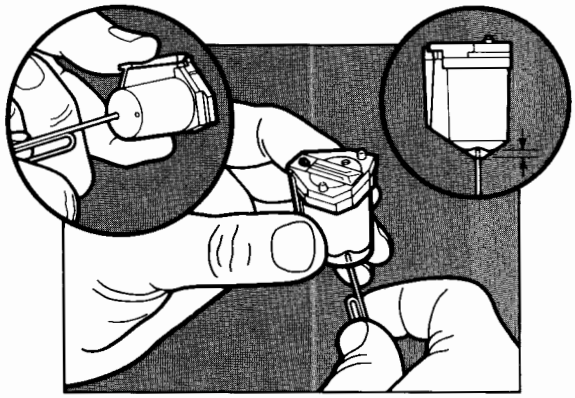


7. Remove the old ink absorber. (Using a tissue, grasp the top of the absorber and pull up.)
8. Discard the old cartridge and ink absorber.
9. Remove the new print head cartridge and ink absorber from the container. *Do not touch the metallic surface of the cartridge.*
10. Install the new ink absorber. (Grasp the ink absorber at one end with the absorbent side facing the front of the computer, then push the ink absorber into its holder.)
11. Close the printer arm.
12. Install the new cartridge. (Grasp the cartridge by its top ridge and lower it into its holder.)
13. Close the print head cartridge latch.
14. Close the printer lid.

If dots are missing in the printed output, clean the new print head according to the instructions in the following section. If the new print head stops printing after the first few lines, it may have lost its prime. A shock to the print head during shipping can prevent the ink from flowing properly. Use the following procedure to fix this problem.

To prime the print head:

1. Hold the cartridge with the metallic end up and the conical end down.
2. Insert the end of a straightened paper clip into the hole in the center of the conical end.



3. Push up gently until a bead of ink appears on the metallic surface, then remove the paper clip.
4. Wipe the silver-colored metallic surface with a soft tissue. Avoid wiping the gold-colored metallic surface.

Improving Print Quality

The printer should produce sharp, consistent printed output. The following conditions affect print quality:

- Use paper intended for Hewlett-Packard ThinkJet printers. Using other types of paper can reduce sharpness, cause smearing because of slow absorption, and can clog the print head with paper dust.
- The print head must be clean. Instructions for cleaning the print head appear below.
- The print head must contain sufficient ink. If boldface characters look irregular while normal characters look good, inspect the ink bladder inside the cartridge. If the ink bladder is substantially collapsed, replace the print head according to the instructions in the previous section.

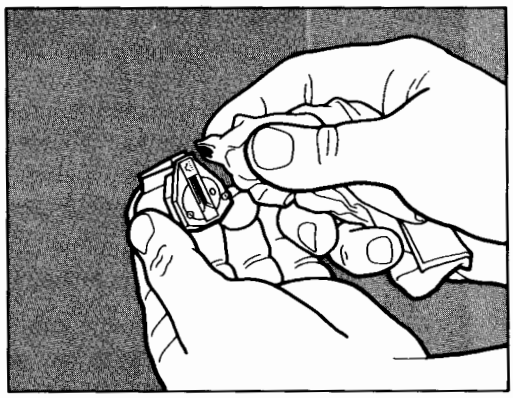
Dried ink, paper dust, or other contaminants can clog the print head. Follow the instructions below for cleaning the print head. If you're unfamiliar with the printer, look at the *Setup Guide* for some helpful illustrations.

Warning

The ink in the print head cartridge contains diethylene glycol, which is harmful if swallowed. Keep new and used cartridges out of the reach of children.

To clean the print head:

1. Open the printer lid.
2. Close the printer arm.
3. Open the print head cartridge latch.
4. Remove the cartridge. (Grasp it by its top ridge and lift up.)
5. Wipe the silver-colored metallic surface with a soft tissue. Avoid wiping the gold-colored metallic surface. If the print head has been inactive for a prolonged period, dampen the tissue with water before wiping.



6. Install the cartridge. (Grasp the cartridge by its top ridge and lower it into its holder.)
7. Close the print head cartridge latch.
8. Close the printer lid.

Caring for the Built-In Disc Drive

Your disc drive does not require attention under normal use. If the diagnostic program indicates a malfunctioning disc drive, refer to the *Support Guide* for information about repair service.

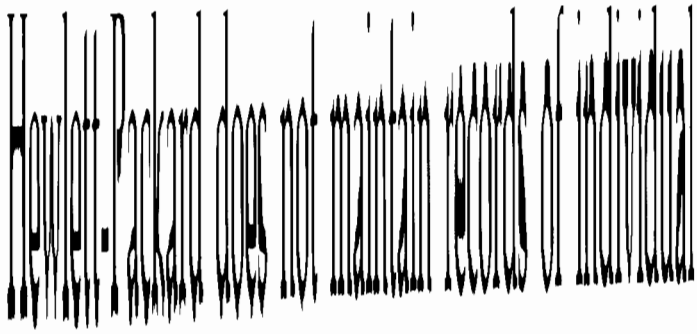
Your computer is shipped with a dummy (cardboard) disc inserted in the disc drive. This dummy disc protects the disc drive from damage in case of sudden motion or impact. Store this disc in the space next to the printer controls, and remember to insert the disc in the drive before you transport the computer. This will protect the drive from damage in case you drop the computer.

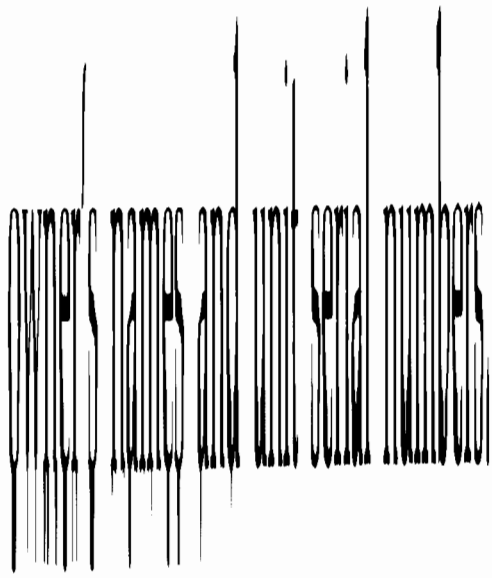
Clock Accuracy

The internal real-time clock is typically accurate to within 30 seconds per month, with an accuracy of 120 seconds per month for worst-case operating temperatures. The quartz crystal in the clock is affected by temperature, physical shock, humidity, and aging. Optimum accuracy is achieved at 20° to 50°C (68° to 86°F). An extreme change in environment can cause the clock to require resetting using the `time_of_day` program.

Serial Number

Each Integral PC carries an individual serial number on the rear panel. You should keep a separate record of this number. Should your unit be lost, the serial number is often necessary for tracing and recovery, as well as for insurance claims.





Specifications

The following list gives detailed specifications for the Integral PC.

Physical Properties.

- Width: 425 mm (16.7 inches).
- Height: 331 mm (13.0 inches).
- Depth: 215 mm (8.5 inches).

- Two I/O ports (expandable to 10 I/O ports using two bus expanders).
- Maximum total power to both I/O ports: 9 watts.
- Two interfaces or memory modules are supported: one can draw up to 6 watts (either port); the other, 3 watts.

Display.

- 255-by-512 pixel electroluminescent display.
- Up to 85 characters by 31 lines (default of 80 characters by 24 lines).
- Viewing angle: greater than 120 degrees in a horizontal plane.
- Display tilt: adjustable in 2-degree steps from 5 to 17 degrees.
- 32K-byte dedicated bit-mapped display memory.

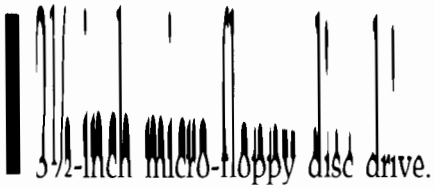
Printer.

- Thermal inkjet printer.
- 80 characters per line at 12 characters per inch (normal mode).
- Peak printing speed: 150 characters per second.
- Screen copy in 26 seconds.
- Roman8 character set.
- Four character sizes.
- Normal and bold character modes.
- Recommended paper: HP 92261M (500 single 8½ by 11-inch sheets) or HP 92261N (2500 fan-fold sheets).
- Plain paper is usable with lower print quality.

Print Head.

- Life: 820,000 characters (approximately 500 pages).
- Shelf life (in shipping container): 18 months at 25°C.
- Shelf life (outside shipping container): 6 months at 25°C.

Disc Drive.



- Burst transfer rate: 62.5K bytes per second.
- Encoding: Modified Frequency Modulation.
- Rotational speed: 600 revolutions per minute.
- Capacity: 630K bytes (at 256 bytes per sector, 16 sectors per track), 709K bytes (at 512 bytes per sector, 9 sectors per track), or 788K bytes (at 1024 bytes per sector, 5 sectors per track). Default is 709K bytes.
- Total tracks per surface: 80.
- Available tracks per surface: 77.
- Spare tracks per surface: 2.
- Surfaces per disc: 2.
- Interleave factor: 1.

Disc Media.

- Life: 3×10^6 passes per track.
- Read errors: one per 10^9 bits read.

Keyboard.

- 90-key condensed ITF (Integrated Terminal Family) keyboard.
- Detachable, low profile (30 mm high).
- Adjustable tilt, 0 or 8 degrees.
- Modified HP-HIL (Hewlett-Packard Human Interface Loop) keyboard interface with two HP-HIL connectors.

Speaker.

- Frequency range: 100 to 5000 Hz.

Real Time Clock.

- Accuracy: 120 seconds per month maximum error; 30 seconds per month typical error.
- Battery life: 6 years minimum; 12 years typical.

HP-IB Interface.

- Maximum transfer rate: 135K bytes per second (software limitations may reduce the rate).
- Conforms to IEEE-488 interface standard.

Environmental Limits.

- Computer:
  Operating temperature: 0° to 40°C (32° to 104°F).
  Storage temperature: −40° to 75°C (−40° to 167°F).
  Operating humidity: 5 to 95% RH at 40°C (104°F).
  Non-operating humidity: 90% RH at 65°C (149°F).
  Operating altitude: 0 to 4600 m (0 to 15,000 feet).
  Non-operating altitude: 0 to 15,300 m (0 to 50,000 feet).
- Disc media:
  Operating temperature: 10° to 40°C (50° to 104°F).
  Long-term storage temperature: 10° to 60°C (50° to 140°F).
  Transporting temperature: −40° to 60°C (−40° to 140°F).
  Humidity: 8 to 80% RH at 40°C (104°F).
- Print Head:
  Operating temperature: 10° to 40°C (50° to 104°F).
  Storage temperature: −20° to 60°C (−4° to 140°F).
  Humidity: 5 to 80% RH at 40°C (104°F).

Power Requirements.

- Voltage: 87 to 127 Vac (115 Vac line) or 195 to 264 Vac (230 Vac line).
- Frequency: 47.5 to 66 Hz.
- Power: 88W.

Fuse.

- 115V line voltage setting: 1500 mA fuse.
- 230V line voltage setting: T800 mA fuse.
- Nameplate on the back of the computer indicates proper fuse.

B Keyboard Maps

This appendix correlates specific keys on the keyboard with control sequences and extended characters. The map of keyboard control sequences indicates what control character or escape sequence is returned when the key is pressed. The extended character map indicates what characters are produced when (Extend char) and a character key are pressed simultaneously.

Control Sequences

The following table lists the control characters and escape sequences that are generated when you press certain keys in a normal alpha window. HP escape sequences are returned in HP mode, and ANSI escape sequences are returned in ANSI mode. To select HP mode, use the escape sequence `Ec&k0S`. To select ANSI mode, use the escape sequence `Ec&k1S`. To determine the character codes for control characters, refer to the U.S. ASCII character set in appendix C.

