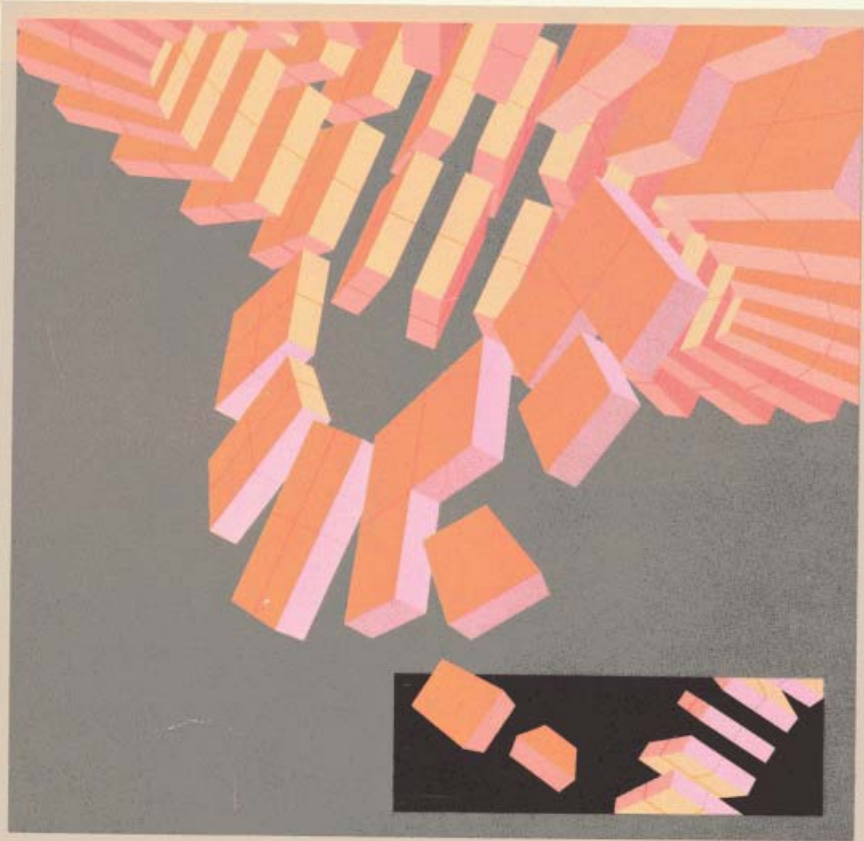


Apple II

ProDOS User's Manual



Customer Satisfaction

If you discover physical defects in the manuals distributed with an Apple product or in the media on which a software product is distributed, Apple will replace the documentation or media at no charge to you during the 90-day period after you purchased the product.

In addition, if Apple releases a corrective update to a software product during the 90-day period after you purchased the software, Apple will replace the applicable diskettes and documentation with the revised version at no charge to you during the six months after the date of purchase.

In some countries the replacement period may be different; check with your authorized Apple dealer. Return any item to be replaced with proof of purchase to Apple or an authorized Apple dealer.

Limitation on Warranties and Liability

Even though Apple has tested the software described in this manual and reviewed its contents, neither Apple nor its software suppliers make any warranty or representation, either express or implied, with respect to this manual or to the software described in this manual, their quality, performance, merchantability, or fitness for any particular purpose. As a result, this software and manual are sold "as is," and you the purchaser are assuming the entire risk as to their quality and performance. In no event will Apple or its software suppliers be liable for direct, indirect, incidental, or consequential damages resulting from any defect in the software or manual, even if they have been advised of the possibility of such damages. In particular, they shall have no liability for any programs or data stored in or used with Apple products, including the costs of recovering or reproducing these programs or data. Some states do not allow the exclusion or limitation of implied warranties or liability for incidental or consequential damages, so the above limitation or exclusion may not apply to you.

Copyright

This manual and the software (computer programs) described in it are copyrighted by Apple or by Apple's software suppliers, with all rights reserved. Under the copyright laws, this manual or the programs may not be copied, in whole or part, without the written consent of Apple, except in the normal use of the software or to make a backup copy. This exception does not allow copies to be made for others, whether or not sold, but all of the material purchased (with all backup copies) may be sold, given or loaned to another person. Under the law, copying includes translating into another language.

You may use the software on any computer owned by you but extra copies cannot be made for this purpose. For some products, a multi-use license may be purchased to allow the software to be used on more than one computer owned by the purchaser, including a shared-disk system. (Contact your authorized Apple dealer for information on multi-use licenses.)

Product Revisions

Apple cannot guarantee that you will receive notice of a revision to the software described in this manual, even if you have returned a registration card received with the product. You should periodically check with your authorized Apple Dealer.

© Apple Computer, Inc. 1983
20525 Mariani Avenue
Cupertino, California 95014
(408) 996-1010

Apple, ProDOS, ProFile, and the Apple logo are registered trademarks of Apple Computer, Inc.

Simultaneously published in the U.S.A. and Canada. All rights reserved.

Apple II

ProDOS User's Manual

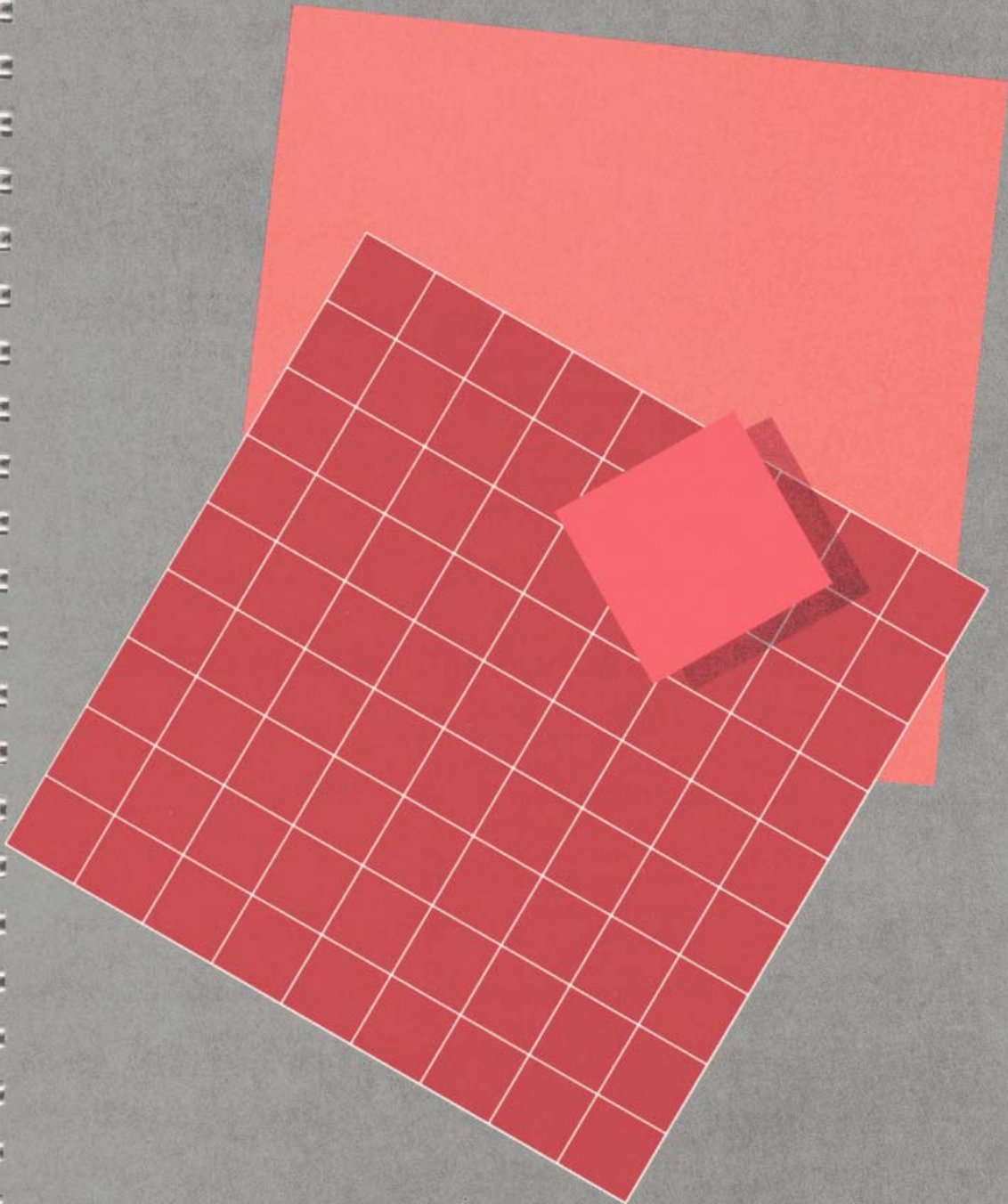




Table of Contents

	List of Figures and Tables	vii
	About This Manual	xi
	xi What You Need	
	xi Read Me Second	
	xii How to Use This Manual	
	xii Practice Makes Perfect	
	xii Summary Sections	
	xiii Visual Aids	
	Part I Introduction	1
	The ProDOS User's Disk	3
	1	
	5 Getting Started	
	6 How Menus Work	
	7 The <i>User's Disk</i> Main Menu	
	8 Tutor: ProDOS Explanation	
	8 ProDOS Filer	
	10 DOS-ProDOS Conversion	
	11 Display Slot Assignments	
	12 Display/Set Time	
	13 Applesoft BASIC	
	13 The Prompt	
	15 The Big Picture	
	15 How It Works	
	17 Summary of Chapter 1	

Part II The ProDOS Filer 19

- 19 The Filer Menu
- 20 The Filer Displays

2

About Volumes, Slots, and Drives 21

- 24 Naming Volumes
- 25 Slot and Drive Numbers
 - 25 Slot Numbers
 - 26 Drive Number
- 27 Summary of Chapter 2

3

Using the Volume Commands 29

- 33 Format a Volume
- 34 Here's How
- 36 Copy a Volume
- 37 Here's How
- 40 List Volumes
- 40 Here's How
- 41 Rename a Volume
- 42 Here's How
- 43 Detect Bad Blocks
- 44 Here's How
- 45 Block Allocation
- 46 Here's How
- 47 Compare Volumes
- 48 Here's How
- 49 Summary of Chapter 3
- 49 Format a Volume
- 50 Copy a Volume
- 51 List Volumes
- 51 Rename a Volume
- 51 Detect Bad Blocks
- 52 Block Allocation
- 52 Compare Volumes

4**About Filenames, Directories,
and Pathnames****53**

- 55 Filenames
- 56 Directories
- 58 Pathnames
- 59 Prefixes
- 59 An Example: Widgets, Inc.
- 63 Wildcards
- 63 The Equal Sign Wildcard
- 64 The Question Mark Wildcard
- 67 Summary of Chapter 4

5**Using the File Commands****69**

- 71 File Commands
- 73 Set Prefix
- 74 Here's How
- 75 List ProDOS Directory
- 76 Here's How
- 79 Copy Files
- 79 Here's How
- 81 Copying With a Wildcard
- 82 Delete Files
- 83 Here's How
- 84 Deleting With a Wildcard
- 85 Compare Files
- 85 Here's How
- 86 Alter Write-Protection
- 87 Here's How
- 88 Rename Files
- 88 Here's How
- 89 Make Directory
- 90 Here's How
- 91 Summary of Chapter 5
- 91 Set Prefix
- 91 List ProDOS Directory
- 92 Copy Files
- 92 Delete Files
- 93 Compare Files
- 93 Alter Write-Protection
- 94 Rename Files
- 94 Make Directory

6	<p>Configuration Defaults 95</p> <ul style="list-style-type: none"> 98 Select Defaults 99 Source Slot 99 Source Drive 99 Destination Slot 100 Destination Drive 100 Output Device 101 Restore Defaults 102 Summary of Chapter 6 102 Select Defaults 102 Restore Defaults <p>Part III The DOS-ProDOS Conversion Program 103</p>
7	<p>The DOS-ProDOS Conversion Program 105</p> <ul style="list-style-type: none"> 108 Using CONVERT 109 The Menu 111 Reverse Direction of Transfer 112 Change DOS 3.3 Slot and Drive 113 Set ProDOS Prefix 114 Set Prefix Using Pathname 115 Set Prefix Using Slot and Drive 116 Set ProDOS Date 117 Transfer (or List) Files 119 Summary of Chapter 7
A	<p>Appendix: Error Messages 121</p>
	<p>Glossary 137</p>
	<p>Index 145</p>

List of Figures and Tables

Chapter 1 The ProDOS User's Disk

- 6 Figure 1-1 The *ProDOS User's Disk* Startup Display:
The Main Menu
- 9 Figure 1-2 The Filer Menu
- 10 Figure 1-3 The DOS-ProDOS Conversion Program:
The CONVERT Menu
- 11 Figure 1-4 The Slot Assignments Display
- 12 Figure 1-5 The Time and Date Display
- 13 Figure 1-6 Applesoft BASIC
- 14 Figure 1-7 The *User's Disk*: The Big Picture
- 16 Figure 1-8 The Quit Display of the ProDOS Filer
- 17 Figure 1-9 How the *User's Disk* Works

Chapter 2 About Volumes, Slots, and Drives

- 25 Figure 2-1 The Expansion Slots of an Apple IIe
- 26 Figure 2-2 Disk Drive Controller Card