Keystroke	Control Character	HP Escape Sequence	ANSI Escape Sequence
Back space	Back space (BS)		
Clear display		&J	&[J
Clear line		&K	&[K
CTRL G	Bell (BEL)		
CTRL H	Back space (BS)		
CTRL I	Horizontal tab (HT)		
CTRL J	Line feed (LF)		
CTRL M	Carriage return (CR)		
CTRL N	Shift out (SO)		
CTRL O	Shift in (SI)		
CTRL Q	Device control 1 (DC1)		
CTRL S	Device control 3 (DC3)		
Delete char		&P	&[P
Delete line		&M	&[M
Enter		&d	&[d
Insert char (if off)		&Q	&[4h
Insert char (if on)		&R	&[4l
Insert line		&L	&[L
Next		&U	&[U
Prev		&V	&[V
Return	Carriage return (CR)		
Shift →		&F	&[F

Keystroke	Control Character	HP Escape Sequence	ANSI Escape Sequence
Shift Tab		&i	&[Z
Shift ▲		&S	&[S
Shift ▲ ◄		&&r1u1L	&[S &[@
Shift ▲ ►		&&r1u1R	&[S &[A
Shift ◄		&&r1L	&[@
			
Shift ▼ ◄		&&r1d1L	&[T &[@
Shift ▼ ►		&&r1d1R	&[T &[A
Shift ►		&&r1R	&[A
Shift Clear line		&G &K	&[2K
Shift Clear display		&h &J	&[2J
Tab	Horizontal tab (HT)		
►		&h	&[H
▲		&A	&[A
▲ ◄		&A &D	&[A &[D
▲ ►		&A &C	&[A &[C
◄		&D	&[D
▼		&B	&[B
▼ ◄		&B &D	&[B &[D
▼ ►		&B &C	&[B &[C
►		&C	&[C

B

Extended Characters

The following table lists all of the extended characters you can type directly from the keyboard. Extended characters are generated only while the active window is a normal alpha window. If no shifted character is indicated in the table, then the same character is produced whether or not (Shift) is pressed. The system accesses the extended characters from the Roman Extension character set (refer to appendix C).

Many accented characters in the Roman Extension character set can be produced by typing an extended accent character and then entering a normal (unaccented) character. To designate the accent character, press (Extend char) with one of the following keys: (I), (R), (T), (U), or (Y). This produces an accent that will appear above the next character entered. To produce á, for example, press (Extend char)(R) followed by (A). If a combination doesn't form a valid extended character, then the normal (unaccented) character is displayed.

Extended Character Keyboard Map

Key	Extended Character	Shifted Extended Character
,	<	
-	_	–
.	>	
/	_	
0	¿	
1	¡	
2	@	
3	#	
4	♯	
5	½	
6	^	
7	\	
8	[	‹
9	]	›
;	£	
=	±	
A	ą	Ą
B	■	
C	ç	Ç
D	đ	Đ
E	æ	Æ
F	ƒ	

Key	Extended Character	Shifted Extended Character
G	ğ	
H	¥	
I	~	
J	$	
K	¢	
L	£	
M	º	
N	ª	
O	ø	Ø
P	þ	Þ
R	´	
S	ß	
T	`	
U	¨	
V	§	
W	~	
X	š	Š
Y	^	
[	°	
]	ı	
`	«	»

C Character Set

The Roman8 character set consists of the standard U.S. ASCII character set and the Roman Extension character set. Each character in the set is assigned a character code with a decimal value from 0 through 255. The system uses these codes for identifying characters. Characters are normally produced from the keyboard by pressing the corresponding keys. (Refer to appendix B for keyboard mapping.)

The first half of the Roman8 character set (decimal values 0 through 127) is identical to the standard character set used on many other computer systems. The second half (decimal values 128 through 255) contains special characters, including those used by other languages.

US ASCII Character Set

C

ASCII Char.	Character Code			
	Dec	Binary	Oct	Hex
NUL	0	00000000	000	00
SOH	1	00000001	001	01
STX	2	00000010	002	02
ETX	3	00000011	003	03
EOT	4	00000100	004	04
ENQ	5	00000101	005	05
ACK	6	00000110	006	06
BEL	7	00000111	007	07
BS	8	00001000	010	08
HT	9	00001001	011	09
LF	10	00001010	012	0A
VT	11	00001011	013	0B
FF	12	00001100	014	0C
CR	13	00001101	015	0D
SO	14	00001110	016	0E
SI	15	00001111	017	0F
DLE	16	00010000	020	10
DC1	17	00010001	021	11
DC2	18	00010010	022	12
DC3	19	00010011	023	13
DC4	20	00010100	024	14
NAK	21	00010101	025	15
SYNC	22	00010110	026	16
ETB	23	00010111	027	17
CAN	24	00011000	030	18
EM	25	00011001	031	19
SUB	26	00011010	032	1A
ESC	27	00011011	033	1B
FS	28	00011100	034	1C
GS	29	00011101	035	1D
RS	30	00011110	036	1E
US	31	00011111	037	1F

ASCII Char.	Character Code			
	Dec	Binary	Oct	Hex
space	32	00100000	040	20
!	33	00100001	041	21
"	34	00100010	042	22
#	35	00100011	043	23
$	36	00100100	044	24
%	37	00100101	045	25
&	38	00100110	046	26
'	39	00100111	047	27
(	40	00101000	050	28
)	41	00101001	051	29
*	42	00101010	052	2A
+	43	00101011	053	2B
,	44	00101100	054	2C
-	45	00101101	055	2D
.	46	00101110	056	2E
/	47	00101111	057	2F
0	48	00110000	060	30
1	49	00110001	061	31
2	50	00110010	062	32
3	51	00110011	063	33
4	52	00110100	064	34
5	53	00110101	065	35
6	54	00110110	066	36
7	55	00110111	067	37
8	56	00111000	070	38
9	57	00111001	071	39
:	58	00111010	072	3A
;	59	00111011	073	3B
<	60	00111100	074	3C
=	61	00111101	075	3D
>	62	00111110	076	3E
?	63	00111111	077	3F

US ASCII Character Set (Continued)

ASCII Char.	Character Code			
	Dec	Binary	Oct	Hex
@	64	01000000	100	40
A	65	01000001	101	41
B	66	01000010	102	42
C	67	01000011	103	43
D	68	01000100	104	44
E	69	01000101	105	45
F	70	01000110	106	46
G	71	01000111	107	47
H	72	01001000	110	48
I	73	01001001	111	49
J	74	01001010	112	4A
K	75	01001011	113	4B
L	76	01001100	114	4C
M	77	01001101	115	4D
N	78	01001110	116	4E
O	79	01001111	117	4F
P	80	01010000	120	50
Q	81	01010001	121	51
R	82	01010010	122	52
S	83	01010011	123	53
T	84	01010100	124	54
U	85	01010101	125	55

ASCII Char.	Character Code			
	Dec	Binary	Oct	Hex
`	96	01100000	140	60
a	97	01100001	141	61
b	98	01100010	142	62
c	99	01100011	143	63
d	100	01100100	144	64
e	101	01100101	145	65
f	102	01100110	146	66
g	103	01100111	147	67
h	104	01101000	150	68
i	105	01101001	151	69
j	106	01101010	152	6A
k	107	01101011	153	6B
l	108	01101100	154	6C
m	109	01101101	155	6D
n	110	01101110	156	6E
o	111	01101111	157	6F
p	112	01110000	160	70
q	113	01110001	161	71
r	114	01110010	162	72
s	115	01110011	163	73
t	116	01110100	164	74
u	117	01110101	165	75

Char.	Character Code			
	Dec	Binary	Oct	Hex
	128	10000000	200	80
	129	10000001	201	81
	130	10000010	202	82
	131	10000011	203	83
	132	10000100	204	84
	133	10000101	205	85
	134	10000110	206	86
	135	10000111	207	87
	136	10001000	210	88
	137	10001001	211	89
	138	10001010	212	8A
	139	10001011	213	8B
	140	10001100	214	8C
	141	10001101	215	8D
	142	10001110	216	8E
	143	10001111	217	8F
	144	10010000	220	90
	145	10010001	221	91
	146	10010010	222	92
	147	10010011	223	93
	148	10010100	224	94
	149	10010101	225	95
	150	10010110	226	96
	151	10010111	227	97
	152	10011000	230	98
	153	10011001	231	99
	154	10011010	232	9A
	155	10011011	233	9B
	156	10011100	234	9C
	157	10011101	235	9D
	158	10011110	236	9E
	159	10011111	237	9F

Char.	Character Code			
	Dec	Binary	Oct	Hex
space	160	10100000	240	A0
À	161	10100001	241	A1
Â	162	10100010	242	A2
È	163	10100011	243	A3
Ê	164	10100100	244	A4
Ë	165	10100101	245	A5
Î	166	10100110	246	A6
Ï	167	10100111	247	A7
´	168	10101000	250	A8
`	169	10101001	251	A9
^	170	10101010	252	AA
¨	171	10101011	253	AB
~	172	10101100	254	AC
Ù	173	10101101	255	AD
Û	174	10101110	256	AE
£	175	10101111	257	AF
¯	176	10110000	260	B0
Ý	177	10110001	261	B1
ý	178	10110010	262	B2
°	179	10110011	263	B3
Ç	180	10110100	264	B4
ç	181	10110101	265	B5
Ñ	182	10110110	266	B6
ñ	183	10110111	267	B7
¡	184	10111000	270	B8
¿	185	10111001	271	B9
¤	186	10111010	272	BA
£	187	10111011	273	BB
¥	188	10111100	274	BC
§	189	10111101	275	BD
ƒ	190	10111110	276	BE
¢	191	10111111	277	BF

Roman Extension Character Set (Continued)

Char.	Character Code			
	Dec	Binary	Oct	Hex
â	192	11000000	300	C0
ê	193	11000001	301	C1
ô	194	11000010	302	C2
û	195	11000011	303	C3
á	196	11000100	304	C4
é	197	11000101	305	C5
ó	198	11000110	306	C6
ú	199	11000111	307	C7
à	200	11001000	310	C8
è	201	11001001	311	C9
ò	202	11001010	312	CA
ù	203	11001011	313	CB
ä	204	11001100	314	CC
ë	205	11001101	315	CD
ö	206	11001110	316	CE
ü	207	11001111	317	CF
Â	208	11010000	320	D0
Î	209	11010001	321	D1
Ø	210	11010010	322	D2
Æ	211	11010011	323	D3
å	212	11010100	324	D4
í	213	11010101	325	D5
ø	214	11010110	326	D6
æ	215	11010111	327	D7
Ä	216	11011000	330	D8
ì	217	11011001	331	D9
Ö	218	11011010	332	DA
Ü	219	11011011	333	DB
É	220	11011100	334	DC
ï	221	11011101	335	DD
ß	222	11011110	336	DE
Ô	223	11011111	337	DF

Char.	Character Code			
	Dec	Binary	Oct	Hex
Á	224	11100000	340	E0
Ã	225	11100001	341	E1
ã	226	11100010	342	E2
Đ	227	11100011	343	E3
đ	228	11100100	344	E4
Í	229	11100101	345	E5
Ì	230	11100110	346	E6
Ó	231	11100111	347	E7
Ò	232	11101000	350	E8
Õ	233	11101001	351	E9
õ	234	11101010	352	EA
Š	235	11101011	353	EB
š	236	11101100	354	EC
Ú	237	11101101	355	ED
Ÿ	238	11101110	356	EE
ÿ	239	11101111	357	EF
Þ	240	11110000	360	F0
þ	241	11110001	361	F1
‐	242	11110010	362	F2
¼	243	11110011	363	F3
¶	244	11110100	364	F4
¾	245	11110101	365	F5
¯	246	11110110	366	F6
¼	247	11110111	367	F7
½	248	11111000	370	F8
ª	249	11111001	371	F9
º	250	11111010	372	FA
«	251	11111011	373	FB
■	252	11111100	374	FC
»	253	11111101	375	FD
±	254	11111110	376	FE
▒	255	11111111	377	FF

D Controlling the Built-In Printer

The Integral PC computer incorporates a built-in ink-jet printer. This printer is the default active printer whenever you turn on the computer. This appendix describes the operation of the built-in printer.