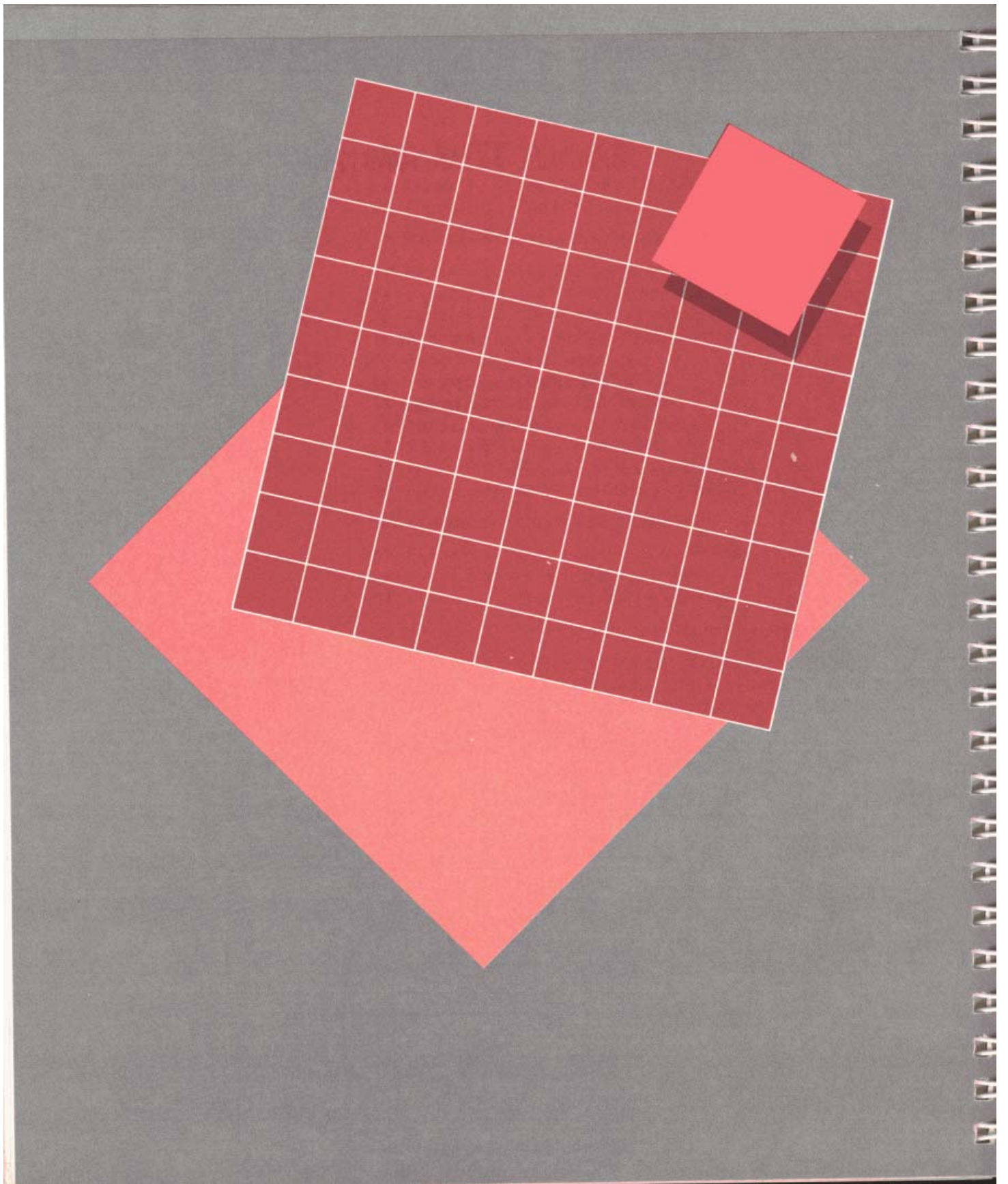
Chapter 3 Using the Volume Commands

- 31 Figure 3-1 The Volume Commands Menu
of the ProDOS Filer
- 33 Figure 3-2 A Formatted Disk
- 34 Figure 3-3 Format a Volume Display
- 37 Figure 3-4 Copy a Volume Display
- 40 Figure 3-5 List Volumes Display
- 42 Figure 3-6 Rename a Volume Display
- 44 Figure 3-7 Detect Bad Blocks Display
- 46 Figure 3-8 Block Allocation Display
- 47 Figure 3-9 Completed Block Allocation Display
- 48 Figure 3-10 Compare Volumes Display

Chapter 4 About Filenames, Directories, and Pathnames	
57	Figure 4-1 Organizing a Disk With Subdirectories
58	Figure 4-2 Pathnames
59	Figure 4-3 Using Prefixes
60	Figure 4-4 The Widgets, Inc. Filing System
62	Figure 4-5 The Hierarchy of Widgets' Records
Chapter 5 Using the File Commands	
71	Figure 5-1 The File Commands Menu
74	Figure 5-2 Set Prefix Display
76	Figure 5-3 List ProDOS Directory Display
77	Figure 5-4 Directory of <i>ProDOS User's Disk</i>
78	Table 5-1 File Types
79	Figure 5-5 Copy Files Display
83	Figure 5-6 Delete Files Display
85	Figure 5-7 Compare Files Display
87	Figure 5-8 Alter Write-Protection Display
88	Figure 5-9 Rename Files Display
90	Figure 5-10 Make Directory Display
Chapter 6 Configuration Defaults	
97	Figure 6-1 Configuration Defaults Menu
98	Figure 6-2 Select Defaults Display
101	Figure 6-3 Restore Defaults Display
Chapter 7 The DOS-ProDOS Conversion Program	
108	Figure 7-1 CONVERT Menu
109	Figure 7-2 CONVERT Menu Showing Transfer Information
110	Figure 7-3 The CONVERT Quit Display
112	Figure 7-4 Change DOS 3.3 Slot and Drive Display
113	Figure 7-5 Set ProDOS Prefix Display
114	Figure 7-6 Pathname Display
115	Figure 7-7 Slot and Drive Display
116	Figure 7-8 Set ProDOS Date Display
117	Figure 7-9 Transfer (or List) Files Display
118	Table 7-1 The Differences Between ProDOS and DOS Filenames

About This Manual

- xi** What You Need
- xi** Read Me Second
- xii** How to Use This Manual
- xii** Practice Makes Perfect
- xii** Summary Sections
- xiii** Visual Aids



About This Manual

This manual explains how to use the *ProDOS User's Disk*. This disk uses ProDOS™ to manage information on disks. The *User's Disk* is designed so that you can use ProDOS without having to memorize ProDOS commands. If you want to know more about ProDOS, see the *ProDOS Technical Reference Manual* and *BASIC Programming With ProDOS*.

What You Need

To get the most out of this manual, you should have

- an Apple II with **64K RAM** and a **display device**
- at least one disk drive
- the *ProDOS User's Disk*
- two blank disks

64K RAM stands for 64 kilobytes of random-access memory. A **kilobyte** is 1024 bytes or 8192 bits. A **byte** is eight **bits** and can hold one character.

Random-access memory is memory whose contents you can change.

Your **display device** can be either a television set or a video monitor.

Note: References in this manual to the Apple II refer to the Apple II Plus and the Apple IIe. The *ProDOS User's Disk* will *not* work on an Apple II. This is because Applesoft BASIC must be in ROM (read-only memory) for the disk to work. Also, to use ProDOS, your computer must have at least 64 kilobytes (K) of random-access memory (RAM). Most Apple II computers have 48K or 64K RAM. Some earlier models have 16K or 32K RAM. If you're uncertain as to your Apple computer's memory capacity, consult your Apple dealer.

Read Me Second

Before you start this manual, you should have a working relationship with your Apple II computer and disk drive. That doesn't mean you need to know how to program in four languages, but you should already have your disk drive hooked up to your Apple II; you should know how to put disks into your disk drive; and you should have some familiarity with the Apple II keyboard. If you aren't yet at home with your Apple II, read the introductory material that came with your computer. Then return to this manual.

How to Use This Manual

This manual is divided into three parts. The first part is a brief introduction to the *User's Disk*. Because the *User's Disk* is fairly self-explanatory, this may be the only part of the manual you will need to read.

The second part discusses the ProDOS Filer in detail. If you are not familiar with how a hierarchical file structure works and how to use pathnames, you may want to read Chapters 2 and 4. Chapters 3 and 5 tell you how to use the volume and file commands. These chapters are written so that you may use them just for reference. But don't hesitate to work through the chapters, trying out every command. Chapter 6 is about the default assumptions used by the Filer.

The third part explains how to use the DOS-ProDOS Conversion Program included on the *User's Disk*. This program allows you to convert your DOS 3.3 files to ProDOS files and vice versa.

Practice Makes Perfect

Reading a recipe for whole wheat bread is not the same as mixing the ingredients, kneading the dough, and baking the bread. Similarly, reading about how to copy a file is not the same as actually copying one. Try out each command as you read about it. That's the best way to find out if you understand the material.

Summary Sections

Some people like to race through manuals; others like to take it slow and steady. The skimmers among you will appreciate the summaries at the end of each chapter. Each summary is a digest of all the new commands and vocabulary presented in that chapter.

You may use the summaries as a quick reference or to test your mastery of the material you've just covered. If any of the commands are still hazy, it's a clue to go back and review before tackling new material. If you're an old hand with computers and disk operating systems, the summaries may be all the instruction you need.

Visual Aids

As you use this manual, you'll notice that typeface and shading have special meanings.

See the margin for notes on a **boldface** term.

Computer terms and words with which you may not be familiar appear in **boldface**. A special type is used for what you type and what you see on the display: `it looks like this.`

When you see a hyphen joining two keys, it means to press the keys simultaneously. For instance, `(CONTROL)-(RESET)` means you should press `(CONTROL)` and `(RESET)` at the same time. In actual practice, you probably will press `(CONTROL)` first and then, while still holding down `(CONTROL)`, press `(RESET)`.

By the Way: Helpful hints and interesting sidelights appear in gray boxes like this one.

Note: Important information or concepts appear in gray boxes like this one.



Warning

Warnings about potential problems and advice about how to avoid them appear in boxes like this one.

Read the marginal notes for quick reference.

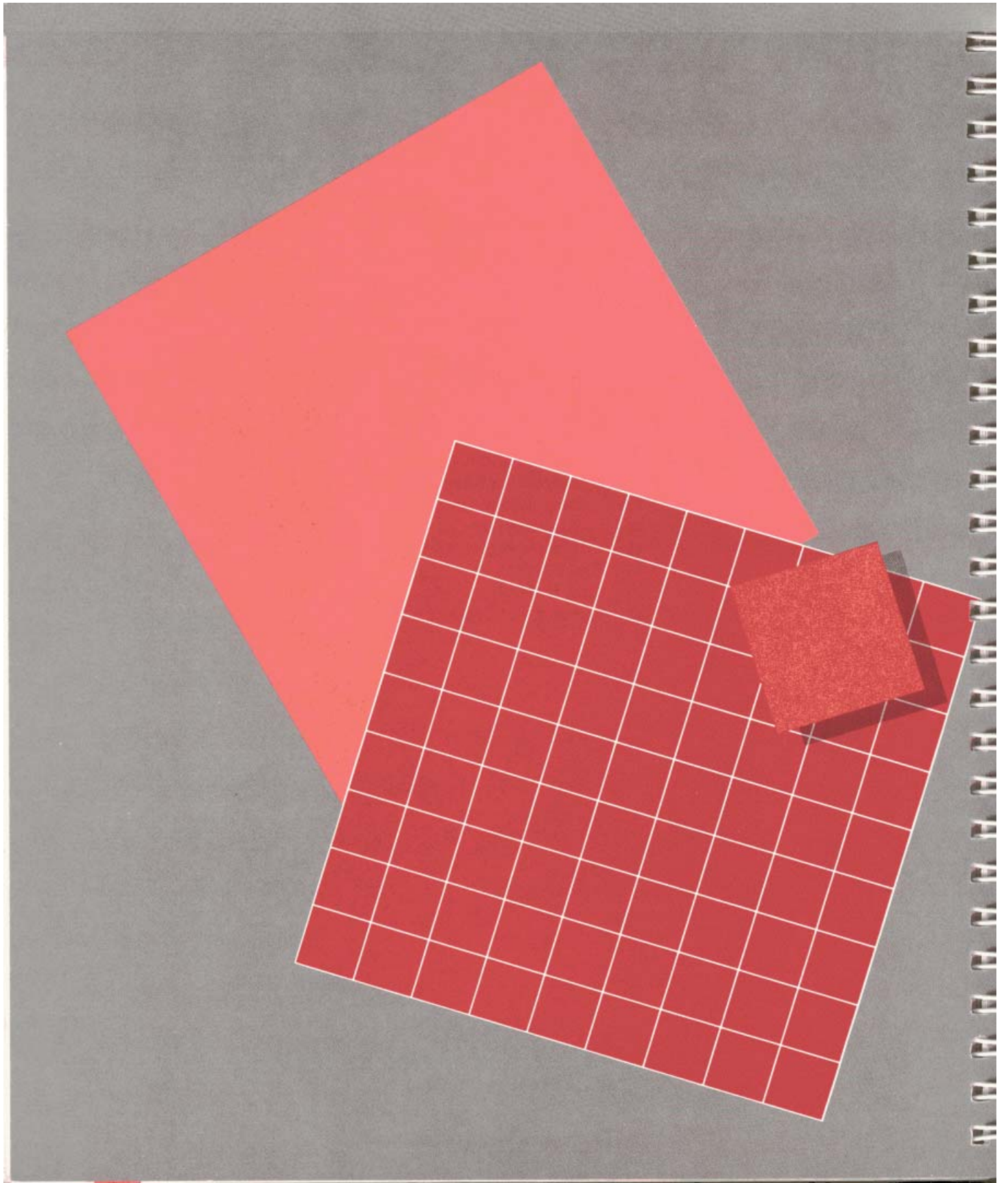
You'll also find notes in the margin that emphasize a point, define terms, or refer you to related information in another part of the book.

Introduction

The *ProDOS User's Disk* uses ProDOS and a set of programs to help you manage the information on your disks. But you don't have to learn ProDOS commands or know anything about programming because the *User's Disk* does all the work for you. Part I is an introduction to using this disk.

The ProDOS User's Disk

-
- 5 Getting Started
 - 6 How Menus Work
 - 7 The *User's Disk* Main Menu
 - 8 Tutor: ProDOS Explanation
 - 8 ProDOS Filer
 - 10 DOS-ProDOS Conversion
 - 11 Display Slot Assignments
 - 12 Display/Set Time
 - 13 Applesoft BASIC
 - 13 The Prompt
 - 15 The Big Picture
 - 15 How It Works
 - 17 Summary of Chapter 1



The ProDOS User's Disk

A **disk** can be either rigid or flexible. A **flexible**, or **floppy**, disk is a thin, flexible circle of magnetized material. A **rigid**, or **hard**, disk is made of rigid, nonflexible magnetized material. The magnetized material is where information is stored, much as music is recorded on magnetic tape.

ProDOS stands for *Professional Disk Operating System*. It differs from the previous Apple II disk operating system (DOS 3.3) in its ability to use drives for **rigid disks**, such as the ProFile™, as well as drives for **flexible disks**.

Note: References in this manual to the Apple II refer to the Apple II Plus and the Apple IIe. The *ProDOS User's Disk* will *not* work on an Apple II. This is because Applesoft BASIC must be in ROM (read-only memory) for the disk to work. Also, to use ProDOS, your computer must have at least 64 kilobytes (K) of random-access memory (RAM). Most Apple II computers have 48K or 64K RAM. Some earlier models have 16K or 32K RAM. If you're uncertain as to your Apple computer's memory capacity, consult your Apple dealer.

Note: A **disk operating system** controls the operation of your disk drives. It prepares disks to receive information, maintains a disk directory that organizes all the information on the disk, allows you to copy disks, lets you move information from disk to disk, and helps you perform many other chores. DOS is pronounced like *boss*.

Getting Started

The *ProDOS User's Disk* is a menu-driven disk. This means that your options are presented to you through a series of menus.

If you haven't done so already, put your *User's Disk* in drive 1 and turn on your computer system. You'll see the display shown in Figure 1-1.

Figure 1-1. The *ProDOS User's Disk* Startup Display: The Main Menu

```
*****
*                                     *
*          PRODOS USER'S DISK        *
*                                     *
*  COPYRIGHT APPLE COMPUTER, INC. 1983  *
*                                     *
*****

YOUR OPTIONS ARE:

    ? - TUTOR: PRODOS EXPLANATION
    F - PRODOS FILER (UTILITIES)
    C - DOS <-> PRODOS CONVERSION
    S - DISPLAY SLOT ASSIGNMENTS
    T - DISPLAY/SET TIME
    B - APPLESOFT BASIC

PLEASE SELECT ONE OF THE ABOVE ❖
```

How Menus Work

If you are familiar with menu-driven programs, you probably know what to do. If you're new to computer menus, consider this analogy:

You've just pulled into Eddie's All-Night Eats on Highway 5. A waiter shuffles over and hands you a breakfast menu that reads

```
EGGS
JUICE
TOAST
COFFEE
```

You order eggs. Eddie realizes, from your answer, that you're not a regular at his truck stop, so he outlines your egg options.

"Do you want your eggs scrambled, fried, or poached?" he asks.

It's as if there was a menu within a menu. If you had ordered juice, you would have had a choice of orange, tomato, or prune. If you had ordered toast, you could have had white, wheat, or rye bread.

The Main Menu of the *User's Disk* is set up much the same way, except that when you make a selection, you don't have to rely on anyone's memory to find out what's on the next menu. Everything is spelled out for you. That's the beauty of a menu-driven program. It's so easy to use that you could almost get by without a manual!

The User's Disk Main Menu

Now that you know how computer menus work, take a look at the options on the Main Menu of the *User's Disk*.

Here, instead of food, your options are

- ? - TUTOR: PRODOS EXPLANATION
- F - PRODOS FILER (UTILITIES)
- C - DOS <-> PRODOS CONVERSION
- S - DISPLAY SLOT ASSIGNMENTS
- T - DISPLAY/SET TIME
- B - APPLESOFT BASIC

When you select an option from the Main Menu, you type the letter that precedes the option you want. You do *not* need to press (RETURN). The Main Menu will disappear, and you'll see—depending upon the option you choose—a Tutor display, another menu, or an informational display.

A Note About Error Messages: Occasionally, you'll type a number when you meant to type a letter, or you'll send your computer off in search of a disk that's nowhere near your disk drive. When this sort of thing happens, you'll hear a beep alerting you to some failure in the communication process. When you hear such a beep, check the display for an error message.

If you're not sure what the error message means or what to do about it, consult Appendix A, "Error Messages."

The following sections briefly explain each of the elements of the Main Menu. For now, just read this information.

Tutor: ProDOS Explanation

The Tutor display that you get from this menu tells you what an operating system does and why you need to know a little about the DOS operating system as well as about the ProDOS operating system. All of the menus on the *User's Disk* offer a Tutor option. Each Tutor display gives specific information about the part you are using.

ProDOS Filer

The ProDOS Filer is a program that uses ProDOS to allow you to organize the information you store on disks. This is the option you'll use more than any other. The Filer has one group of commands, called **volume commands**, that works with the disk as a whole and another group of commands, called **file commands**, that works with individual files. Together, these commands are often referred to as *utilities* because they let you *use* your computer.

Note: A **file** is a small body of information—like a business report or a budget—on a disk. A **volume** is a collection of files. Terminology relevant to the file and volume commands is presented in Chapters 2 and 4.

See Chapter 3 for how to use the **volume commands**.

See Chapter 5 for how to use the **file commands**.

If the definition of a ProDOS term or any other computer-specific word is not in the margin or the text, check the Glossary.

When you choose the Filer option, you'll see another menu, shown in Figure 1-2.

Figure 1-2. The Filer Menu

```
*****
*
*   APPLE'S PRODOS SYSTEM UTILITIES   *
*
*           FILER   VERSION 1.0      *
*
*   COPYRIGHT APPLE COMPUTER, INC., 1983 *
*
*****

      ? - TUTOR
      F - FILE COMMANDS
      V - VOLUME COMMANDS
      D - CONFIGURATION DEFAULTS
      Q - QUIT

PLEASE SELECT AN OPTION: *
```

Whether you use your Apple computer for word processing, financial modeling, programming, or hobgoblin games, you will need to organize the files on your disks. The ProDOS Filer exists for that purpose—organizing. There are Filer commands for preparing disks to receive the files you create, for copying files from one place to another, and for deleting files you don't need anymore.

Important! If you change the prefix while using the Filer, you should send the prefix back to the name of your startup disk before leaving the Filer for another part of the *ProDOS User's Disk*.

Put the *User's Disk* (or your startup disk) in drive 1 and then select the Set Prefix option on the File Command Menu.

Part II discusses the ProDOS Filer in detail.

DOS-ProDOS Conversion

When you choose the DOS-ProDOS Conversion option, you get a program named CONVERT. The CONVERT Menu is shown in Figure 1-3.

Figure 1-3. The DOS-ProDOS Conversion Program: The CONVERT Menu

```
                CONVERT Menu
Direction: DOS 3.3 S6,D2 ---> ProDOS
Date: <NO DATE>
Prefix: /USERS.DISK/

-----

R - Reverse Direction of Transfer
C - Change DOS 3.3 Slot and Drive
D - Set ProDOS Date
P - Set ProDOS Prefix
T - Transfer (or List) Files

-----

Enter Command: ?  ? - Tutor, Q - Quit
```

The CONVERT program lets you convert DOS 3.3 files to ProDOS files and vice versa.

If you have been using any games or application programs based on DOS 3.3, you must use this program before you can use the game or application program with ProDOS.

Important! If you change the prefix while using CONVERT, you should send the prefix back to the name of your startup disk before moving to another part of the *ProDOS User's Disk*.

Put the *User's Disk* (or your startup disk) in drive 1 and then select the Set ProDOS Prefix option.

Part III explores this program in detail.

The **startup disk** is the one you use to get ProDOS running.

A **peripheral card** is a removable printed-circuit board that plugs into an expansion slot and expands or modifies the computer's capabilities.

An **expansion slot** is a long, narrow connector inside the Apple II in which a peripheral card, such as a disk controller card, can be installed.

Figure 1-4. The Slot Assignments Display

Display Slot Assignments

The Display Slot Assignments option gives you information about the computer system you are using. When you select this option, you'll see the name of the **startup disk**, how much memory your Apple computer has, and a display similar to Figure 1-4. The display shows you which **peripheral cards** you have in each **expansion slot**.

```
*****
*
*          DISPLAY SLOT ASSIGNMENTS          *
*
*****
STARTUP DISK:  /USERS.DISK/
YOUR Apple //e HAS:
    64K OF RANDOM ACCESS MEMORY
    APPLESOFT IN ROM
SLOT 1:  SILENTYPE
SLOT 2:  EMPTY
SLOT 3:  80-COLUMN CARD
SLOT 4:  THUNDERCLOCK
SLOT 5:  PROFILE
SLOT 6:  DISK DRIVE
SLOT 7:  EMPTY
PRESS RETURN TO DISPLAY MAIN MENU  ⌘
```

A **disk controller card** is a peripheral card that connects one or two disk drives to the Apple II and controls their operation.

Of course, the information you see on your display may be different from that shown in Figure 1-4, depending on what you have connected to your computer. You might want to make a note of the number of the slot that holds your **disk controller card**. You'll need to use this number again and again.

An **integrated circuit** is an electronic component that has many circuits on a single piece of semiconducting material, such as silicon—often called a **chip**, although the chip is just the piece of silicon.

Display/Set Time

If you don't have one of the peripheral cards or **integrated circuits** installed in the computer that keeps time for you, the Display/Set Time option lets you set the date and time manually.

When you choose this option, you see the display shown in Figure 1-5.

Figure 1-5. The Time and Date Display

```
*****
*
*   SYSTEM DATE AND TIME UTILITY   *
*
*
*****

THE CURRENT SETTINGS ARE:
      DATE: <NO DATE>
      TIME: <NO TIME>

UPDATE SYSTEM DATE AND TIME? Y/N  ⌘
```

Many of the utilities on the *User's Disk* record the date and time when you use them. This is especially useful when you are saving different versions of something and want to remember the sequence.

Applesoft BASIC

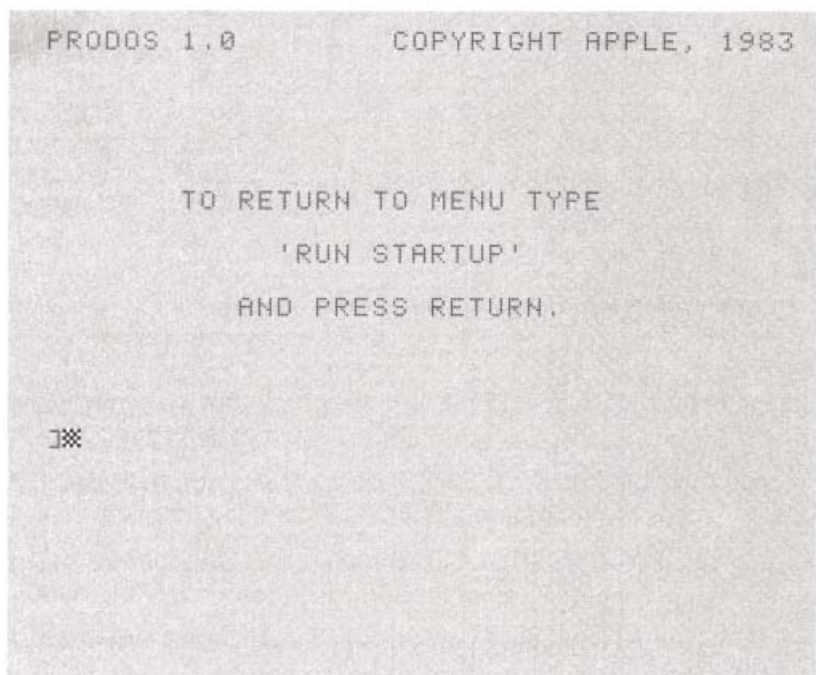
Selecting the Applesoft BASIC option puts you into BASIC. Unless you are familiar with programming, you probably won't be selecting this option. However, if you do, and want to get back to the *User's Disk Main Menu*, type

```
RUN STARTUP
```

as shown in Figure 1-6.

For an explanation of why you type this, see the section "How It Works."

Figure 1-6. Applesoft BASIC



For more information, see *BASIC Programming With ProDOS*.

A **prompt** is a message from the program that asks you for information.