The built-in printer can accommodate a variety of application needs, including graphics and the printing of formatted text. Control characters and escape sequences are used to select the various features available on the built-in printer. These control characters and escape sequences are normally inserted automatically by an application program, such as a text editor.

You have the option of sending specific control characters and escape sequences to the printer programmatically by embedding them in text files or creating programs that send them. By doing this, you put the printer in a specific mode before you use it for printing normal data files. All control characters and escape sequences that are used with the built-in printer are listed at the end of this appendix.

The built-in printer is assigned as the active printer at power on and can be accessed through the device file `/dev/lp`. The `print` command can be used to send output to whatever printer the `/dev/lp` file currently accesses.

To try out the information in this chapter, you will need an editor, such as the `vi` editor (appendix F), that can be used for entering program statements. Whatever editor you use must allow you to enter control characters and escape sequences without causing a control action in an alpha window. If you attempt to enter control characters and escape sequences with an editor that does not have this capability, they either will be ignored or will have unpredictable results. This is because the control characters used by alpha windows have a format similar to the printer control characters.

D

Control Sequences

The printer control sequences consist of control characters and escape sequences that perform certain actions and select certain modes that change how the printer responds to output data. The modes selected by the escape sequences remain in effect until the computer is turned off or until the printer is reset, but can be changed by sending other escape sequences. The following default modes are selected automatically for the printer when the computer is turned on:

- Normal print pitch of 12 characters per inch and 80 characters per line.
- Bold mode is disabled.
- Underlining mode is disabled.
- Line spacing is set at 6 lines per inch.
- Perforation skip mode is disabled.
- Page length is set at 66 lines.
- Text length is set at 60 lines per page.
- Wrap-around mode is enabled.
- Printing direction is bidirectional (text only).
- Graphics mode is disabled.
- Display functions mode is disabled.

Control Characters

Control characters are entered by pressing a specific key while the (CTRL) key is depressed. The entry for the back space character (BS) is (CTRL)(H), for example. Hold the (CTRL) key and press the specified key to enter a control character.

In certain applications, it is more convenient to specify a control character using its decimal value instead of pressing (CTRL) with another key. The decimal values of the specific characters that are returned by the computer when (CTRL) is pressed with another key are listed in appendix B. These decimal values can be used in functions, such as the BASIC CHR$ function, to designate control characters.

Escape Sequences

All escape sequences begin with the escape control character, shown as ←. Escape sequences are used for changing printer modes and specifying other functions. There are two types of escape sequences: two-character escape sequences and parameterized escape sequences (which contain more than two characters). Two-character escape sequences perform one function; parameterized escape sequences can specify several related functions.

Escape sequences are normally entered by pressing (ESC) followed by one or more keystrokes. The escape character, returned by (ESC), tells the printer that an escape sequence will follow. The next character determines whether a two-character escape sequence or a parameterized sequence follows. The printer interprets the correct number of characters after the escape character as part of the escape sequence. To enter an escape sequence that the printer can properly interpret, you must observe the following rules:

- Avoid the use of spaces in escape sequences. Spaces are treated as actual characters in the sequence and should not be used except in the value fields of parameterized escape sequences. Each space in a value field is interpreted as a zero.
- Avoid confusing uppercase and lowercase letters. Uppercase and lowercase versions of the same letter are not interchangeable.

In some applications, it is convenient to use the decimal values of the escape character (decimal 27) or other characters in an escape sequence instead of entering them directly from the keyboard. You can specify the decimal values for the characters in escape sequences using the BASIC `CHR$` function or a similar function in other programming languages. The decimal values for characters in escape sequences are listed in appendix C. All the characters shown for escape sequences in this guide are Roman8 characters.

Two-Character Escape Sequences. Two-character escape sequences are expressed as Et followed by a single character:

Et *character*

where *character* can be any character with a value from decimal 48 (0) through decimal 126 (~).

Parameterized Escape Sequences. Parameterized escape sequences are expressed as Et followed by a specific combination of characters. An optional value field in these escape sequence allows you to specify a range of settings, such as the number of lines on a page. Either of the following two general formats for parameterized escape sequences can be used:

Et *character1 character2 number CHARACTER3*
Et *character1 character2 CHARACTER3*

where *character1* is a single character that identifies the escape sequence as parameterized, and it can have a value from decimal 33 (!) through the decimal 47 (/).

character2 is a single character that specifies the type of control being performed. It's called the group character, and it can have a value from decimal 96 (') through decimal 126 (~).

number is a sequence of characters that specifies an optional decimal number. It's called the value field, and the number can range from −32,767 through 32,767. Each character in then number must be a digit 0 through 9, a space character, a plus sign, or a minus sign—they're the only characters allowed in the number.

CHARACTER3 is one uppercase character that indicates the end of the escape sequence. It's called the terminating character, and it can have a value from from decimal 64 (@) through decimal 94 (^).

Linking Escape Sequences. Parameterized escape sequences can be linked together to reduce the number of separate escape sequences needed to change specific modes of the printer. If you're linking several escape sequences, each of the linked sequences must use the same two characters following the Ec character.

When escape sequences are linked, the two "shared" characters are entered for only the first sequence. In addition, the ending characters of all sequences are changed to lowercase characters—except that the last character must remain uppercase.

For example, the following escape sequences can be used with the printer. They can be used separately, or they can be linked (because the two characters that follow Ec in each sequence are identical: &l).

Linking Escape Sequences

Escape Sequence	Description
Ec&l33P	Sets a page length of 33 lines.
Ec&l29F	Sets a text length of 29 lines.
Ec&l1L	Enables perforation skipping.
Ec&l33p29f1L	Sets a page length of 33 lines, a text length of 29 lines, and enables perforation skipping.

When the escape sequences are linked as Ec&l33p29d1L, the action caused by this linked escape sequence is equivalent to entering the three separate escape sequences, one after the other.

The remainder of this chapter describes the individual control characters and escape sequences that you can use with the built-in printer.

Printing Text

You can use the control sequences described below for controlling the printing of text.

Back Space

Pressing (CTRL)(H) or (Back space) returns the *back space* control character. This moves the current printer position to the left one character. If the current active position is already in the first printing position, it is not moved.

Line Feed

Pressing (CTRL)(J) returns the *line feed* control character. This moves the current printer position to the same character position on the next line. If perforation skip is enabled and the current active position is in the last line of the text area, the paper is advanced to the first line of the next page. Any data in the buffer is printed before the paper is advanced.

Form Feed

Pressing (CTRL)(L) returns the *form feed* control character. This advances the paper vertically so the current printer position is in the same character position on the first line of a new page.

Carriage Return

Pressing (CTRL)(M) returns the *carriage return* control character. This moves the current printer position to the first printing position on the same line. (*Don't* use the (Return) key for carriage return—it returns two characters: carriage return *and* line feed.)

Half-Line Feed

Pressing (ESC) followed by (=) advances the paper one-half of the current line spacing and leaves the current printer position at the same character position.

You can use half-line feeds to print superscripts and subscripts. For example, printing `1024↓=2` produces

1024_2

If you have a superscript or subscript in the middle of a line of text, you'll have to print it on a "half-feed" line by itself—the printer *can't* perform a reverse half-line feed.

Print Pitch

Four different print pitches are available on the printer. The escape sequence for selecting the print pitch is `Ec&k` *pitch_number* `S`. The number *pitch_number* selects the character width, or pitch. Print pitches and numbers are listed below.

Internal Printer Print Pitches

Mode	Characters Per Inch	Characters Per Line	Pitch Number
Normal (default)	12.0	80	0
Expanded	6.0	40	1
Compressed	21.3	142	2
Expanded-Compressed	10.7	71	3

Increasing the character pitch decreases the number of characters printed per line. The rightmost characters are truncated if you attempt to print excess characters. To print excess characters on the next line, you can enable wrap-around (described below).

Programming Example. The following samples can be printed with statements in a program by using escape sequences to specify different print pitches. Once a pitch is selected, the printer continues to print with the same pitch until another pitch is selected. Although only one print pitch is shown for each language, different pitches could be specified by changing the escape sequences. The statements are shown as if the terminal were in display functions mode when the escape sequences were entered.

```
This is compressed print.
This is normal print.
This is expanded-compressed print.
This  is  expanded  print.
```

Sample Program Print Statements

Language	Statement
BASIC	`100 PRINT "&k2S This is compressed print."`
C	`printf("&k0S This is normal print.");`
FORTRAN	`5 WRITE (6,10)` `10 FORMAT (" ", "&k3S This is expanded-compressed print.")`
Pascal	`writeln ('&k1S This is expanded print.');`

With C, FORTRAN, and Pascal, the standard output from the program should be directed to the built-in printer by specifying the device file `/dev/lp`. If the compiled file name is `a.out`, for example, then the command `a.out >/dev/lp` could be used.

Bold Print

Bold print mode is specified with two control codes. To enable this mode, use (CTRL)(N). Bold print mode is disabled with (CTRL)(O). At power on, bold printing is disabled. Bold print can be used for emphasizing a single word or for printing an entire file with darker-than-average type. Bold printing is slower than regular printing because the print head must pass over a bold line twice.

Specific **words** in this sentence are **highlighted.**
This entire line is printed in bold mode.

Underlining

In underlining mode, text is automatically underlined. To enable underline, the escape sequence is &d0. To disable underline, use &d@. Underline is disabled at power on. Any blanks in the specified text are also underlined. To avoid underlining blank spaces, put escape sequences around words or lines individually.

Words can be underlined.
Spaces may also be underlined: _____.

Mixing Modes

Modes for print pitch, bold print, and underlining can be combined to produce different character forms. Several examples of mixed print modes follow.

```
This combines expanded
type & bold face.
This combines expanded-compressed type
with underline mode.
```

Line Spacing

Line spacing affects the distance that the paper is moved when a line feed is executed. The escape sequence for six lines per inch is &l6D. To select eight lines per inch, use the escape sequence &l8D. Paper is moved 1/6 or 1/8 of an inch, respectively, before printing the next line. The default line spacing is 6 lines per inch.

Changing the line spacing does not affect the physical location of the top of form or the perforation skip area, but it *does* change the logical page length. At power on, the logical page length is 66 lines. If line spacing is changed to eight lines per inch, the logical page length increases to 88 lines—the physical length of the page doesn't change.

Perforation Skip

Perforation skip mode prevents accidental printing on the perforations between forms with fanfold paper. To enable perforation skip, the escape sequence is &l1L. The default perforation skip area is six lines. Perforation skip is disabled with &l0L. The default mode is perforation skip disabled.

The size of the perforation skip area is adjusted by first setting the correct page length, and then adjusting the text area to leave the desired skip area. Defaults for page length and text length in lines are 66 and 60, respectively, giving a six-line skip area.

Perforation skip mode can be used to simulate top and bottom margins. This involves adjusting the top-of-form so the perforation on the forms falls in the middle of the perforation skip area. This formula gives the line number on the paper where you should set the top-of-form:

top-of-form = (*perforation_skip_area_in_lines* / 2) + 1

Feed the paper until it advances to the proper line number. Press the upper (*top-of-form*) button on the printer to set the current paper position as the top-of-form position. Refer to chapter 4 for additional information about printer operation.

Page Length

The page length is specified by the number of lines on a physical page—the number of lines that can fit on the paper. (The number of lines actually printed is determined by the text length.) The number of lines is set with the escape sequence ₤&l*lines*P, where *lines* can be numbers from 1 through 255. The default page length is 66 lines.