The Prompt

The last line on the menu is called a **prompt**; it's there to remind you to do something. On the *User's Disk Main Menu*, you see this prompt:

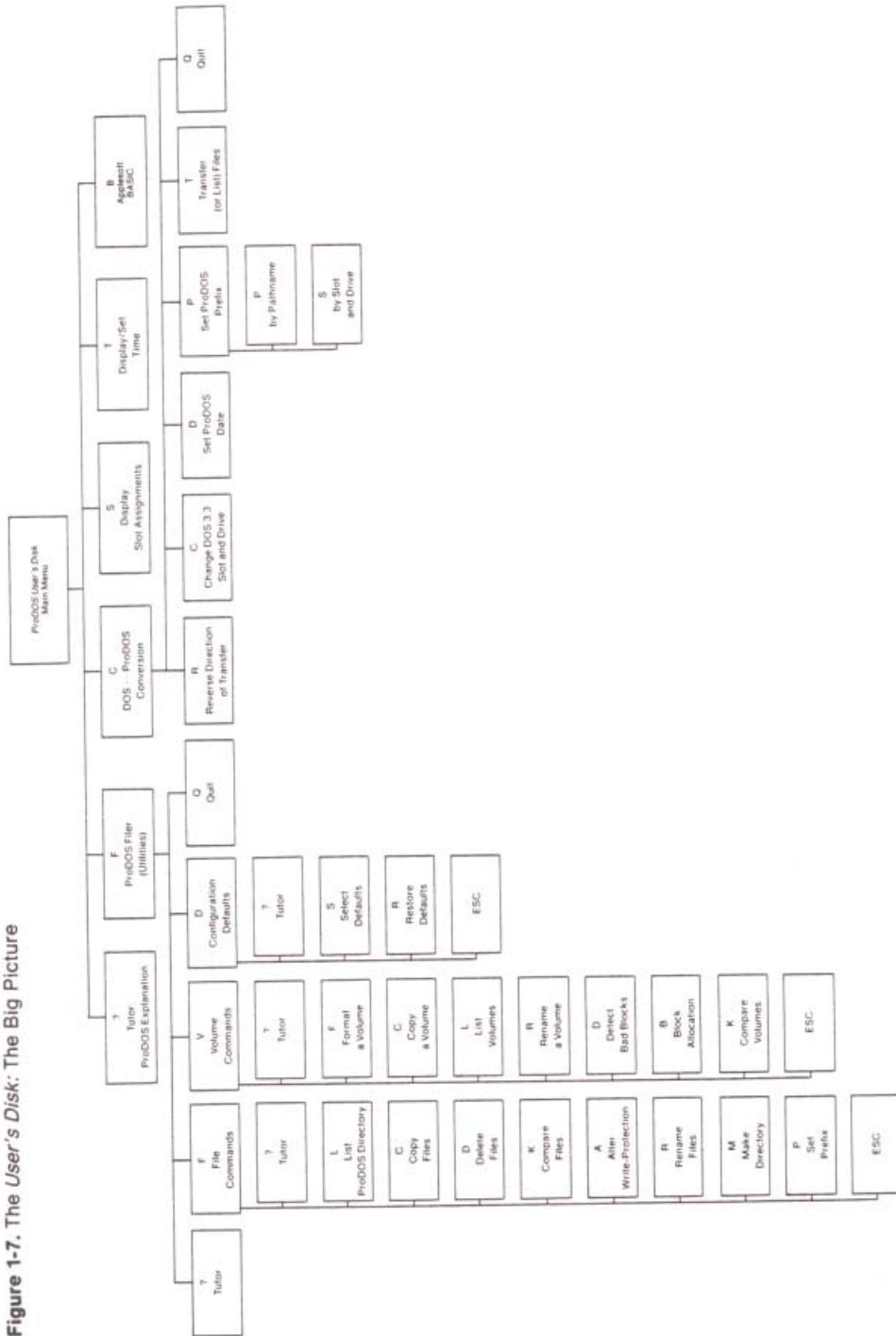
```
PLEASE SELECT ONE OF THE ABOVE ⌘
```

In other words, to select one of the options from the Main Menu, you simply type the character corresponding to your choice.

The prompts vary, as you'll soon see, but they are there to make your life easier. If you forget what you're supposed to do at any given time, consult the prompt.

The User's Disk Main Menu

Figure 1-7. The User's Disk: The Big Picture



The Big Picture

From the Main Menu, you can take many paths through the *ProDOS User's Disk*. For those of you who like to see the big picture, Figure 1-7 is a map to your choices. If you find this sort of diagram overwhelming, don't despair: you will only work with one display at a time, and there's a prompt at the bottom of each display that outlines your options.

How It Works

The *User's Disk* is a collection of **system programs**. These programs are named

PRODOS
BASIC.SYSTEM
STARTUP
FILER
CONVERT

A **system program** makes the computer available for general purposes.

Main memory is the part of the Apple II computer that is used to store information.

However, not all of these programs can fit into the **main memory** of the Apple II at the same time.

When you start your computer with the *User's Disk* in drive 1, several things happen:

1. The system program named PRODOS, containing the most sophisticated parts of ProDOS, is put into memory.
2. The system program named BASIC.SYSTEM is also put into memory.
3. Applesoft BASIC looks for a program named STARTUP. If it finds one, it runs that program. In the case of the *User's Disk*, the STARTUP program is the Main Menu.

The end result is that you see the Main Menu of the *User's Disk*. When you select the ProDOS Filer (Utilities) or the DOS-ProDOS Conversion options from the Main Menu, the system program for those options takes the place of BASIC.SYSTEM and STARTUP in memory.

That's why the Filer Menu and the CONVERT Menu have Quit options. When you choose to quit, you see the Quit display shown in Figure 1-8.

Figure 1-8. The Quit Display of the ProDOS Filer

```
*****
*
*                               QUIT                               *
*
**PREFIX: /USERS.DISK/*****
--QUIT AND LOAD--
  PATHNAME: (BASIC.SYSTEM)
--ENTER PATHNAME AND PRESS <RET>--
```

You use this display to put the BASIC system program back into memory and run the STARTUP program, which is the Main Menu of the *User's Disk*.

This may sound complicated, but all you have to do is accept the **default** whenever you quit the Filer or the Conversion Program.

A **default** is the computer's best guess at what you are about to do next. If the computer is correct, just press **(RETURN)**.

Figure 1-9 illustrates how ProDOS shifts the programs that make up the *User's Disk* in and out of memory. ProDOS is the hub of the disk and is always in memory. When you start up your computer, BASIC.SYSTEM and STARTUP are automatically put in memory. When you shift to the Filer, that program (FILER) replaces BASIC.SYSTEM and STARTUP in memory. When you shift to the Conversion program, CONVERT replaces BASIC.SYSTEM and STARTUP in memory. When you quit either the Filer or the Conversion program, BASIC.SYSTEM and STARTUP are put back in memory and you return to the Main Menu.

Figure 1-9. How the *User's Disk* Works

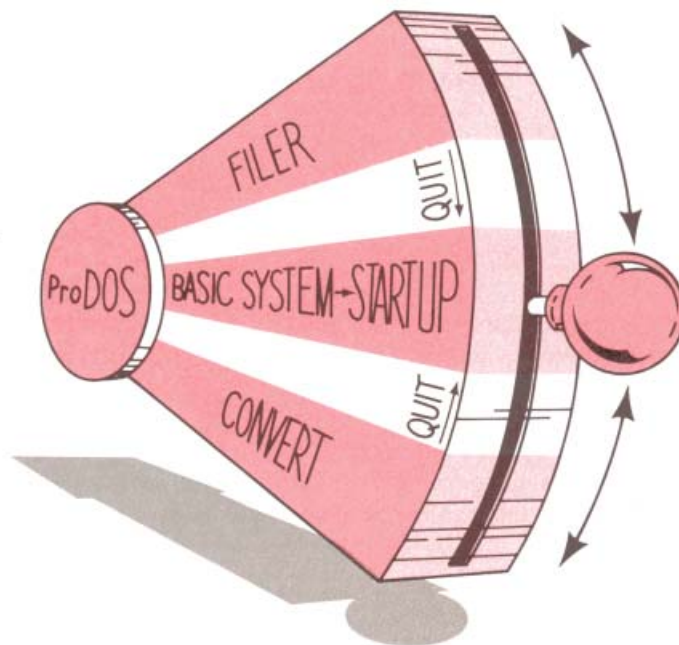
PRODOS holds the sophisticated ProDOS features.

BASIC.SYSTEM holds most ProDOS commands.

STARTUP holds the *User's Disk* Main Menu.

FILER holds the ProDOS Filer (Utilities).

CONVERT holds the DOS-ProDOS Conversion Program.



Summary of Chapter 1

Menus: Give you a list of choices. You communicate your choice by typing the key associated with that choice.

User's Disk Main Menu: Offers six options.

```
? - TUTOR: PRODOS EXPLANATION
F - PRODOS FILER (UTILITIES)
C - DOS <-> PRODOS CONVERSION
S - DISPLAY SLOT ASSIGNMENTS
T - DISPLAY/SET TIME
B - APPLESOFT BASIC
```

Tutor: If you press (?), you get a display with information and hints outlining your options.

Prompt: Appears at the bottom of each display to remind you of your options.

The ProDOS Filer

The ProDOS Filer is designed to help you organize the information you store on disks. You'll use the Filer more than any other part of the *User's Disk*.

The Filer commands let you prepare disks to receive information, copy whole disks, copy files from one place to another, delete files you don't need anymore, and accomplish many other utilities.

The Filer Menu

The Filer Menu has five options you can choose from:

- Tutor
- File Commands
- Volume Commands
- Configuration Defaults
- Quit

You see the Tutor displays by pressing **[?]**. The Tutor gives you information about the Filer commands and defines terminology that may be unfamiliar to you.

Volume commands are those that affect the entire disk. Chapter 2 presents background information, definitions, and naming conventions for **volumes**. Chapter 3 gives you the practical information for using each of the commands.

File commands are those that affect individual files. Chapter 4 presents background information on filenames, directories, and pathnames. You'll want to read this chapter if you aren't familiar with **files** and **pathnames**. Chapter 4 also explains how the hierarchical file structure of ProDOS works and how to use **wildcards** with the file commands. Chapter 5 gives you the practical information for using each of the commands.

The five chapters in Part II discuss the main part—file commands, volume commands, and configuration defaults—of the Filer in detail.

A **volume** is another name for a disk, either flexible or rigid. It is a collection of files.

A **file** is a collection of information stored on a disk.

A **pathname** is the path the computer takes to find a file. It always begins with the name of the volume and ends with the name of the file.

Wildcards can be a handy shortcut to file handling.

See "How It Works" in Chapter 1 for an explanation of the Quit display.

Chapter 6 explains configuration defaults and the commands that allow you to customize the defaults to match your system setup.

You will use the Quit option when you want to return to the Main Menu of the *User's Disk*. When you choose this option, accept the default answer on the Quit display, by pressing **(RETURN)**, to return to the Main Menu.

The Filer Displays

You will need to provide some information on many of the displays you'll encounter as you use ProDOS Filer.

A **default** is provided by the computer.

The Filer usually supplies an answer for you. This answer is the program's best guess. It is called a **default** because the answer is used by default if you don't change it.

To accept a default as is, press **(RETURN)**.

The **cursor** marks where the next character you type will appear on the display.

To change the default, type over the character or string of characters that appears on the display. If it's a string of characters, such as a pathname, press **(RETURN)** when the **cursor** is at the end of the string. This indicates that you've completed your entry. If it's a single character response, such as a slot number, you don't need to press **(RETURN)**.

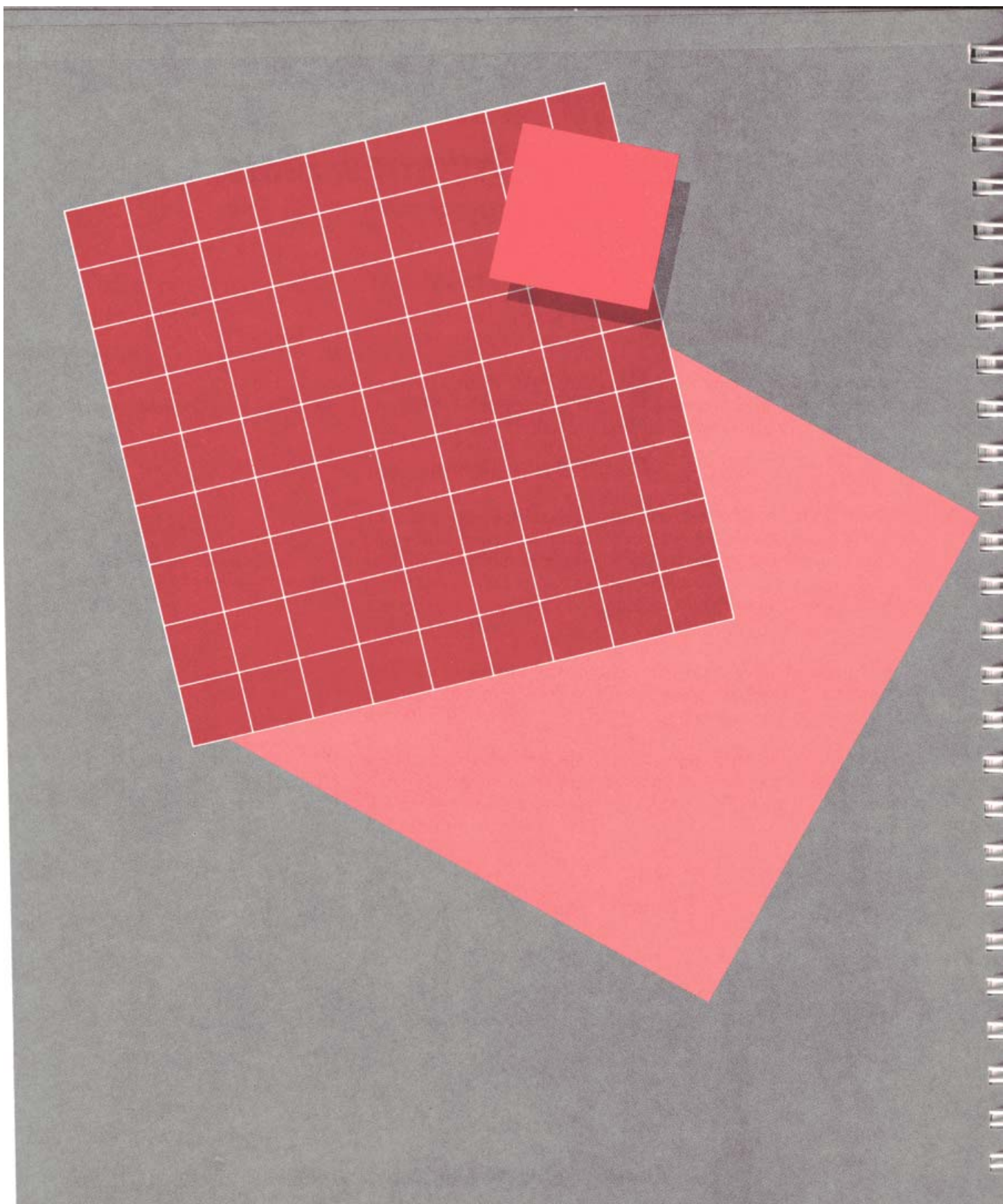
To edit a string of characters, press **(←)** to move the cursor over the characters you want to accept and type over the characters you want to change. You cannot insert characters into the string.