Before making changes to the page length, observe the following:

- Changing the page length automatically sets the text length to one inch less than the new page length, but this has an effect only while skip perforations is enabled.
- Changing the page length can also change the top-of-form location.

Text Length

The text length sets the number of printed lines per page.

Text length can be changed with the escape sequence ₤&l*lines*F. The length of the text area in lines can range from 0 up to the current page length (maximum 255 lines) and can include graphics. The default text length is 60 lines.

Note

Perforation skip mode must be enabled to use the text length setting.

The following effects should be observed before changing the text length:

- Changing the *page length* also sets the text length to one inch less than the page length specified.
- Changing the *line spacing* does not affect the logical text length. The text length in lines is adjusted instead.
- Changes to the *text length* also affect the perforation skip area. When the text length is increased or decreased, the perforation skip area is adjusted to keep the page length the same.

Printing of the text area begins at the top-of-form position. After the last line in the text area is printed, the paper is advanced past the skip perforation area to the next top-of-form, where printing continues. The text length setting specifies the last line of text to print before executing a form feed.

Top and bottom margins can be simulated by centering the text area on the logical page. Refer to "Perforation Skip" above.

Overstrike

You may occasionally need to print text on top of previous text. To overstrike a character, the current printer position must be moved back to the desired area.

To overstrike single characters, use the back space character (CTRL H).

To overstrike an entire line, use the carriage return character (CTRL M) between the two lines of text. This causes the second line to be printed over the first line because no line feed occurs.

Wrap-Around Mode

Wrap-around mode prevents loss of characters that would logically be printed beyond the last print position of the current line. The escape sequence to enable wrap-around mode is `&s0C`. Wrap-around mode is disabled with `&s1C`. The default condition at power up is wrap-around mode enabled. Wrap-around mode is disabled at reset (`E`).

With wrap-around mode enabled, the printer moves to the first character position on the next line whenever it encounters another character after reaching the last position on a line. The excess characters are then printed on this extra line. Wrap-around mode is automatically disabled while graphics are printed.

Printing Direction

You can specify whether printing is bidirectional or unidirectional (left-to-right only). The default printing direction for text is bidirectional. The escape sequence to disable bidirectional printing is `&k0W`. Bidirectional printing is restored with `&k1W`. Graphics are printed in unidirectional mode automatically.

Printing Graphics

Graphics mode is used for printing illustrations, charts, and graphs with graphics software. Dot patterns you define with escape sequences can also be printed. The default graphics density is 96 × 96 dots per inch, or 640 dots per line. The horizontal density can be doubled to 192 dots per inch, or 1280 dots per line. The vertical density is fixed at 96 dots per inch.

The printer uses the HP standard "raster scan" method for printing graphics. With this method, the logical page becomes a grid of dot positions. The binary representation of bytes sent to the printer is converted to a series of dots. Each dot represents a binary digit or bit. Bits with a value of "1" are printed, and bits with a value of "0" are left blank. The sequence of the bytes determines the pattern that's printed.

Each byte sent to the printer is interpreted as a row of dots. The number of bytes sent with each escape sequence determines the width of the rows in the pattern. The number of escape sequences sent is the number of rows printed or the length of the pattern. Printing is unidirectional and 12 rows are printed at a time (6 rows for double-density printing).

Four escape sequences can be used to control the printing of graphics. These sequences start graphics mode, set the dot density, transfer the graphics data, and end graphics mode.

Begin Graphics

This sequence prints any text data in the buffer, advances the paper one line, and starts graphics mode. The escape sequence is ESC*rA. The paper isn't moved if the buffer is empty.

Dot Density

This sequence sets the horizontal density to normal (96 dots per inch, 640 dots per line) or double-density (192 dots per inch, 1280 dots per line). The density is set with the escape sequence ESC*r*dots*S, where *dots* values less than or equal to 640 specify normal density, and values greater than 640 specify double density. Normal density is the default.

Graphics Data Transfer

This sequence causes the printing of graphics. The escape sequence ESC*b*bytes*W also indicates the number of bytes of graphics data that follows. These bytes of data must immediately follow the escape sequence, and they're interpreted as one row of dots. A maximum of 80 bytes (160 bytes for double density) can be printed on one line. Additional bytes aren't used.

The printing commences after 12 rows of graphics data have been sent with normal density (after 6 rows for double density). The most significant bit of a graphics data byte prints as the leftmost dot. The least significant bit prints as the rightmost dot. If a space is included in the graphics data, it's counted as one byte (its decimal value is 32).

End Graphics

This sequence prints any graphics data still in the buffer and advances the paper to avoid being overprinted with text. The escape sequence is ESC*rB.

Display Functions Mode

Display functions mode is a debugging aid that prints—but does not execute—all characters received by the printer. Display functions mode is enabled with ESC Y and is disabled with ESC Z. The carriage return and line feed control characters and ESC Z are printed *and* executed as a convenience. The default mode is display functions disabled.

The printed output (see below) in display functions mode consists of printable characters and mnemonics used by the printer. Escape sequences appear the same as when you entered them on the computer screen. Control characters you entered appear as mnemonics on the printer. These mnemonics differ from the special characters that you can access on the screen.

D

```
Control characters are printed as
mnemonics: backspace--BS Z

Escape sequences are printed with the
escape character: [ESC]Z--ESC Z

Alphanumeric characters are printed
exactly as you enter them: abc--abcESC Z
```

Self-Test

Checking the operation and functioning of the printer with the built-in self-test feature is recommended after setting up the system and after each change of the print head cartridge. You can start the self-test at power-on according to the instructions under "Verifying That Your Printer Works" in appendix A. The escape sequence ESC z also starts the test. Standard ink-jet paper should be loaded in the printer with the smooth side facing the print cartridge before starting the self-test.

The self-test normally takes about one minute to complete. The printer runs internal checks first and then prints one page of the character set. Internal error conditions cause the yellow attention light to blink. Failure to print the character set indicates problems with the printer mechanism. The disc-based diagnostic program tests the printer—refer to "Running the Diagnostic Program" in appendix A. Refer to the *Support Guide* for information about obtaining repair service.

Resets

The printer is reset by sending the escape sequence ⁞E. This simulates the original power-on conditions, except that wrap-around is disabled. Specifically, resetting the printer causes any data in the buffer to be printed, returns the printer to its default modes, and executes a form feed unless the paper is already at top-of-form.

The printer is automatically reset if the print head carriage jams or is moved manually while the printer is on. This reset clears the buffer and returns the printer to default modes. The yellow attention light also begins blinking. If this occurs, press the upper (*top-of-form*) button on the printer. If the attention light continues to blink, turn the computer off, then check its operation as described under "Self-Test" above.

Control Sequence Summary

The following tables list all the control characters and escape sequences that you can use with the built-in printer. The control characters and escape sequences are listed alphabetical according to the ASCII character set. To determine the decimal value of a specific control character, refer to the character set in appendix C.

Control Characters:

CTRL H	Back space.
CTRL J	Line feed.
CTRL L	Form feed.
CTRL M	Carriage return.
CTRL N	Bold print mode on.
CTRL O	Bold print mode off.

Escape Sequences:

Esc&d@	Underline off.
Esc&dD	Underline on.
Esc&k0G	CR→CR, LF→LF, FF→FF.*
Esc&k1G	CR→CR/LF, LF→LF, FF→FF.*
Esc&k2G	CR→CR, LF→CR/LF, FF→CR/FF.*
Esc&k3G	CR→CR/LF, LF→CR/LF, FF→CR/FF.*
Esc&k0S	Normal print pitch.
Esc&k1S	Expanded print pitch.
Esc&k2S	Compressed print pitch.
Esc&k3S	Expanded-compressed print pitch.
Esc&k0W	Unidirectional printing.
Esc&k1W	Bidirectional printing (text mode only).

* CR is a carriage return, LF is a line feed, and FF is a form feed. The arrows indicate how the printer interprets these control characters.

Escape Sequences (Continued):

Et&l6D	Line spacing of 6 lines per inch.
Et&l8D	Line spacing of 8 lines per inch.
Et&l *lines* F	Sets text length—0 through page length.
Et&l0L	Disables perforation skip.
Et&l1L	Enables perforation skip.
Et&l *lines* P	Sets page length—0 through 255 lines.
Et&s0C	Wrap-around mode on.
Et&s1C	Wrap-around mode off.
Et*rA	Begin graphics.
Et*r *dots* S	Graphics dot density. If *dots* is less than or equal to 640, then 640 dots per line is selected. If *dots* is greater than 640, then 1280 dots per line is selected.
Et*b *bytes* W	Graphics data transfer for specified number of bytes.
Et*rB	End graphics.
Et=	Half-line feed.
EtE	Resets printer to power on defaults, except wrap-around is disabled.
EtY	Enables display functions.
EtZ	Disables display functions.
Etz	Performs the self-test.

D

E Messages

This appendix lists messages that the computer displays in the PAM window and in the copyright window (at power on), and explains their meanings.

PAM Messages

The messages listed below can appear in the top line of the PAM window. Following each message are an explanation and, for error messages, some hints about solving the problem. Where the listing below shows *file_name* or *folder_name*, for example, the message you'll see includes the name of the actual file or folder.

The first group of messages are listed alphabetically. A second list includes messages that start with the appropriate file name or folder name, and can't be included in the alphabetical listing.

`Cannot copy '`*folder_name*`'.`

You must create a new folder and copy the files in the original folder to the new folder.

`Cannot create '`*file_name*`'.`

1. Make sure that the destination disc is write-enabled (the colored tab blocks the hole in the corner). 2. Check that there is enough room on the disc for a new file. Refer to the `verify_disc` program on the utilities disc.

`Cannot delete 'name'.`

Make sure that the disc is write-enabled (the colored tab blocks the hole in the corner).

`Cannot delete the open folder.`

Press **Close** (f8). The next outer folder becomes the open folder, and the folder you want to delete is listed among the contents of the outer folder. You can then delete the folder.

`Cannot find folder's location.`

Check that you correctly typed the name of the destination folder and, if the folder is on a disc, that the disc is fully inserted.

`Cannot make a unique window name.`

When there is an existing window with the same name as a new program, PAM appends "!" to the name for the new window. This error occurs if PAM can't append "!" without exceeding the 14-character limit.

`Cannot make a window for the command.`

A system limitation was reached, usually by trying to have too many windows at one time.

`Cannot move a folder between discs.`

You must create a new folder on the destination disc and move the files from the original folder to the new folder.

`Cannot move a folder into itself.`

You can't introduce circularity into the file system. For example, you can't move `Folder1` into `Folder2` if `Folder1` contains `Folder2`. First move `Folder2` so it is no longer contained in `Folder1`, and then move `Folder1` into `Folder2`.

`Cannot open '`*folder_name*`'.`

You don't have execute permission for the folder. Use the HP-UX command `chmod` to change the permission.

`Cannot read '`*name*`'.`

You don't have read permission for the file or folder. Use the `chmod` command to change the permission.

`Cannot rename '`*folder_name*`'.`

You can't rename or move a folder specified by `.` or `..` or by a name ending in `/`.

`Cannot run '`*command_name*`'.`

There is a problem with the operating system or a disc. Check whether the program file is usable by using the `verify_disc` utility. If necessary, run the diagnostic tests, which test the operating system (and clear memory).

```
Cannot start another program.  Too many
 are running.
```

You can run up to 18 processes at one time.

`Cannot write in folder 'name'.`

You don't have write permission for the folder. Use the `chmod` command to change the permission.

`Command missing in command line.`

You specified I/O redirection or a command terminator without a corresponding command. Supply the command.

`Done with copy (or move or rename).`

The command is completed.

`Error while reading 'file_name'.`

The disc may be bad. Check it using the `verify_disc` utility.

`Error while writing 'file_name'.`

1. There isn't enough room to store the file. 2. The disc may be bad. Check it using the `verify_disc` utility. 3. A transfer to a peripheral device had an error. Check the peripheral and its connection to the computer.

`Missing name after '>' (or '<' or '^' or '#'or '>>'or '^^').`

The name that should follow the symbol is missing. Supply the name.

`Now complete the copy (or move or rename) with 'to file name'.`

After you give a one-parameter `copy`, `move`, or `rename` command, this message prompts you to supply the second parameter.

```
Please wait ...
```

A copy, move, or rename is in progress.

```
Ready for a disc.
```

PAM is ready to make the disc you insert the open folder. You can also press Reread (f1) to display the root folder.

```
Sorry, no more memory.
```

1. The programs currently running use all of the computer's memory. Stop any unneeded programs. 2. Scripts are nested more than five levels deep. Modify the script file.

```
Specify file to copy, move, or rename
 before specifying 'to'.
```

You gave a to command before specifying the action to perform.

```
The 'name' command does not accept
 parameters.
```

You specified a parameter for a command that doesn't take parameters.

```
The application window is ambiguous in
 the command.
```

A window is specified more than once in the command. Only one window can be specified for all the commands in a single command line.

```
The standard input (or output or error) is
 ambiguous in the command.
```

The standard input, output, or error is specified more than once in the command.

```
Too many characters in script command
 line.  Limit is 160.
```

The number of characters is determined *after* parameter substitution.

```
Too many command parameters.
```

The specified pattern matches so many file names that the resulting command uses up command memory. Enter the command so that it has fewer parameters.

```
Too many commands in the command line.
  Limit is 10.
```

Edit the command line and try again.

E

```
Too many parameters.  Use 'cd folder'.
```

The `cd` command takes only one parameter.

```
Unmatched quote character ' (or ").
```

Supply the matching single- or double-quote character.

```
Use 'to file name' to copy (or move
 or rename) the file.
```

You used the wrong syntax for the `to` command.

```
Use the [Next] and [Prev] keys to see
 more files.
```

The folder contains more files than can be displayed at once in the PAM window.

```
Window name cannot contain a '/'.
```

Windows must have simple names.

Messages Starting With Names. The following messages start with a name of some type. They're listed alphabetically according to the *second* word.

'name' cannot be found.

PAM doesn't understand the command you typed in. If *name* is a PAM command, such as `makefolder` or `print`, check the spelling. If *name* is a simple file name, make sure it's listed in the open folder (or elsewhere in the search path defined by the environment `PATH` variable—refer to the discussion of search paths under "Environment Files" in chapter 9). If *name* isn't in the open folder (or elsewhere in the search path), you must specify *name* as a full path name.

'*name*' could not be matched.

No file name matches the indicated pattern. (You used a special character to indicate a *pattern* rather than a *file name*.)

'*file_name*' exists and cannot be overwritten.

Make sure that the destination disc is write-enabled (the colored tab blocks the hole in the corner).

'*command* file name' is the proper form of the command.

You gave the wrong number of parameters. Check the command's syntax.

'*name*' is a device.

You tried to specify a device as a command.

'*name*' is a pipe.

You tried to specify a pipe as a command.

'*name*' is matched more than once;
one match is required.

The command requires one file name, but the indicated pattern matches more than one file name. (You used a special character to indicate a *pattern* rather than a *file name*.)

'*name*' is not a folder.

You must specify a folder as the parameter for `cd`.

'*name*' is not a window.

The indicated name isn't the name of an existing window.

'*file_name*' is not a runnable script.

The indicated file either isn't a script file or doesn't have execute permission. Use `chmod` to change the permission.

'*folder_name*' is not empty so it cannot be
deleted.

Move or delete all files in the folder. You can then delete the folder.

Power-On Messages

The following messages can appear in the copyright window when the computer is switched on. These messages indicate problems that are detected by the power-on self-test. Refer to the *Support Guide* for information about obtaining repair service.

`Option rom checksum error.`

A plug-in ROM board failed the verification test—the board requires service.

`Ram test error on internal ram.`

Internal memory (RAM) failed the verification test—the computer requires service. (Not all memory locations are tested, so memory *may* be bad without causing this message to be displayed.)

`Ram test error on port xx.`

The plug-in memory (RAM) module in the indicated port failed the verification test—try the module in another port to determine if the module or port requires service. Port A is the top port on the rear panel of the computer, port B is next below, and so on. If you have a bus expander connected, the number 1 indicates the top-left port (as viewed from the rear)—for example, port B1 is the upper-left port in the bus expander connected to the computer's port B. (Not all memory locations in each module are tested, so a module *may* be bad without causing this message to be displayed.)

`Rom checksum test error.`

Internal memory (ROM) failed the verification test—the computer requires service.

F Introduction to the vi Editor

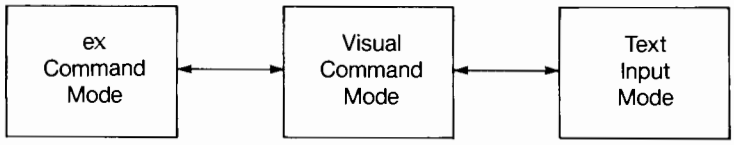
The `vi` (*vi*sual) editor is a screen-oriented, interactive text editor. Like other editors, it holds a copy of the file you are editing in a screen buffer. All editing operations are performed on the text in the buffer. Because it is a screen-oriented editor, the user can freely move and scroll through the entire buffer.

The purpose of this appendix it to introduce new users to the `vi` editor. While sufficient information is provided in this appendix to meet the needs of the novice, some users may need to more advanced information. For advanced users and programmers in particular, we recommend that you refer to the command summary provided at the end of this appendix or to "The Vi Editor" article in *HP-UX Concepts and Tutorials* . If you have not used the `vi` editor before, read through this appendix to become familiar with the basic operations of the `vi` editor.

vi Modes

The `vi` editor has three modes: the visual command mode, the `ex` command mode, and the text input mode.

vi Modes



The Visual Command Mode

When the `vi` editor is first invoked, it is in the visual command mode. From the visual command mode you can enter the `ex` command mode by typing a colon, or you can enter the text input mode by entering an insert, append, or change command. Once you have terminated the text input mode, you return to the visual command mode. Similarly, after the editor has completed the `ex` command, you be left in the visual command mode.

Understanding which mode you are in is important, but it may also be confusig at first. Using the wrong commands when you are in a particular mode can cause unpredictable results. For example, you can not use the cursor keys when you are in text input mode without causing problems. Part of the confusion results because the editor will not prompt you or otherwise let you know what mode you are currently in. To avoid these problems, think of the visual command mode as your "home" mode. Whenever you want to cancel a partially formed command, press the (ESC) key. This should get you back into the visual command mode. To be sure you really are back in this mode, press (ESC) again. If you are in the visual command mode the editor will beep after the second (ESC) to tell you that it expects a command.

The Ex Command Mode

The `ex` command mode is entered from the visual command mode by typing a colon (`:`). In this mode you can enter certain commands that are not available from the visual command mode. In the command summary at the end of this appendix, `ex` commands are shown with the colon prefix. After the command has executed, you will be left in the visual command mode.

You may notice that, unlike the visual command mode, your `ex` command will be echoed at the bottom of the screen. This is because the `ex` command mode is actually an *ex*tended version of the `ed` line editor. Like the `ed` editor, it echoes your command to the console. In our simplified model of the `vi` editor (which is actually just a *vi*sual mode of the `ex` editor), we have omitted mentioning the full features of the `ex` editor because most users on the Integral PC will want to use the full screen editing capability. If you are interested in the `ed` or `ex` editors, we recommend that you read the appropriate articles in *HP-UX Concepts and Tutorials,* Vol. 1: Text Processing and Formatting.

The Text Input Mode

As you type in the text, note that the tilde (~) on each line disappears as you begin typing on that line. Also, if the line is longer than 80 characters, the line of text will wrap around to the next line on the screen. (A "line" is a string of up to 511 characters terminated by a Return. If you type in more than 511 characters without pressing the Return key, the editor will throw away the most recent text change, print the error message `line too long`, and leave you in the `ex` mode. To return to the visual mode, type `vi` Return.)

To terminate the text input mode and return to the visual command mode, press ESC.

Getting Started Using vi

How To Invoke the vi Editor

The `vi` editor is located on the Commands 2 disc. It may be invoked with or without a file name argument. If you start `vi` without a file name, it will create a buffer area in memory without opening a disc file. The text buffer can be edited and then saved on disc during the editing session. The full syntax for invoking the `vi` editor can be found in the `vi.doc` file (in the `/hp-ux2/hp-ux2.doc` folder on the Commands 2 disc).

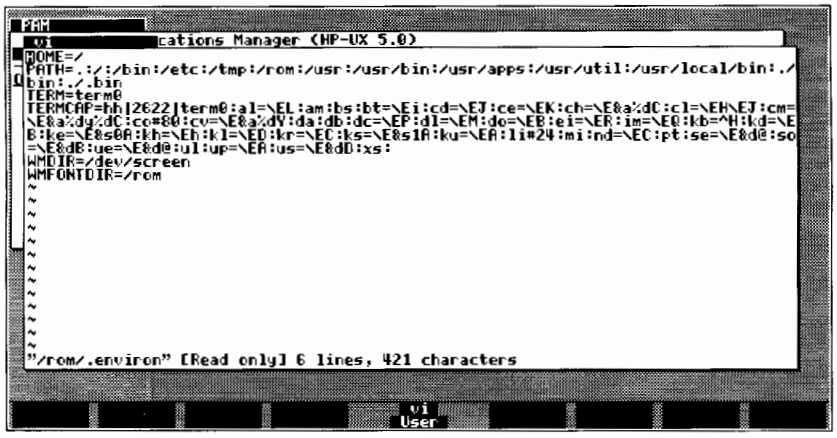
Example. To illustrate how to invoke the `vi` editor, we will edit the default environment file `/rom/.environ` as an example.

To invoke the `vi` editor:

1. Insert the Commands 2 disc.
2. Enter the following in the PAM command line:
   `vi /rom/.environ` Return
3. This will start the `vi` editor and load the `.environ` file into the text buffer area.

F

The following screen should appear with the cursor positioned at the beginning of the first line.



Note that each empty line below the cursor is marked with a *tilde*. The *tilde* "~" is a special character used to mark the end of the text in a file that already exists, or, in the case of a new file, to show that the file is empty. The tildes will not become part of the text in your file.

A Sample Editing Session

For a quick example, let's add a time zone variable to your environment file. (The environment file is described in chapter 11 under `getenv`.) When the `vi` editor is first invoked, it is in the visual command mode. It is waiting for a command. We will only introduce a few commands in this section to give you a feeling for the editor. For more information, refer to the command summary at the end of this appendix.

Undoing Changes. If something goes wrong and you get hopelessly lost, do the following:

1. Press the (ESC) key.
2. Press (Shift)(U) to undo all changes in the current line, or type `:e!` (Return) to throw away all your changes and edit the file again.
3. You can continue editing (you are left in the visual command mode) by entering a command, or quit and throw away your changes, by entering `:q!` (Return).

To add a time zone variable, do the following:

1. Move the cursor to the end of the last line by entering the command `L$` (the `L` command moves to the beginning of the last line, the `$` command moves the cursor to the end of the line). Now you are ready to append text.
2. Enter the text input mode, type `a` (Return) (append).
3. Enter your time zone variable, for example:
   `TZ=PST8PDT` (Return)
4. Press (ESC) to return to the visual command mode.
5. Type in `:w /.environ` (Return).

   This will save your new environment file in the root folder.

To use your new environment file, enter the following command in the PAM command line:

`getenv /.environ` (Return).