If you make a mistake while typing a string of characters, you can press **(←)** and move the cursor back to where you want to make the correction. But don't forget to press **(→)** to get back to the end of the string before pressing **(RETURN)**.

Note: Once you are in the Filer, you can return to the previous menu by pressing **(ESC)**.

About Volumes, Slots, and Drives

- 24 Naming Volumes
- 25 Slot and Drive Numbers
 - 25 Slot Numbers
 - 26 Drive Numbers
- 27 Summary of Chapter 2



About Volumes, Slots, and Drives

Chapter 3 discusses how to use each of the volume commands.

The **ProFile** is a mass storage device manufactured by Apple Computer, Inc. It holds information equivalent to dozens of flexible disks on two nonremovable rigid disks.

One of the five options on the Filer Menu is Volume Commands. This group of commands works with the entire disk. This chapter provides some background information on volumes, slot numbers, and drive numbers—the stuff of which volume commands are made.

Volume is another term for *disk*, just as *canine* is another term for *dog*. The volume most commonly used with an Apple II is the flexible, or floppy, disk. But the term *volume* can also apply to rigid disks, like the **ProFile**. Because you can use ProDOS commands on flexible and rigid disks, *volume* is used to refer to those commands that allow you to format, copy, alter, and otherwise scan whole disks at one time.

If you have trouble thinking of disks as volumes, look at it this way:

Volume is a word associated with books. Encyclopedias come in volumes. *The Rise and Fall of the Roman Empire* comes in volumes. It's a handy way of dividing information into manageable chunks. So it's not surprising that computer libraries also come in volumes.

Think of each flexible disk as one volume in your computer library.

A ProFile is a volume too, but its capacity is so great (one five megabyte ProFile disk equals 35 flexible disks) that it's like getting a whole encyclopedia in one volume.

Naming Volumes

Like books, volumes are identified by name. You can be as creative as you like when you name your volumes, but you must observe some rules.

Volume names must

- begin with a slash, then a letter.
- be made up of letters, numbers, or periods.
- not have any spaces or punctuation characters other than periods.
- not exceed 15 characters (not counting the slash).

Here are some acceptable volume names:

`/USERS.DISK` The name of the disk you're using right now.

`/RISE.AND.FALL` Notice how periods are a convenient substitute for spaces.

`/LETTER.7.29` You can use numbers, too.

Here are some unacceptable names:

`/THE.RISE.AND.FALL.OF.THE.ROMAN.EMPIRE` Too long

`/RISE AND FALL` Spaces aren't allowed.

`/3.PENNY.OPERA` You can't start a volume name with a number.

By the Way: If you forget and try to enter an unacceptable name, no harm's done. The Filer won't accept an invalid name.

Slot and Drive Numbers

You'll be identifying the volume you want to use for a particular operation by its slot and drive number. (If you have a ProFile, you'll only be asked for the slot number).

Slot Numbers

When a program asks you to supply the slot number for a given volume, it's really just asking you for that volume's address. But you can't supply the address unless you know how your disk drive is hooked up to your computer. So take a minute to review the way that disk drives are connected to cables, the way cables are connected to controller cards, the way controller cards are connected to computers, and what all this has to do with volume commands.

External storage devices, like drives for flexible disks and ProFiles, are attached to your computer by cables connected to **controller cards** (printed-circuit boards) plugged into narrow connectors on the **main logic board** inside your computer. Those connectors are called **expansion slots**, and they're clearly numbered.

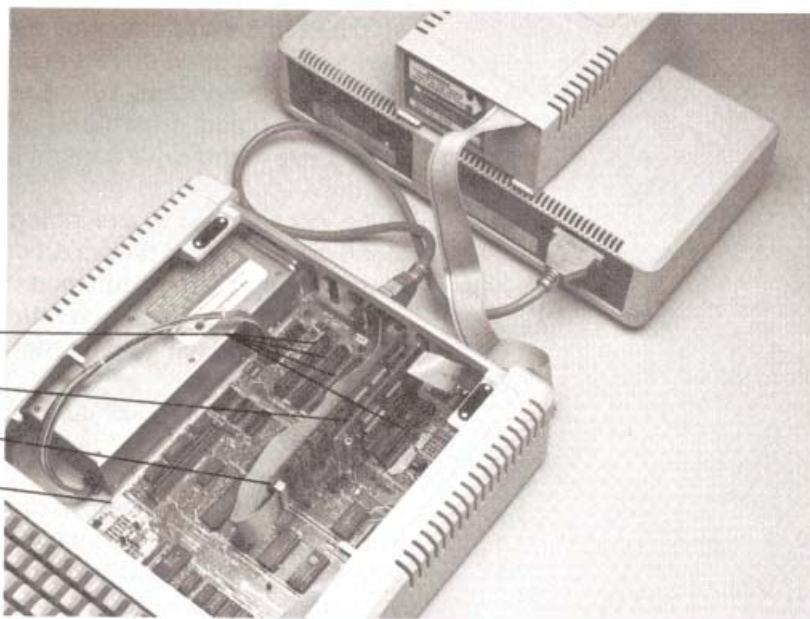
A **controller card** is a peripheral card that controls the operation of a peripheral device, such as a disk drive or a printer.

The **main logic board** is a circuit board into which various integrated circuits and printed-circuit boards are plugged.

An **expansion slot** is a long, narrow connector in which a peripheral card can be installed.

Figure 2-1. The Expansion Slots of an Apple IIe. Notice that the disk drive is connected to a controller card in slot 6 and the ProFile is connected to a controller card in slot 5.

Expansion Slots
Controller Card for Disk Drive
Controller Card for ProFile
Main Logic Board



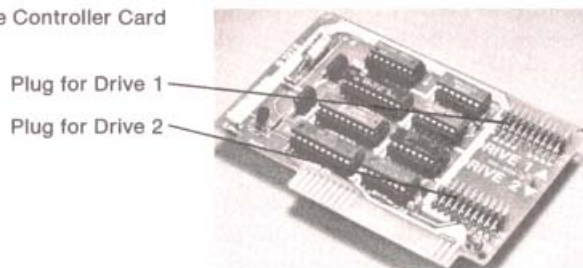
If you're asked for the volume address of a ProFile, all you have to provide is the slot number. For drives for flexible disks, you also need to provide a drive number. The reason you have to specify a drive number is that there can be two disk drives attached to one card in one slot.

Drive Numbers

There is usually a label on the outside of a drive for flexible disks, telling you whether it's drive 1 or drive 2. But those labels are for your convenience; they don't mean a thing to the computer. The computer distinguishes drive 1 from drive 2 based on how the disk drive cables are connected to the controller card.

If you have one disk drive, it's attached to the upper set of pins on the controller card. Those pins are labeled *DRIVE 1*. If you have a second disk drive, it is attached to the lower set of pins on that same controller card. The lower pins are labeled *DRIVE 2*.

Figure 2-2. Disk Drive Controller Card



If you have more than two disk drives, the additional drives are attached to a second controller card. Like the first two disk drives, they are designated drive 1 and drive 2. The only way to tell them apart from the original drive 1 and drive 2 is by their slot number. While the original drive 1 and drive 2 are hooked up to slot 6, the other drive 1 and drive 2 might be hooked up to slot 5.

Once you get used to describing a volume's address as drive 1 in slot 6, to distinguish it from drive 1 in slot 5, it's no more confusing than referring to the addresses of two houses with the same number that are located on different streets or referring to two streets with the same name that are located in different cities.

So much for theory. How do you find out which drive is which and how your controller card is plugged into the computer?

The easiest way is to choose the Display Slot Assignments option from the *User's Disk* Main Menu: it will tell you exactly what is connected to what. Another way is to turn off the power, open up the computer and look for yourself.

Summary of Chapter 2

Volume: Another term for *disk*. The volume used most commonly with the Apple II is the flexible disk. The term *volume* also applies to rigid disks like the ProFile.

Volume commands: Commands that affect whole disks.

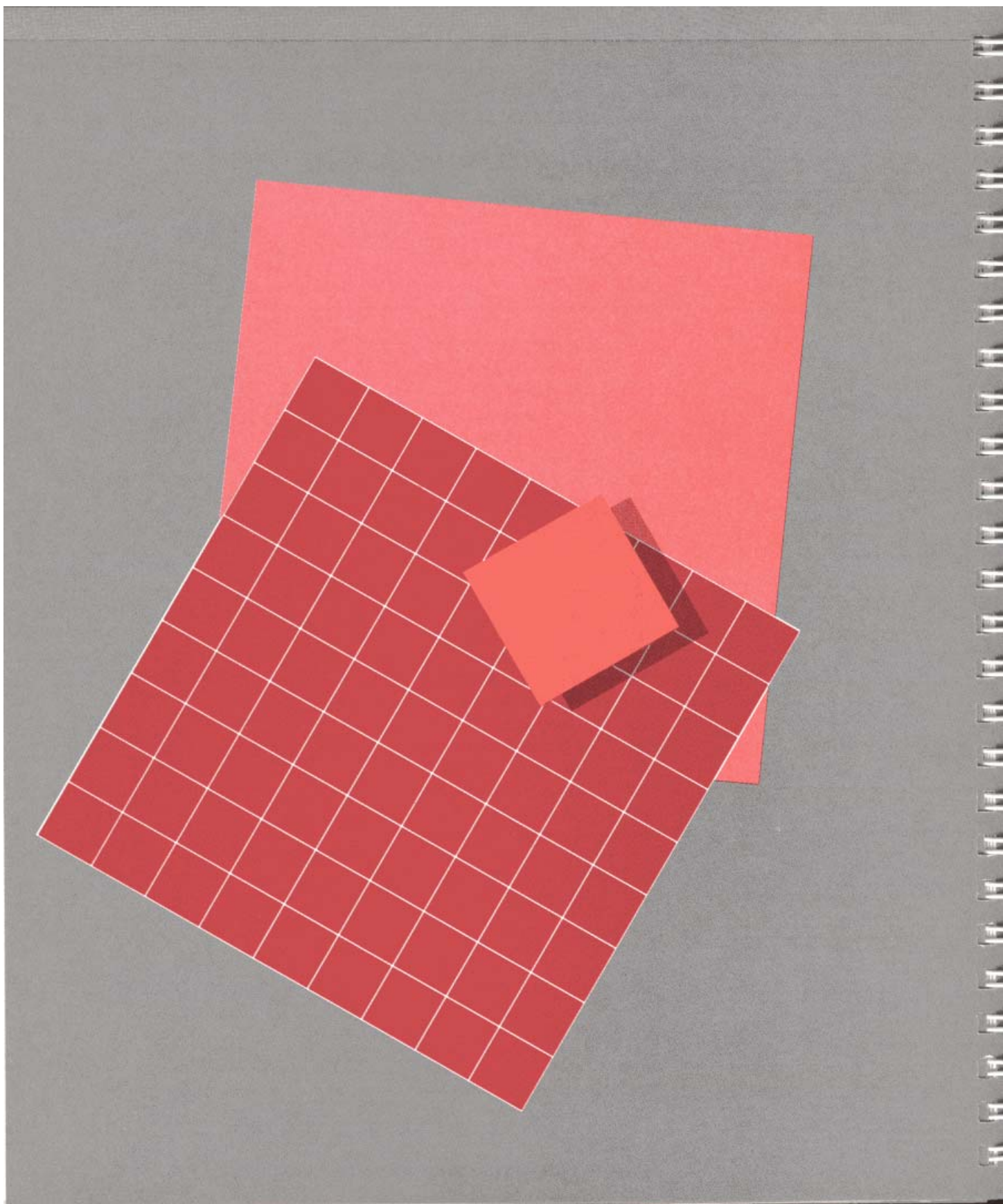
Volume name: The name of a volume. There are rules for naming a volume: must begin with a slash, then a letter; must be made up of letters, numbers, or periods; must not have any spaces or punctuation characters other than periods; must not exceed 15 characters (not counting the slash).