Saving Your Text File

In the last example, we used the `:w` command to save the text buffer in file. If you want to save the text in a file and stop the `vi` editor, first insert the destination disc. Next type `ZZ`. You do not need to press the (Return) key with the `ZZ` command. The `vi` editor will respond by saving your file, displaying a line at the bottom of the screen containing the file name and size of file, then quit. You have now terminated the `vi` program and saved your file. The `vi` window is now "dead" and will disappear when the next PAM command is executed. To return to PAM, press (Shift)(Select) or (Stop).

Using the Cursor Commands

Moving around in your text buffer is one of the most important editor functions. As you can see from looking through the command summary, the `vi` editor has large and powerful set of cursor movement commands. This section describes only a few commands which will enable a beginner to move around.

The following cursor movement commands allow you to move around the text buffer while in the *visual command mode*. They will *not* work in the text input mode!

Basic Cursor Movement Commands

Command	Action
h, ◄	Moves the cursor one space to the left. The Backspace key also moves the cursor one space to the left.
j, ▼	Moves the cursor down one line in the same column.
k, ▲	Moves the cursor up one line in the same column.
l, ►	Moves the cursor one space to the right. The space bar also moves the cursor one space to the right.
+	Move to start of next line.
–	Move to start of previous line.
$	Move to end of line.
↖	Move cursor to home position on screen.
0	Move to start of line.

Notice that the h, j, k, and l keys form a row of keys directly under the right hand. Some users will find these keys easier or faster to use than the arrow keys.

All of the cursor movement commands can be prefixed by a number specifying how many times the command is to be repeated. For example, 3j will move the cursor down three lines.

The cursor movement command may also be used with other commands to specify a text segment. For example, d0 will delete text from the current cursor position to the start of the current line.

Adding, Deleting and Correcting Text

The visual command mode provides several simple commands to add, delete or change text in your file. This table below lists a few of the most frequently used commands.

Commands for Adding, Deleting and Correcting Text

Command	Action
i	Insert
a	Append
o	Insert line below current line.
O	Insert line above current line.
x	Delete character
dd	Delete line
r	Replace character.
u	Undo last change to text buffer.
U	Undo last change to current line.
Y	Yank current line into buffer.
P	Put yanked text back into text.

The `vi` editor uses both uppercase and lowercase keys as commands. For this reason, you must be careful to enter the correct command. If you make a mistake, use the undo command to correct the problem.

Like the cursor movement keys, some of the commands listed above can be combined with prefix specifying the number of times the command is to be repeated, or with a cursor movement command. For example, `5x` will delete five characters. Refer to the command summary for more information of the syntax of these command.

Appending and Inserting Text

The i and a commands are used for inserting and appending text, respectively. The i command places text to the left of the cursor, and the a command places text to the right of the cursor. Both commands leave you in the text input mode.

Entering Text. Once you are in the text input mode, every regular character typed at the keyboard will be entered in the text buffer. Each line of text is terminated by a (Return) and may be up to 511 characters long.

Correcting Errors. If you make a mistake while in the text input mode, use the (Backspace) key to backspace over your mistake. The letters will remain on the screen until you enter new text or return to the visual command mode. If you want to delete all the text in the latest change, return to the visual command mode and use the undo command.

Entering Control Characters. You can enter non-printing control characters or escape characters in the text input mode by using the (CTRL)(V) command. The editor will respond by displaying a caret (^) in your file. At this point it is ready for you to enter a control character or escape character. The *next* character will be passed along to your text buffer without being processed by the editor. For an example, refer to "Entering Non-Printing Characters" below.

F

Inserting New Lines

New lines may be inserted while in the text input mode simply by pressing (Return). From the visual command mode, the o command can be used to create a new line *after* the line containing the cursor. This command will create a new line, reposition the cursor to the beginning of the new line, and leave you in the text input mode. This may save you some time and allows you to enter text into a blank line. The O command is identical to the o command, except it creates a new line *before* the line containing the cursor. Both commands leave you in the text input mode. Pressing (Return) after an o or O command creates an additional new line.

Like the a and i commands, the o and O commands are invoked only from the visual command mode (remember to press the (ESC) key to get back to the visual command mode).

Deleting Text

Deleting Characters. The x command deletes the character marked by the cursor. You can delete more than one character at a time by preceding x with a number. For example, 5x deletes five characters beginning at the cursor.

Deleting Lines of Text. To delete a line, place the cursor on the line you want to delete and type dd once. Like the x command, you can repeat the command by prefixing it with a number: 5dd will delete five lines, starting at the current line.

Replacing Text

The r command is used to replace the character immediately under the cursor. For example, to replace the character w with a y, simply position the cursor over the w, then type ry. This command will leave you in the visual command mode.

Like the previous commands, if you prefix the r command with a number, it will repeat. But look what happens! The command 10rx will replace the next 10 characters with x's! This usually is *not* what is expected, but it may be a useful feature in some cases.

If you want to change the 10 characters following the cursor, try using the change command: 10c. This command will place a dollar sign after the 10th character to mark the text that is to be changed, then it will place you in the text input mode. The next 10 characters entered will replace the text in your screen buffer; afterwards, you will be in the text input mode and can continue inserting text.

Moving Lines of Text

The Yank Command. The yank command Y allows you to place a copy of text from your screen buffer into one of several temporary buffers so it can be moved to another location. The Y command can be prefixed by a number to specify the number of lines to yank. The upper case Y form of the command will place your text into an unnamed buffer. The lower case y command can be prefixed with a buffer name (use a double quote mark followed by one of the lower case letters a through z to specify a buffer). For example, "a3y will yank three lines into the "a" text buffer. The original lines in you screen buffer remain unaffected.

The Put Command. The put command takes the text out of one of the buffers and puts it into your screen buffer. The command has two forms, `p` (lower case) and `P` (uppercase). The lower case form puts the text after or below the cursor; the upper case form puts it before or above the cursor. Both forms can take a buffer name prefix. The form of the buffer prefix is the same as for the yank command.

Unlike the yank command, the put command can also be used to restore text that has been deleted. Text from the last 10 delete operations is stored in numbered buffers `0` through `9`. These buffer names may be used in place of the letter buffer names.

Writing Text to a Disc File

The Write Command. Writing the screen buffer to a file is accomplished using a `:w` command in the `ex` command mode. The contents of the screen buffer remain unchanged.

If you specified a file name when the `vi` editor was invoked, the `:w` command will use the name of the file you are editing or creating, and writes to that file by default. If you have not specified a file name up to this point, you must supply a name with the `:w` command, or by using the `:file` command.

Editing Another File. After executing the `:w` command you may continue editing your file or use the `:e` command to edit another file. This can also be used at the beginning of your editing session to specify a file.

Exiting vi

If you want to exit the `vi` editor and save your file, use the `ZZ` command. This will save your screen buffer in the current file name.

To quit without saving your file, use the `:q!` command.

After exiting vi using either the ZZ or :q! command, the vi window is "dead," which means the vi program has been terminated and the window will disappear when the next PAM command is executed. To get back into PAM, press (Shift)(Select) to shuffle the windows.

Entering Non-Printing Characters

Non-printing characters (ASCII codes 0–31) may be entered in text mode by typing (CTRL)v (this will be displayed as a ^) followed by the control character used to produce the non-printing character. For example, to enter ESCz in your text, use the following sequence of keystrokes:

(CTRL)v then (ESC) then z

This will be displayed on the screen as: ^[z. In this case, the (ESC) key is formed by the combination of ^[. To correlate specific keys on the keyboard with control sequences, refer to appendix C.

Note that the extended characters (ASCII codes 128–255) cannot be entered using the vi editor. If you need to create a text file using these characters, use a general-purpose editor such as MemoMaker (available separately).

F

Where To Find More Information on vi

For more in-depth information on the vi or ex editors, refer to *HP-UX Concepts and Tutorials*, Vol. 1, Text Processing and Formatting.

To quickly find a command, refer to the vi quick reference guide at the end of this appendix.

For information about the syntax used in invoking vi, refer to the vi.doc file (in the /hp-ux2/hp-ux2.doc folder on the Commands 2 disc).

vi Command Summary

The basic editor commands are listed in the following summary. The syntax consists of three elements: a prefix number *n*, one or more characters that must be entered literally, and one or more parameters. In the following summary the prefix *n* represents the number of times to repeat the command. This prefix is optional and may be omitted in all cases where it appears in the command summary. Characters shown in `dot matrix` must be entered exactly as shown, while characters shown in *italics* are parameters and must have the appropriate substitution made by the user. Not all commands will have all these elements.

Cursor Positioning Keys

*n*k, ↑	Move the cursor up *n* lines in the same column.
*n*l, →, Space	Move the cursor *n* characters to the right.
*n*j, ↓	Move the cursor *n* lines down.
*n*h, ←, Backspace	Move the cursor *n* characters to the left.
↖	Move the cursor to the home position.

Scrolling and Paging

CTRL d	Scroll down half page.
CTRL u	Scroll up half page.
CTRL e	Scroll up one line.
CTRL y	Scroll down one line.
CTRL f	Move forward one page in the file.
CTRL b	Move backward one page in the file.

Moving From Line to Line

n+, (Return)	Move cursor to the first printable character on the *n*th line ahead.
n-	Move cursor to the first printable character on the *n*th previous line.
*n*G	Go to first printable character on line number *n*; default *n*=last line of the file.
H	Home cursor to the first printable character on the screen.
M	Move cursor to the first printable character of the middle line on the screen.
L	Move cursor to the first printable character of the last line on the screen.
/*pattern*	Initiates a forward search for *pattern*.
?*pattern*	Initiates a backward search for *pattern*.
n	Repeats the most previous pattern search.
N	Repeats the most previous pattern search in the opposite direction.
:s/*pattern*/*replacement*/	Substitute command. Search on current line for first instance of *pattern* and replace with *replacement*. Patterns can be regular expressions. (Refer to `ed.doc` file in the `/hp-ux1/hp-ux1.doc` folder on the Commands 1 disc for more information on regular expressions.)
:s/*pattern*/*replacement*/g	Substitute command with global suffix option (to match all instances on a single line). Replaces all instances of *pattern* on current line with *replacement*.
:s/*pattern*/*replacement*/c	Substitute command with confirm option. Type `y` to accept substitution marked with quotes.
:g/*pattern*/s//*replacement* /	Global search and substitute command (the `g` command seaches for the first instance on each line in the buffer that matches the pattern). Replaces first *pattern* match on each line in buffer.
:g/*pattern*/s//*replacement* /c	Global substitute command with confirm option.

Moving Within A Line

*n*w	Move cursor forward to the beginning of *n*th word or punctuation mark.
*n*b	Move backward *n* words.
*n*f*c*	Move cursor forward to the *n*th instance of the specified character *c* in the current line.
*n*F*c*	Move cursor backward to the *n*th instance of the specified character *c* in the current line.
;	Repeats the most previous f, F, t, or T command.
,	Same as ; command but in the opposite direction.
0 (Zero)	Move cursor to the first printable character on the next line.
$	Move cursor to the end of the current line.

Inserting and Appending Text

i	Insert text before cursor.
a	Append text after cursor.
A	Append text at the end of a line.
o	Create new line below line containing cursor.
O	Create new line above line containing cursor.

Character Corrections

*n*x	Delete next *n* characters starting at cursor.
r*c*	Replace character under cursor with character *c*.
*n*s	Replace *n* characters with one or more characters.
~	Toggles the upper/lower case setting of character under cursor.

Line Corrections

*n*dd	Delete *n* lines.
d*n*w	Delete the next *n* words. (The d command can also be combined with any other cursor movement command.)
c*n*w	Change next *n* words. (The c command can also be combined with other cursor movement keys, for example c$).
. (Period)	Repeat last operation which changed text in buffer.