Slot: Short for *expansion slot*. One of the narrow connectors inside the Apple II where you plug in controller cards for disk drives, printers, and other peripheral devices.

Drive: Short for *disk drive*. An external storage device that retrieves information from and stores information on disks.

Using the Volume Commands

-
- 33 Format a Volume
 - 34 Here's How
 - 36 Copy a Volume
 - 37 Here's How
 - 40 List Volumes
 - 40 Here's How
 - 41 Rename a Volume
 - 42 Here's How
 - 43 Detect Bad Blocks
 - 44 Here's How
 - 45 Block Allocation
 - 46 Here's How
 - 47 Compare Volumes
 - 48 Here's How
 - 49 Summary of Chapter 3
 - 49 Format a Volume
 - 50 Copy a Volume
 - 51 List Volumes
 - 51 Rename a Volume
 - 51 Detect Bad Blocks
 - 52 Block Allocation
 - 52 Compare Volumes



Using the Volume Commands

See Chapter 2 for an explanation of the terminology associated with the volume commands.

Volume commands affect the entire disk. This chapter explains how to use each of the volume commands in the ProDOS Filer.

Figure 3-1. The Volume Commands Menu of the ProDOS Filer

```

*****
*
*                               VOLUME COMMANDS
*
*****

? - TUTOR
F - FORMAT A VOLUME
C - COPY A VOLUME
L - LIST VOLUMES
R - RENAME A VOLUME
D - DETECT BAD BLOCKS
B - BLOCK ALLOCATION
K - COMPARE VOLUMES

SELECT AN OPTION OR <ESC>: ❖

```

A **block** is a unit of information 512 bytes long. A **byte** is a basic unit of your computer's memory, equivalent to eight bits. A **bit** is the smallest amount of information that a computer can hold.

Here's a list of the commands (or options) on the Volume Commands Menu, shown in Figure 3-1, and a short description of what they're used for:

Command	What It Does
Tutor	Explains the terms used in volume command displays, and tells how to type entries on the display.
Format a Volume	Prepares volumes for use by dividing the recording surface into sections, called blocks , for information storage and retrieval. You can name volumes as you format them or accept default names in the form /BLANKXX (where XX is a number between 00 and 99).
Copy a Volume	Lets you create an exact duplicate of another volume. You can give the copy a new name or give it the same name as the original by accepting the default.
List Volumes	Lets you find out which of your disk drives and/or slots contain ProDOS-formatted disks, and what slot, drive, and volume name corresponds to each disk.
Rename a Volume	Lets you rename a volume without changing its contents.
Detect Bad Blocks	Makes it possible to scan a volume for damaged blocks, which could cause you to lose access to your data.
Block Allocation	Lets you see the total number of blocks on a volume, how many blocks are used, and how many are available.
Compare Volumes	Lets you compare two volumes to see if they are exact copies of each other.

In the following pages, each of these commands will be examined in detail.

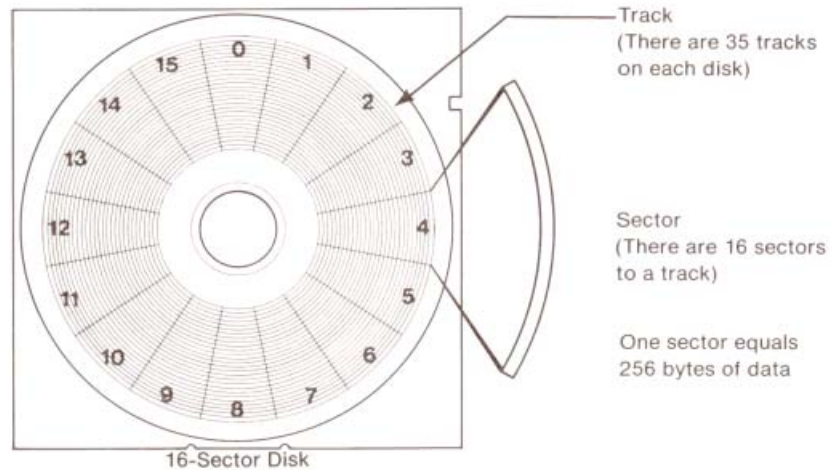
Format a Volume

When you want to put words on a blank piece of paper, you can do so with a typewriter, a pen, or a pencil. There's nothing special you have to do to the paper to prepare it to receive your prose.

When you **format** a disk you prepare it to receive information by dividing the surface into tracks and sectors. Formatting is also called **initializing**.

The same is not true of blank disks. Disks must be **formatted** before they can store your words away. When a volume of magnetic mass storage, like a disk, is manufactured, it contains no information at all. Formatting the volume prepares it for use by dividing its recording surface into standard size **blocks** where information can later be stored.

Figure 3-2. A Formatted Disk



Although you must format blank disks before you can use them to store information, disks don't have to be blank to be formatted. It is important, however, that the information on the used disk be expendable; when you reformat a disk, everything that was stored on the disk becomes inaccessible.

The **destination** disk is the disk that will receive the new information.

The only time you don't have to format a blank disk before you use it to store information is when you use the Copy a Volume command. This command formats **destination** disks before copying information onto them.

You might wonder why disks aren't formatted for you at the time they are manufactured. The reason is that different computers require different kinds of formatting.

Here's How

1. Get to the Filer Menu. (If your computer is turned off, put the *ProDOS User's Disk* in drive 1, turn on the power, and press **F** when you see the Main Menu. If you're still staring at the list of slot assignments, press **RETURN** to return to the Main Menu and press **F** for **PRODOS FILER**.)
2. Press **V** (for **VOLUME COMMANDS**) from the Filer Menu.

Note: If you don't see the Volume Commands Menu, you probably typed one of the other letters from the Filer Menu. You can get back to the Filer Menu and try again by pressing **ESC**.

If nothing at all happened and you're still staring at the Filer Menu, you probably typed a letter that wasn't on your list of options. In which case, you've discovered that the Filer is very forgiving. If you make a typing mistake and it isn't one of your options, you'll get a beep, but you won't see smoke, hear sirens, or get whisked away to never-never land. You'll get another chance, and another, and another. Like a discreet butler, the Filer overlooks innocent blunders.

3. Press **F** (for **FORMAT A VOLUME**) from the Volume Commands Menu. You'll see the display shown in Figure 3-3.

Figure 3-3. Format a Volume Display

```
*****
*                                     *
*                FORMAT A VOLUME      *
*                                     *
*****

-FORMAT-
  THE VOLUME IN SLOT: (6)
                DRIVE:

NEW VOLUME NAME:

-PRESS <RET> TO ACCEPT: <ESC> TO EXIT-
```

The **cursor** marks where the next character you type will appear on the display.

A **default** is the program's best guess as to what you will do next.

Note: The display in front of you isn't a menu, it's a series of statements. The flashing square over the number 6 on the upper right of your display is a **cursor**. The numbers in parentheses are defaults (the programmer's calculated guess as to how you'll complete the statement.) They're called **defaults** because they'll be used by default unless you supply an alternative. You can accept the default by pressing **(RETURN)**. You can change the default by typing over it.

4. Put the volume you want to format (the blank disk, or a disk that contains information you no longer want) in any available disk drive. (If you have a one-drive system, take out the *User's Disk* and replace it with the volume you want to format.)
5. Fill in the slot number of the volume you want to format. (Remember: To accept the default value between parentheses, press **(RETURN)**. To alter the given value, type over it. In either case, the cursor will jump down to the next line.)
6. If you're asked to supply a drive number, type it in or accept the default by pressing **(RETURN)**. (You won't be asked for a drive number if you're formatting a ProFile disk.)
7. Now you can either type a name for the volume you're formatting, or you can accept the default. The default name is /BLANKXX (where XX is a number between 00 and 99).

Note: If you're formatting several disks at one time, and use the default name (BLANKXX), you'll notice that the disks are numbered in sequence: BLANK23, BLANK24, BLANK25, and so on. This feature lets you format a whole box of blank disks in a few minutes without having to worry about giving them each a unique name. You can change the volume names of any of these disks, at any time, with the Rename a Volume command discussed later in this chapter.

If you want to accept the default, press **(RETURN)**. If you have some other name in mind, type over the default. The slash is provided for you, but remember the name you choose must begin with a letter, and can have no more than 15 characters made up of letters, numbers, or periods. To restore the default, press **(ESC)**. Press **(RETURN)** when you've finished typing the name.

If you are formatting a previously formatted disk, you'll get a message like this:

```
DESTROY 'XXX'? (Y/N)
```

(where XXX is the name of the disk you're about to reformat)

If you knew you were about to format XXX, and are willing to lose all the information stored on that disk, press **(Y)** (for *yes*) and the formatting will begin.

If you don't want to format XXX, press **(N)** (for *no*) and there's no damage done. Press **(ESC)** to return to the Volume Commands Menu.

If the volume you are formatting has never been formatted before, and all is well, you'll see this message:

```
FORMAT COMPLETE
```

Remove the formatted disk and label it with its new volume name. If the label is already on the disk, be sure you use a felt-tip pen. A pencil or ball-point pen can damage the surface of the disk.

ProFile Note: If you try to format a mass storage disk like the one in your ProFile, you'll get this message:

```
WARNING: YOU ARE ABOUT TO FORMAT A LARGE DISK  
If it was a mistake, press (ESC).
```

If you knew you were about to format a large disk, press **(RETURN)**. However, it's unlikely that you'd want to format your ProFile disk more than once, so think twice before you press **(RETURN)**. Remember, formatting makes all the information on the disk inaccessible.

If something goes wrong during the format procedure, you'll see an error message. If you're not sure what the message means or what to do about it, consult Appendix A.

Copy a Volume

By the time you finish reading this manual, your *User's Disk* will be as familiar to you as an old shoe. Unfortunately, it's more fragile than an old shoe, and if your dog decides to fetch it for you one morning, goodbye *User's Disk*. For this reason, and a hundred more, it's a good idea to make backup copies of all your important disks.

If you believe in auto insurance for your car and fire insurance for your home, you should also believe in making copies of important disks. There is no other insurance against disk disaster.

Maybe you don't use your disks for coffee coasters. Maybe your children won't mistake a disk for a frisbee. Maybe you won't bring work home from the office on a disk and leave it basking on the dashboard all weekend. Maybe you'll be lucky.

The trouble is, it's so easy to make copies of your disks that no one will sympathize with you if disaster strikes. Don't take chances. Take out some volume insurance. You can practice using this important command right now by making a copy of your ProDOS disk.

ProFile Note: You won't use this command to make copies of your ProFile disk (unless you have two ProFiles). You can use the Copy a Volume command only to make copies of like volumes. Use the Copy Files command to duplicate data stored on your ProFile to flexible disks.

Here's How

1. Get to the Filer Menu.
2. Press **[V]** (for VOLUME COMMANDS) from the Filer Menu.
3. Press **[C]** (for COPY A VOLUME) from the Volume Commands Menu. You'll see the display shown in Figure 3-4.

Figure 3-4. Copy a Volume Display

```
*****
*
*                COPY A VOLUME                *
*
*****

-COPY-
  THE VOLUME IN SLOT: (6)
                   DRIVE:

  TO VOLUME IN SLOT:
                   DRIVE:

  NEW VOLUME NAME:

-PRESS <RET> TO ACCEPT: <ESC> TO EXIT-
```

The **source volume** is the original.

The **destination volume** is the copy.

A **write-enable notch** is a square cutout on the edge of the disk's jacket that allows information to be written on the disk.

A **write-protect tab** is a small adhesive sticker, usually silver, used to cover the write-enable notch so that the information on the disk cannot be altered.

See "The Filer Displays" at the opening of Part II for information on editing the default.

4. Type the slot number of the **source volume** or accept the default by pressing `(RETURN)`. In either case, the cursor will jump down to the next line on the display.
5. Type the source drive number (if asked) or accept the default by pressing `(RETURN)`. Again, the cursor will jump down to the next line on the display.

6. Type the slot number of the **destination volume** or accept the default by pressing `(RETURN)`. The cursor will jump to the next line on the display.
7. Type the destination drive number (if asked) or accept the default by pressing `(RETURN)`. You'll see this message:

```
INSERT DISKS AND PRESS <RET>
```

8. Put the source (original) and destination (new) disks in the appropriate drives and press `(RETURN)`.

If you have a one-drive system, put your source volume in your disk drive and be ready to do quite a bit of disk swapping. Messages at the bottom of the display will tell you when to insert the source volume and when to insert the destination volume.

By the Way: Some disks don't have **write-enable notches** on the side. That means that you can't change the contents of the disk even if you want to. Other disks have notches that allow you to modify disk contents. When you're making copies of important disks it's a good idea to cover the notch on your source disk with a silver **write-protect tab**, just in case you get your source and destination disks confused during the copy process.

9. Type the name you want to give the destination volume. The default is the name of the source volume. You can accept the default by pressing `(RETURN)`, or you can type a new name over the default. Remember, your volume name must begin with a letter (the opening slash is provided for you; if you try to type it, you'll get a beep) and can be no longer than 15 characters—made up of letters, numbers, or periods. Press `(RETURN)` when you finish entering your new volume name.

You'll hear some whirring and see `FORMATTING`, `READING`, `WRITING` as the disk is copied.

Note: When you see `READING` and `WRITING` on the display, the Filer reads information from the source disk into the memory of the Apple II computer and then writes it out to the destination disk, creating an exact copy.

If the destination volume wasn't blank, you'll see this message:

```
DESTROY 'XXX'? (Y/N)
```

(Where XXX is the actual name of the volume you are about to copy over.)

If you knew you were formatting and copying over that particular volume, you can press **(Y)** (for *yes*) and the copying will proceed. If you put the volume in by mistake, thinking it was a blank disk, you can press **(N)** (for *no*) and no harm will come to the disk. You'll return to the top of the Copy a Volume display. You can return to the Volume Commands Menu by pressing **(ESC)**.

By the Way: You can use the Copy a Volume command to copy non-ProDOS disks. (The default for NEW VOLUME NAME will be NON-PRODOS DISK.) If you are copying a non-ProDOS disk to another non-ProDOS disk, no warning will be given before the disk is copied. You also will not get a warning if you copy a ProDOS disk or a non-ProDOS disk to a DOS 3.2 disk or a blank disk. If you copy a ProDOS disk to a non-ProDOS disk, you will see

```
DESTROY 'NON-PRODOS' DISK? (Y/N)
```

If you want to make the copy, press **(Y)** and the command will proceed. If you put the non-ProDOS disk in by mistake, you can stop the copy by pressing **(N)**.

When it's all over, you'll see this message:

```
COPY COMPLETE
```

Once the copy is complete, the cursor will jump back to the first line on the Copy a Volume display in case you want to make additional copies. If you don't want to make any more copies, press **(ESC)**; you'll return to the Volume Commands Menu.

If there's a problem during the copy operation, you'll see an error message. If you're not sure what it means or what to do about it, consult Appendix A.



Warning

If there is a problem during the copy operation, the volume name of the destination disk will be /PARTIALCOPYDISK. This means it's a bad copy, and you should try to make another copy. If you suspect a problem, use the Detect Bad Blocks or Compare Volumes commands described later in this chapter to find out if your copy was successful.

The **startup slot** is the one that connects the disk drive that holds the startup disk.

The **Monitor program** is a system program that is built into the hardware of the computer.

The List Volumes display tells you which ProDOS volume is in which disk drive connected to which slot number. The list will begin with the **startup slot**. If one of your disk drives is empty or contains an unformatted disk, it won't show up on the list. If one of the disks in your disk drives is not a ProDOS disk or isn't recognizable to ProDOS, the message

```
<NO DIRECTORY>
```

will appear in place of the volume name.

Note: When you turn on your computer, the **Monitor program** looks for information on the disk in drive 1 connected to the highest numbered slot (usually slot 6). If information is found, such as ProDOS, that gets the computer running, that slot then becomes known as the **startup slot**.

Rename a Volume

When a blank disk is first formatted, it is often given a name like /BLANK14. Maybe you name your newly formatted disks more whimsically—after racehorses, say, or country western singers. Nevertheless, there comes a time when it would be more informative if the disk containing your income tax computations were called /TAXES82 instead of /SEABISCUIT or /DOLLYPARTON. This is especially true if you work in an office where others need access to your files.

The Rename a Volume command lets you change the name of a volume without changing any of its contents.

Here's How

1. Get to the Filer Menu.
2. Press **(V)** (for VOLUME COMMANDS) from the Filer Menu.
3. Press **(R)** (for RENAME A VOLUME) from the Volume Commands Menu. You'll see a display like the one shown in Figure 3-6.

Figure 3-6. Rename a Volume Display

```
*****  
*                                     *  
*                               RENAME A VOLUME                               *  
*                                     *  
*****  
  
--RENAME--  
  THE VOLUME IN SLOT: (6)  
    DRIVE:  
  
  
NEW VOLUME NAME:  
  
  
  
  
  
  
  
  
  
--PRESS <RET> TO ACCEPT;<ESC> TO EXIT--
```

ProFile Note: If you're renaming a ProFile disk, skip to step 5.

4. Put the volume you want to rename in any available disk drive. (If you have a one-drive system, take out the *User's Disk* and replace it with the volume you want to rename.)
5. Fill in the slot number of the volume you want to rename.
6. If prompted, fill in the drive number of the volume you want to rename. (You won't be asked for a drive number if you're renaming a ProFile volume.)
7. Now you can either type in a new name for the volume, or you can edit the default name, which is the old name of the volume. To edit the name, press `[←]` to accept letters and then type over the letters you want to change. Press `[RETURN]` to indicate that you've finished typing or editing the new volume name. (If you edit the default name, make sure that the cursor is at the end of the name before you press `[RETURN]`; otherwise the character under the cursor and all characters to the right of the cursor will be left off the new volume name.)

If all goes well, you'll see this message:

```
RENAME COMPLETE
```

If there's a problem, you'll see an error message. If you're not sure what the message means, or what to do about it, consult Appendix A.

Detect Bad Blocks

Sometimes it's obvious when a disk is on its last legs—you crease it trying to force it into your disk drive or you set it down in the overflow from a plant you've just watered. But sometimes volumes go bad in more subtle ways. They develop bad blocks. Bad blocks can be caused by such things as fingerprints or dust on the disk. If you suspect that a volume has been damaged (unusual clicking noises coming from the disk drive is a good clue), or if you're having trouble accessing information on a disk, use the Detect Bad Blocks command.

Here's How

1. Get to the Filer Menu.
2. Press **(V)** (for VOLUME COMMANDS) from the Filer Menu.
3. Press **(D)** (for DETECT BAD BLOCKS) from the Volume Commands Menu. You'll see the display shown in Figure 3-7.

Figure 3-7. Detect Bad Blocks Display

```
*****
*
*          DETECT BAD BLOCKS          *
*
*
*****

--DETECT BAD BLOCKS--
  FOR VOLUME IN SLOT: (6)
                DRIVE:

--PRESS <RET> TO ACCEPT:<ESC> TO EXIT--
```

ProFile Note: If you're checking for bad blocks on a ProFile disk, skip to step 5.

4. Put the volume you want to check in any available drive. If you have a one-drive system, take out the *User's Disk* and replace it with the volume you want to check.
5. Fill in the slot number of the volume you want to check.
6. If prompted, fill in the drive number of the volume you want to check. (You won't be asked for a drive number if you're checking for bad blocks on a ProFile.)

If all goes well, you'll see this message:

```
0 BAD BLOCKS
```

If bad blocks are detected, you'll get a message like this:

```
BAD BLOCK NUMBER :  
XXXX  
XXXX  
XXXX
```

(where XXXX stands for the numbers of the bad blocks). If you want to check other volumes for bad blocks, press **(RETURN)**. If not, press **(ESC)**.

By the Way: You can also use this command to detect bad blocks on DOS 3.3 disks.

If the volume you're checking has bad blocks, copy all the files to another disk (using the Copy Files command explained in Chapter 5). When you come to the files with bad blocks, you'll get the message **I/O ERROR**. Once you've salvaged all the files you can, format the disk that has the bad blocks.

Note: You can get a printed list of the bad blocks by changing the output device from the video monitor to your printer. This procedure is explained in Chapter 6, "Configuration Defaults."

If you get an error message and you're not sure what the message means or what to do about it, consult Appendix A.

Block Allocation

This command lets you find out how many blocks on a given volume are taken up with files, how many are available, and the total number of blocks on the volume.

If you're copying files to a disk, it's nice to know in advance whether there's room on that disk to hold them. This command gives you that information.

If all goes well, you should see a display similar to Figure 3-9 indicating the number of blocks used, the number of blocks free, and the total number of blocks on the volume:

Figure 3-9. Completed Block Allocation Display

```
*****  
*                                     *  
*                BLOCK ALLOCATION      *  
*                                     *  
*****  
  
-BLOCK ALLOCATION-  
  FOR VOLUME IN SLOT: (6)  
    DRIVE: 1  
  
      170 BLOCKS USED  
      110 BLOCKS FREE  
      280 BLOCKS TOTAL  
  
-PRESS <RET> TO ACCEPT; <ESC> TO EXIT-
```

If you don't see this display, you probably got a beep and an error message. If you're not sure what the message means, or what to do about it, consult Appendix A.

Compare Volumes

As the name suggests, you can use this command to compare two volumes for blocks that don't match. It's a handy command to use if you get disks mixed up and want to make sure a copy is really identical to the original.

Once you've completed the statements on the Compare Volumes display, the Filer does a byte-by-byte comparison of the two volumes. If the disks aren't exact images of each other, the numbers of any mismatching blocks are shown on the display.

ProFile Note: This command won't do you much good unless you have two Profiles with identical contents, and that's very unlikely.

Here's How

1. Get to the Filer Menu.
2. Press **(V)** (for VOLUME COMMANDS) from the Filer Menu.
3. Press **(K)** (for COMPARE VOLUMES) from the Volume Commands Menu. You'll see a display similar to Figure 3-10.

Figure 3-10. Compare Volumes Display

```
*****
*
*          COMPARE VOLUMES          *
*
*****

-COMPARE-
  THE VOLUME IN SLOT: (6)
                    DRIVE:

  TO VOLUME IN SLOT:
                    DRIVE:

-PRESS <RET> TO ACCEPT; <ESC> TO EXIT-
```

4. Type in the slot number of the first volume.
5. If prompted, fill in the drive number of the first volume.
6. Type in the slot number of the second volume.
7. If prompted, fill in the drive number of the second volume.

You'll see this message:

```
INSERT DISKS AND PRESS <RET>
```

8. Put the volumes in the drives you specified and press **(RETURN)**. (If you have a one-drive system, put one volume in your disk drive. There will be prompts on the display telling you when to take out your first volume and put in your second.)

By the Way: You can also use this command to compare two DOS 3.3 disks.

If all goes well, you'll get the following message:

```
COMPARE COMPLETE
```

If any of the blocks failed to match, you'll get a message like this:

```
BLOCK NUMBERS DO NOT MATCH:
```

```
0  
2  
6
```

```
-PRESS <RET> TO CONTINUE: <ESC> TO EXIT-
```

Only three of the mismatching blocks are shown initially. If you press **RETURN**, the numbers of the remaining mismatching blocks will be displayed.

If the only mismatching block is number 2, the volumes are exactly the same except for their names or some other information in the volume directory. If block number 6 is displayed, it means that the maps for the disks are not the same. In other words, the copy procedure did not work.

By the Way: You can get a printed list of mismatching blocks by changing the output device from the video monitor to your printer. This procedure is explained in Chapter 6, "Configuration Defaults."

If you got an error message and you're not sure what the message means or what to do about it, consult Appendix A.

Summary of Chapter 3

Format a Volume

The Format a Volume Command prepares disks to receive information.

You supply

- the slot number of the disk to be formatted.
- the drive number of the disk to be formatted (if required).
- the volume name of the disk to be formatted.

If there is already a directory on the disk to be formatted, you'll be asked to confirm your intention to reformat the disk.

If you're about to format the disk of a mass storage device, such as a ProFile, you'll be asked to confirm the operation.

Copy a Volume

The Copy a Volume command copies the contents of one volume to another volume of the same type and size.

You supply

- the slot number of the source (original) volume.
- the drive number of the source volume (if required).
- the slot number of the destination (copy) volume.
- the drive number of the destination volume (if required).
- the volume name of the destination volume.

This command copies disks block by block. Because of this, you can copy DOS 3.3 or ProDOS disks. The command starts by formatting the destination disk with 16 sectors. The copy can be either DOS or ProDOS depending upon the nature of the original disk.

The Filer checks for a ProDOS directory. If the source is a ProDOS disk and the destination is non-ProDOS, you'll get the message

```
DESTROY 'NON-PRODOS' DISK (Y/N)?
```

If you want to make the copy, press **(Y)** and the command will proceed. If you put the non-ProDOS disk in by mistake, you can stop the copy by pressing **(N)**.

When you are copying a non-ProDOS disk, the default for `NEW VOLUME NAME` will be `NON-PRODOS DISK`. When you copy a non-ProDOS disk to another non-ProDOS disk, no warning will be given before the disk is copied. You also will not get a warning if you copy a ProDOS disk or a non-ProDOS disk to a DOS 3.2 disk or a blank disk.

If you're copying with one drive, you'll get prompts telling you when to alternate the source and destination disks.

I/O stands for *input/output*, which refers to the transfer of information into and out of a computer.

In the event of an **I/O** error, the destination disk receives a temporary volume name to alert you that the copy procedure was not successful.

List Volumes

The List Volumes command lists the slot number, drive number (if applicable), and the names of all ProDOS volumes in disk drives connected to your Apple II.

The volumes are listed in the order in which they are encountered, with the startup slot and drive first. If one of the volumes isn't a ProDOS disk, the message `NO DIRECTORY` will be displayed.

Rename a Volume

The Rename a Volume command renames the volume you specify.

You supply

- the slot number of the volume to be renamed.
- the drive number of the volume to be renamed (if required).
- the new volume name.

Once you supply this information, the volume name is changed.

Detect Bad Blocks

The Detect Bad Blocks command performs a validity check by searching for damaged blocks.

You supply

- the slot number of the volume you want to check.
- the drive number of the volume you want to check (if required).

Once you supply this information, the volume is searched. If a bad block is detected, a message is displayed indicating the number(s) of the bad block(s). If there are no bad blocks you'll get the message:

```
0 BAD BLOCKS.
```

You can also use this command to detect bad blocks on DOS 3.3 volumes.

Block Allocation

The Block Allocation command tells you how many blocks are used, how many blocks are available, and the total number of blocks on a specified volume.

You supply

- the slot number of the volume you want to check.
- the drive number of the volume you want to check.

Once you supply this information, the disk is checked to see how many blocks are used, how many are free, and the total number of blocks on the volume.

Compare Volumes

The Compare Volumes command does a byte-by-byte comparison of any two volumes of the same type and size.