Copying and Moving Text

Y	Yank line of text into unnamed buffer.
"*x*nY	Yank *n* complete lines into buffer *x* (where *x* is a letter from a through z).
y*n*w	Yank *n* words into unnamed buffer.
p	Puts saved text from unnamed buffer in screen buffer after cursor. The unnamed buffer hold the last text deleted or yanked.
P	Puts saved text from unnamed buffer in screen buffer before cursor. The unnamed buffer hold the last text deleted or yanked.
"*x*p	Puts saved text back from buffer *x* in screen buffer (after or below the cursor).
"*x*P	Puts saved text from buffer *x* in screen buffer (before or above the cursor).

Join Lines

*n*J	Joins *n* lines.

Undoing a Command

u	Undo last change to text buffer.
U	Undo all changes in current line.

Setting vi Options

`:set`	Display those options whose values have been changed.
`:set all`	Display settings of all options.
`:set` *option* `?`	Display current value of *option*.
`:set` *option=value*	Set numeric *option* to *value*.
`:set` *option*	Turns on *option*.
`:set no`*option*	Turns off *option*.

Insert Control Characters

CTRL v*special_character*	Used in insert mode to enter a *special_character*. (Appears in text as control character symbol ^. Refer to "Entering Non-Printing Characters" for an example.)

Reading and Writing Files

`:r` *file*	Read contents of *file* into current cursor position.
`:w` *file*	Write buffer to new *file*.
`:w!` *file*	Write buffer to existing *file*.
`:w >>` *file*	Append buffer to *file*.
`ZZ`	Write buffer to current file name and exits editor.

F

Quiting

`:q`	Quit (if contents of buffer have been saved).
`:q!`	Quit, discard buffer.

Open Files For Editing

`:e` *file*	Edit *file*.
`:e`	Edit current *file* over again.
`:e!` *file*	Edit new *file*, discard contents of screen buffer.
`:file`	Display information about current *file* name.
`:file` *file*	Change name of current file to use in next disc operation to *file*.

G Glossary

A

absolute path name: A list of folder names that specifies the location of a folder or file in relation to the root folder. The list starts with the root folder (/) and ends with the file or folder being identified. One slash separates each subsequent folder in the list. An absolute path name that identifies a file is also referred to as an absolute file name.

active menu: The menu currently displayed, which indicates the commands currently assigned to the function keys. The name of the active menu appears in the status block (in the middle of the menu).

active window: The window that currently receives your input from the keyboard. The name of the active window is displayed above the menu label in the center of the menu. This window is normally displayed in front of the other windows. *See also* **select a window**.

alpha window: A window used for displaying character data that is created by the PAM program automatically when programs, utilities, and other processes are started. A menu is available for using some of these features. The use of control characters and escape sequences is necessary for using the additional terminal features.

anchor: A movable bracket that is displayed over the upper left corner of a window. The bracket specifies the new location of the upper left corner of a window when it is moved. This corner is fixed and remains stationary when a window is stretched.

arrow keys: The ▲, ▼, ◄, and ► keys. These keys are used for moving the cursor, moving the graphics pen, rolling the contents of a window (with Shift), and moving the display pointer (with CTRL). Pressing two arrow keys such as ▲► specifies diagonal movement; in this case, up and to the right.

autostart file: A file named `Autost` that is automatically searched for at the beginning of each session. At power on, the first file called `Autost` found in the outermost folder of a disc is accessed. The specific action depends on whether the file is a program, folder, or data file.

B

backup disc: A copy of a master disc or a working disc that you create and then store in a safe place. If the master or working disc is lost or damaged, you can make a new copy from the backup disc.

border: The shaded window outline that enables the display pointer to select common operations. The four corners of the border select the "move", "stretch", "hide", and "stop" operations. Four directional areas select scrolling operations.

buffer: Memory that functions as storage area for data that is not immediately operated on by the system. Buffers are used by devices and the computer whenever input or output operations are performed. When a data file is edited, for example, the edited version can be kept in a terminal buffer until the file is "saved" by storing it on a disc.

built-in device: A device contained in the computer that can be accessed without the use of an interface. Disc drive A is an example of a built-in device.

byte: The fundamental unit of information used by the computer. In data operations, each character is represented with one byte of memory. Each byte contains eight binary digits or bits. A byte equates to a decimal number in the range 0 through 255.

C

character font: The specific bit image that is generated in a window when keys are pressed or output is directed to a window. The character set and display enhancements assigned to the character font determine what characters are generated.

character set: A specific sequence of characters and control characters used by the computer for handling information. Each character has a numeric value that establishes its order in the character set. The computer uses the U.S. ASCII character set with Roman extension (ASCII/Roman) by default.

command: A sequence of characters that, when entered in the command line, cause an operation to occur. Commands generally start with a name that identifies an executable file, such as a program or utility, and may include parameters.

command line: The second line of the PAM window, where the cursor is located. Any characters you type in the PAM window go to the command line. When Return is pressed, the computer attempts to execute this text as a command.

compiler: A program that reads program statements from a data file and creates a program file containing instructions that cause the computer to perform the operations specified by the program statements. A different compiler is needed for each programming language.

computing environment: A group of settings the system uses for determining the result of specific operations. The environment is set by variables in an `.environ` file at power on and includes settings for the home folder and the search path, for example.

concurrent execution: The simultaneous execution of two or more programs, utilities, or commands in the computer. Commands entered in the command line are executed concurrently unless sequential processing is specified.

control character: A character that can cause a specific action to occur instead of displaying or printing a character. In the ASCII/Roman character set, the decimal equivalents of 0 through 31, 127, and 255 are reserved for control characters. Control characters can be entered by pressing CTRL and another key. Pressing CTRL J generates the line feed control character, which moves the cursor to the next line, for example. In display functions mode, control characters are displayed as mnemonics. Control characters are also referred to as control codes. *See also* **control sequence** and **escape sequence**.

control sequence: A control character or escape sequence. *See* **control character** and **escape sequence**.

current line: The line of text that contains the cursor.

cursor: A position indicator that shows where the next character entered will be placed. In the PAM window, the cursor only appears in the command line. Unless the application limits its cursor movement, the cursor can be moved to any character position with the arrow keys.

data file: A file that contains text or other data in a character format. Each byte in the file represents a specific character in a character set. Data files can be displayed with the `view` command.

data path: A stream of characters that is being transferred from one point to another point in the file system. A data path can result from input or output operations. Each program has three data paths associated with it: standard input, standard output, and error output. Each of these data paths can be directed to specific files, including device files, in the file system.

decimal equivalent: The decimal value assigned to a character in a character set. In the ASCII/Roman character set, the decimal equivalent of the line feed control character is 10. This value is used by the computer to represent each character stored in the computer and can be used to represent characters in a program.

default action: The action that occurs when a file is accessed. The file is usually accessed by entering its name as a command in the command line and pressing Return. The system uses the file type to determine what action to take. Folders are opened, data files are displayed, and programs are executed. The `ACTION` variable can change the action for a data file.

destination: In a data transfer operation, the file, folder, or disc to which data is transferred. In a copy operation, the file being copied to is referred to as the destination. When a compiler is used, the data file containing the compiled statements is called the destination file. *See also* **source**.

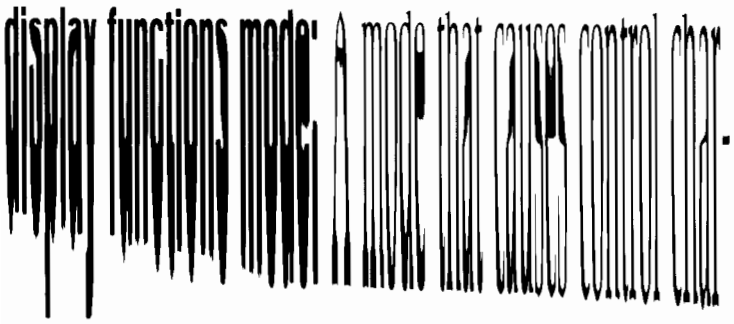
device address: A peripheral device setting that allows an interface in the computer to identify devices uniquely. The HP-IB interface can access each device it is connected to with a different address. The device address switch on a peripheral changes its address.

device file: A file that represents a specific device that the computer can access when performing input or output operations. The files `lp` and `plotter` are examples of device files that can be accessed for user output operations. Each device file contains information about what type of device it accesses and information for uniquely identifying the device it is currently assigned to. Device files are stored in the `dev` folder.

device name: The name assigned to each disc drive, plotter, printer, or other device by the operating system at power on, used as part of a path name. For example, a device name can be used to identify a disc drive as the source or target of a disc copy operation. The device names are listed in the `dev` folder.

directory: Another name for a folder. *See* **folder**.

display bar: An indicator (highlight) used for selecting specific files or folders in the PAM window. The display bar can be moved over different file and folder names, including the name of the open folder, with (Tab) and (Shift)(Tab) or the arrow keys. Selecting a file or folder with the display bar and pressing (Select) is an alternative to entering its name in the command line.



acters and escape sequences to be displayed instead of executed. This is useful for debugging purposes when output from a program, for example, is not what you expected.

display pointer: An image that can be moved anywhere on the screen to select a location. Depending on what is currently displayed in the location selected, different actions are taken. If the selected location is in a menu label, then the action of pressing the associated function key is performed when (Select) is pressed or when the left mouse button is pressed. The display pointer can be moved with a mouse or the arrow keys with (CTRL). The default pointer shape is an arrow, but different forms can be assigned programmatically. The display pointer is also called a sprite.

"drive busy" light: The colored light near the corner of the disc drive. When the computer is reading from or writing to a disc, this light comes on to warn you not to eject the disc.

driver: A program that controls the input or output operations of a specific type of interface or device. When a special device or interface is to be accessed, a loadable driver can be installed in the computer. Drivers are built in and automatically loaded for disc drives, printers, and plotters connected to the built-in HP-IB interface, and for printers and plotters connected to a plug-in serial interface.

E

editing keys: Keys on the keyboard that can be used for updating information on the screen by deleting, clearing, and inserting characters. Keys are provided for editing on a character, line, or window level. Keys used for moving the cursor, rolling the window, and moving the display pointer are also considered editing keys.

electronic disc: A logical device that is used for storing files in RAM. The electronic-disc folder is the root folder of the file system (/). Files on the electronic disc can be accessed very quickly.

environment file: A file containing variable assignments that set the computing environment. These include the HOME variable for naming the home folder and the PATH variable for setting the search path. A file called .environ is accessed automatically after power on to set the computing environment. Other environment files can be used during a session to modify this environment.

escape sequence: A string of two or more characters that begins with the escape control character (ESC), and performs a specific function when received by a device, such as a printer or terminal. *See also* **control character** and **control sequence**.

extended character: A character accessed by pressing (Extend char) and another character key. Extended characters include foreign characters and other special characters in the second half of the ASCII/Roman character set.

F

file name: A sequence of 1 to 14 characters used to identify a particular file. Any keyboard character can be used except slash and leading colon (for convenience, it's best to avoid all special characters). File names are used in combination with folder names to form absolute path names and relative path names. The file name is entered into the directory in which the file is located. *See also* **absolute path name**, **file path name**, **relative path name**, and **simple file name**.

file path name: A file name preceded by a list of the successive folders that must be opened to access the file. For example, the file path name for a file named `Myfile` in a folder named `Myfold` on a disc named `Mydisc` would be `/Mydisc/Myfold/Myfile`. *See also* **absolute path name**, **file name**, **relative path name**, and **path name**.

file permissions: Indicators that determine what types of operations a file can be used for. There are three file permissions: *read* permission, *write* permission, and *execute* permission. Each permission is set either on or off for a file. File permissions can be changed by running the `chmod` utility. Only files with execute permission can be executed as a command, for example. File permissions are also referred to as file modes.

file size: The amount of data that is stored in a file, represented in bytes. The size of a file determines how much space it uses when stored on a disc.

file system: The hierarchical structure of folders and subfolders that is used for storing and organizing information in the computer. The location of any file or folder in the file system can be expressed in relation to the root folder with an absolute path name.

file type: The file characteristic that is set when a file is created and determines how it is used. There are four file types: *data* files, *folder* files, *device* files, and *program* files. Data files can be changed to script files by modifying the file permissions.

fixed disc drive: A disc drive in which the medium is more or less permanently installed. It typically operates faster than a microflexible disc drive.

folder: A file that contains pointers to other files and folders. These files and folders are what is logically stored in a folder file. Folder is another name for directory. *See also* **directory**.

formatting: The creation of a file system on a disc. This includes assigning the disc a volume name, calculating the number of files that can be stored on it, and creating a folder for storing that many files. Before a disc can be formatted, it must be initialized.

function keys: Keys (f1) through (f8) at the top of the keyboard. *See also* **function-key labels**, **menu**.

function-key labels: Individual elements in the menu at the bottom of the display. Each function-key label displays the command currently assigned to the corresponding function key. The labels can be turned on or off with (Menu). *See also* **function keys**, **menu**.

G

graphics window: A window used for plotting graphics data that is created when output is sent to the device file `plotter`. The plotting emulates that of a peripheral plotter.

H

hidden file: A file or folder with a name that begins with a period. The period prevents the file name from being listed in the PAM window. The `ls` command lists hidden files.

hidden window: A window that has been removed from the screen with the Hide function key ((f7) in the System menu) or the "hide" element in the optional border. The banner of each hidden window is displayed in the lower left corner of the screen. Information in hidden windows is kept in off-screen storage until the window is displayed again by selecting its banner with the display pointer.

highlight: In the PAM window, to indicate a displayed file or folder name by moving the inverse display bar to that name.

home folder: A folder identified by the computing environment. If this folder is included in the file system, it is displayed in the PAM window at power on and can be accessed during a session by executing the `cd` command without parameters.

HP-IB: The abbreviation for Hewlett-Packard Interface Bus. This interface standard allows multiple devices to be accessed with one interface. The built-in HP-IB interface on the rear of the computer can be used for accessing disc drives, printers, plotters, and other peripheral devices.

HP-UX: The Hewlett-Packard version of the UNIX operating system.

I

initialization: The preparation of a disc for storing information. The initialization process involves marking the disc medium with track and sector marks. After a disc is initialized, it can be formatted.

M

menu: A set of choices that are selected with the function keys, f1 through f8 (shifted and unshifted). Each choice is identified by a function-key label. The system provides two main menus: the System menu and the User menu. The System menu remains the same and can be accessed from any window. The User menu depends upon the type of window that's active and upon the program that's running. The PAM User menu, for example, is different from the User menu displayed by a text editor. *See also* **function keys**, **function-key labels**.

microflexible disc drive: A disc drive that operates with removable, interchangeable microflexible discs. It typically operates more slowly than a fixed disc drive. The built-in disc drive is a microflexible disc drive.

mounted disc: A disc that is accessible through the file system. The folder with a name corresponding to the volume name of each mounted disc is stored in the root folder of the file system. When a file in a mounted disc is read or updated, the operating system automatically accesses the disc in the drive associated with the mounted disc.

mouse: A hand-operated device used to move the display pointer and to select items located by the display pointer. By rolling the mouse on a table top, you can move the display pointer in any direction, select menu choices and windows, and turn the menu labels on or off. Many applications that are available for the computer incorporate the use of a mouse.

N

null string: A representation of the absence of any characters. In a program, character strings are usually enclosed in quotes. The null string is represented by two adjacent double quotes ("").

O

octal number: A number expressed in the base 8 number system. This system is often used for representing values stored in binary digits since groups of three binary digits can be conveniently converted to one octal digit. Octal numbers represent values with the digits 0 through 7.

open folder: The folder whose contents are currently listed in the PAM window. At any time, there is only one open folder. The open folder is also called the current folder.

operating system: The collection of programs used by the computer for managing its devices, processing capability, and memory.

overwriting: The process of replacing original data in a file with new data from an output operation. The data that is replaced is lost. Overwriting can occur accidentally when you intend to specify a new file for output but instead name a file that already exists.

P

PAM: The abbreviation for the Personal Applications Manager program. This program provides many of the features found in traditional UNIX shells and has a friendly user interface consisting of the PAM window and its user menu.

PAM command line: *See* **command line**.

PAM window: The window displayed by the PAM program. This window displays the open folder and has a User menu for performing routine operations.

parameter: A data item that is placed after a command to specify how an operation is to be performed or to supply other needed information. Programs, script files, and other processes also use parameters. Parameters that are identified by their order of appearance in the command line are called positional parameters.

parent folder: The folder that a specified folder is stored in. A parent folder is usually identified in relation to the open folder. The parent folder of the open folder is the folder that is displayed when the open folder is closed. The name of the parent folder can be represented with two periods (. .) as a parameter.

path name: An extended name that includes the names of outer folders. The folder names are separated by slashes, where each folder is stored in the folder that precedes it, and the last item may be either a folder or a file. Path names are used to specify folders or files that can be accessed through the file system. A leading slash represents the root folder. Example: `/MEMOS/January/Re:Sales`.

peripheral device: An external device that is connected to the computer with an interface. External disc drives, daisy-wheel printers, and mechanical plotters are examples of peripheral devices.

permission: *See* **file permissions**.

pipe: A data path that connects two programs that are executed concurrently. The pipe directs the standard output of one program to the standard input of the other program. A pipe between two programs is specified with a vertical bar (|) when the program names are entered in the command line. The use of pipes to connect programs is referred to as piping.

plotter device: A logical device that is associated with each graphics window. Each plotter device can access the plotter function keys and, when the graphics window is active, pen movement instructions directly from the keyboard.

print pitch: The number of characters printed per inch (for a non-proportional character set). The print pitch is used to represent a specific character size on a printer. By sending an escape sequence to certain printers, you can change the size of characters printed. The built-in printer, if present, has four print pitches.

printer waiting list: A list of one or more files that have been designated for printing while the printer is busy. These files are remembered by the system and are automatically printed when the printer stops. Files can be placed in the printer waiting list with the `print` command.

program file: An executable file. Program files are normally created with a compiler. Script files are special types of program files that are created by adding execute permission to a data file that contains commands. The utility files provided with your system are also examples of program files.

R

RAM: The abbreviation for random-access memory. RAM is the read/write memory resource of the computer that is used for running programs and storing information for current operations. Information that is accessed in each window is stored in RAM, for example. The amount of RAM available to the computer can be increased by installing plug-in memory modules. *See also* **electronic disc**.

README: A text file, stored on a disc, that describes the contents of the disc.

redirection: The specifying of alternate files or devices for a program or other process to use for its input and output data paths. Certain symbols are used to identify files for standard input, standard output, and error output. A separate window can also be named for a program to use.

relative path name: A list of folder names separated by slashes that indicates the location of a specific file or folder in relation to the open folder. A relative path name that identifies a file is also referred to as a relative file name. *See also* **absolute path name**, **file name**, **file path name**, and **path name**.

reset: An operation that causes the computer to return to its power on state without erasing the electronic disc. A reset stops all programs and removes all windows.

ROM: The abbreviation for read-only memory. This memory is not lost when the computer is turned off and is used for storing the operating system, the PAM program, and other system information.

root folder: The top-most folder in the file system. This folder is identified by a slash (/). All files and folders contained in the system are accessed through the root folder. The root folder is also used for accessing the electronic disc.

S

script file: A program file that resembles a data file but contains commands and has execute permission. When the name of a script file is entered as a command, the commands in it are executed as if they were entered individually from the command line. Script files are useful for performing frequently needed operations and for executing commands from programs. Script files are classified as a special type of program file.

select a window: To choose a window to be the active window. *See also* **active window**, **shuffle windows**.

search path: A list of folders that are searched in order when the name of a file is entered in the command line. The search path is specified by the environment file variable PATH. Certain application programs also use this variable for determining what folders to search to locate files. The search path is not used to locate files entered as parameters.

sequential processing: The serial execution of programs and other processes in the computer. Sequential processing can be specified for individual commands by placing a semicolon after them in the command line. This causes the commands to be executed one at a time. A command following a command with a semicolon is not started until the command preceding it is completed.

shell: A program that is run continuously to provide a user interface to a computer. This computer uses the PAM program as a shell program. This interface allows programs to be submitted to the operating system for execution, displaying of information in folders, and provides many other capabilities with shell commands. The shell commands for the PAM program are the built-in commands that can be entered in the command line.

shuffle windows: To move the front window to the back, bringing the next window to the front and making it the active window. Performed by pressing (Shift)(Select). *See also* **active window**, **select a window**.

simple file name: A file name that identifies a file by only its name and does not include a path name (folder names). The names of files displayed in the PAM window are an example of simple file names. Simple file names are used for specifying files in the open folder. *See also* **file name**.

source: In a data transfer operation, the file, folder, or disc from which data is transferred. In a copy operation, the file being copied from is referred to as the source. When a compiler is used, the data file with the statements being compiled is called the source file. *See also* **destination**.

status block: The central area at the bottom of the display in the middle of the menu. This area displays the names of the active window, the active menu, the caps indicator (C), and the insert indicator (I).

string: A sequence of characters and other symbols that can be used as one unit by a program, command, or other function. A string that contains only characters is referred to as a character string.

subfolder: A folder (directory) that is contained in another folder. All folders stored in the current folder are called subfolders, for example. All folders in a file system are subfolders relative to its root folder.

System menu: The menu containing the most common commands for window control.

T

tab stop: A column to which the cursor can be moved with the Tab key.

target: The destination file or folder for an operation that transfers data from one location to another. A parameter is usually used for identifying the target. In a copy operation, for example, the target folder is the folder the source file is copied to.

text: Alphabetic character data used to represent written material in a data file or window. Text is usually intended for a user to read and not as a specific instruction for the computer.

text editor: An application program that allows you to create data files, enter text, and perform various editing functions such as searching and replacing characters, moving blocks of text, and justifying margins. Some text editors allow you to designate different print modes such as underline or bold print for use with a printer.

text file: A file that contains text or other data in a character format.

top of form: The position of the first printed line on each page.

U

unit number: A number used to identify a specific disc drive in a disc drive device. A dual disc drive normally uses 0 for the first drive unit and 1 for the second drive unit. The unit number is usually beside the slot where discs are inserted. Single disc drives have a unit number of 0.

User menu: The menu containing commands to control a program.

utility: A program designed for use with HP-UX systems that performs a specific function. Utility programs are used for initializing discs, listing folders, and performing many other necessary operations.

V

volume name: A one- to six-character name used by single-user systems with removable file systems to identify a particular disc. A volume name is assigned to the disc when it is formatted. When the disc is mounted by the operating system, the volume name is entered into the root folder as the name of the top-level folder on the disc.

volume number: A number used to identify each file system or volume stored on a disc. The only disc devices that are normally used for storing multiple volumes are fixed discs. The volume number is incorporated into the device file names for peripheral disc drives so that different volumes on the same fixed disc are uniquely identified. The volume number of a file system stored on a flexible disc is 0.

waiting list: *See* **printer waiting list**.

window: The part of the display used for a particular program or an alpha or graphics device. Your input to a program and its displayed output are contained in the program's window. When more than one of these programs or devices exists, their windows are stacked on the screen like pieces of paper on a desk. *See also* **active window**.

window banner: The tab on the upper left corner of each window that contains the name of the window.

window border: *See* **border**.

window manager program: A program responsible for maintaining separate windows on the screen and implementing other window operations. These operations include moving windows and stretching windows.

working disc: A disc that you commonly use while working at the computer. *See also* **backup disc**.

write protect: A method of protecting information stored on a disc. The location of a tab or notch on the disc prevents write operations on the disc. A disc that is write protected can still be used for read operations.

Index

Page numbers in **bold type** are primary references; page numbers in regular type are secondary references.

D

E

G

H

I

K

L

M

N

O

P

Q

R

S

T

U

V

W