You supply

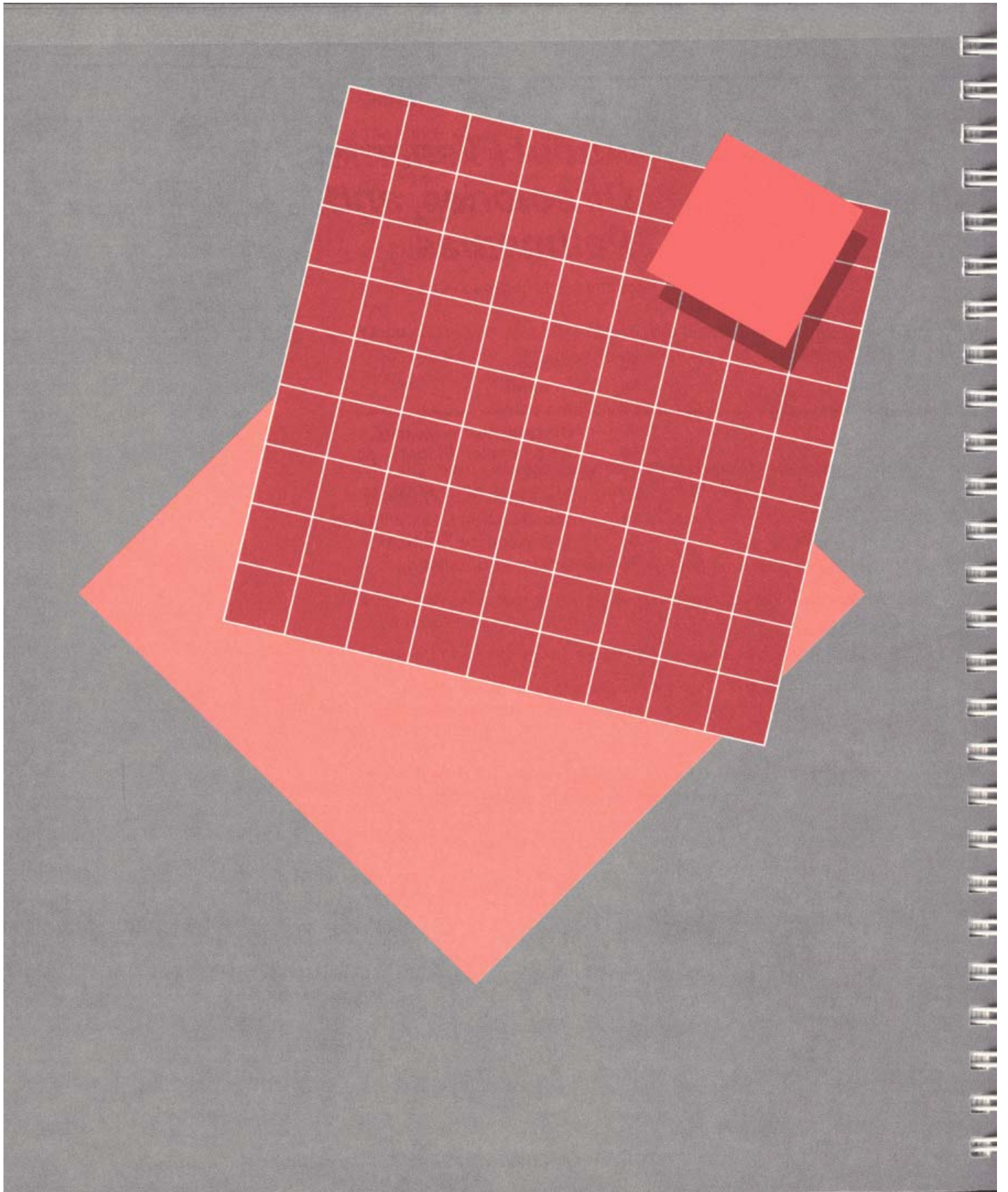
- the slot number of the primary volume you want compared.
- the drive number of the primary volume you want compared (if required).
- the slot number of the secondary volume you want compared.
- the drive number of the secondary volume you want compared (if required).

The disks in the two volumes are accessed for a byte-by-byte comparison. If the volumes aren't exact images of each other, the mismatching block numbers will be displayed.

You can also use this command to compare two DOS 3.3 volumes.

About Filenames, Directories, and Pathnames

- 55 Filenames
- 56 Directories
- 58 Pathnames
- 59 Prefixes
- 59 An Example: Widgets, Inc.
- 63 Wildcards
- 63 The Equal Sign Wildcard
- 64 The Question Mark Wildcard
- 64 A Wildcard Example
- 67 Summary of Chapter 4



About Filenames, Directories, and Pathnames

Chapter 5 explains how to use each of the file commands.

One of the five options on the Filer Menu is File Commands. This group of commands works with individual files on a disk. This chapter provides background information on files, directories, pathnames, and prefixes—the stuff of which file commands are made.

A file is an orderly collection of information on a disk. It can be a program, a poem, or a recipe for potato salad.

All files have names and when you want access to a file, you refer to it by its name.

Filenames

The computer doesn't name files—you do. But there are a few ground rules. Filenames can have no more than 15 characters. The first character must be a letter, but after that it doesn't matter if the characters are letters, numbers, or periods. (Filenames can't have spaces or punctuation marks other than periods.)

Here are a few acceptable filenames:

POTATOSALAD
SILLYPROGRAM
BIZ.LETTER
MEMO.1.APRIL.84

Here are a few unacceptable filenames:

POTATO SALAD	Spaces can't be included in a filename.
RIDICULOUSPROGRAM	Too long. The limit is 15 characters.
EMILY'SPOEM	Apostrophes aren't legal characters.
MEMO: BILL	Colons and spaces are illegal characters.

If you forget these rules and try to leave a blank in the middle of a filename, the Filer helps you by refusing to print the invalid character. Each time you type a character, the Filer checks it. If it's invalid, you'll hear a beep and notice that the cursor won't budge until you type a character that is acceptable.

Directories

When a volume is first formatted, it gets a name and a directory. Anything you save on that volume is accessed through that directory.

A directory is also a file. But unlike a file that holds a recipe for potato salad, a directory file keeps track of where all the other files on the disk are stored. You could say that a directory file is like an address book.

If you want to see the contents of the volume, you can do so at any time by using the List ProDOS Directory command, which you'll learn about in the next chapter.

It seems very convenient. But after a period of time, one volume can accumulate a staggering array of unrelated files: a recipe for guacamole, a Christmas card mailing list, your tax return, a recipe for clam chowder, your child's Little League roster, a financial forecast for a critical work project, and so on.

There comes a time—especially with a mass storage disk—when it would be convenient to have a way of organizing your files so you don't have to sift through guacamole and chowder to find your financial forecasts.

Subdirectories allow you to group your files logically. A subdirectory is just like a directory except that it is subordinate to the directory. And just as you can use the List ProDOS Directory command to list the volume directory, you can use the command to list a subdirectory.

You'll learn how to create subdirectories by using the Make Directory command described in the next chapter.

By the Way: All subdirectories are directories, but not all directories are subdirectories. Each volume has only one volume directory; the rest are subdirectories.

Figure 4-1. Organizing a Disk With Subdirectories

Here's an example of how you might organize a disk with subdirectories:



If you were to look in YOUR.DIRECTORY, you would only find the five subdirectories listed. YOUR.DIRECTORY doesn't know what is in RECIPES or KIDS. The directory is like the boss at a company who has five people reporting to him. He delegates responsibility to those five people for the people who work for them. The volume directory keeps track of all the files on its level. When those files are also subdirectories, the subdirectories keep track of the files that are under them.

In other words, YOUR.DIRECTORY knows the addresses for RECIPES, XMAS.LIST, KIDS, and HI.FINANCE, while RECIPES knows the addresses for GUACAMOLE and CHOWDER. This kind of arrangement is called a hierarchical file structure because files are organized into successive levels.

You're not required to use a hierarchical file structure with directories within directories within directories. There's nothing to stop you from lumping everything into one big volume directory. But don't make your decision just yet. This chapter presents some examples that will show how you can use directories to build a storage system whose structure reflects the relationship between the pieces of information you're storing. If the system makes sense, use it. If it seems like more trouble than it's worth, you can go back to lumping all your files into one directory.

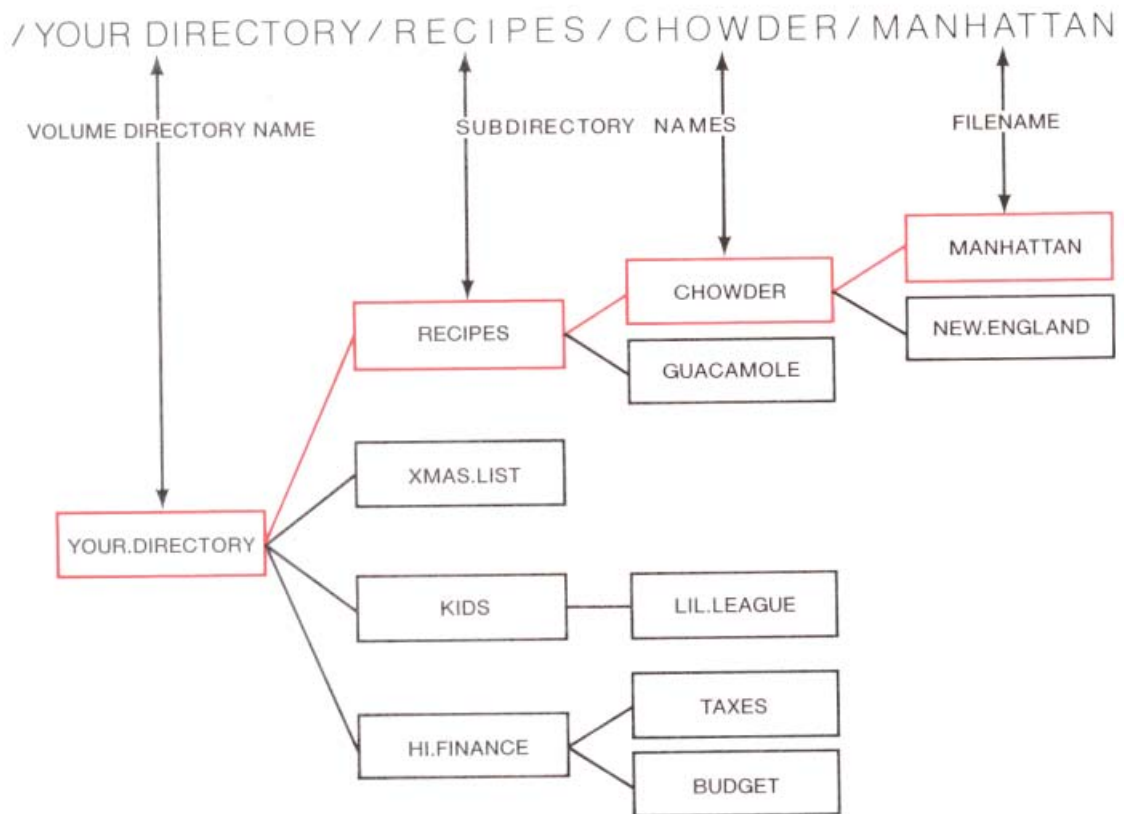
Pathnames

A **delimiter** is used to mark the beginning and end of a sequence of characters. In written English, the space is used as a delimiter between words.

A pathname is the volume directory name, followed by any number of subdirectory names, followed by the filename. The entire pathname is preceded by a slash and each name within the pathname is separated by a slash. The slash is called a **delimiter**.

In other words, the pathname tells ProDOS what *path* to take to get to a particular file. First it reads the volume directory name, then it looks in the volume directory for the location of the next directory, then it looks in that directory for the location of the file. You can have as many subdirectories as you want as long as the total number of characters in the pathname doesn't exceed 64.

Figure 4-2. Pathnames



Note: If you don't use subdirectories to organize your files, your pathnames will have only two components: the volume directory name and the filename. They're still pathnames, just very short ones.

Pathnames are a great aid to organization, but they're tedious to type. Wouldn't it be nice if there were some way to abbreviate pathnames? There is: the prefix.

Prefixes

A **prefix** is just what it sounds like. It is part of a pathname that is added to the beginning of what you type.

You'll learn how to set a prefix in Chapter 5.

Figure 4-3. Using Prefixes. Once the prefix is set, type the rest of the pathname (the partial pathname) to get to the file the full pathname specifies.

It can be inconvenient to have to specify a complete pathname every time you want to locate a file. For example, if you wanted to look at all the chowder recipes you have, it would be nice not to have to type an entire 41-character pathname for each recipe. ProDOS saves the day, and the fingers, by means of a stored pathname called a **prefix**.

A prefix is a pathname of a directory or subdirectory; it is placed in front of a filename to form the file's full pathname. As you can see in Figure 4-3, the length and content of prefixes can vary.

Prefix	+	Partial Pathname	=	Full Pathname
/YOUR.DIRECTORY/		RECIPES/CHOWDER		/YOUR.DIRECTORY/RECIPES/CHOWDER
/YOUR.DIRECTORY/RECIPES/		CHOWDER		/YOUR.DIRECTORY/RECIPES/CHOWDER

An Example: Widgets, Inc.

Widgets, Inc., a wholly owned subsidiary of Donothing Gadgets Ltd., has offices in a building on Fourth Street. When you walk into their offices, you see three large filing cabinets against the wall. The left cabinet is named ACCOUNTING, the right cabinet is named PERSONNEL, and the center cabinet is named INVENTORY.

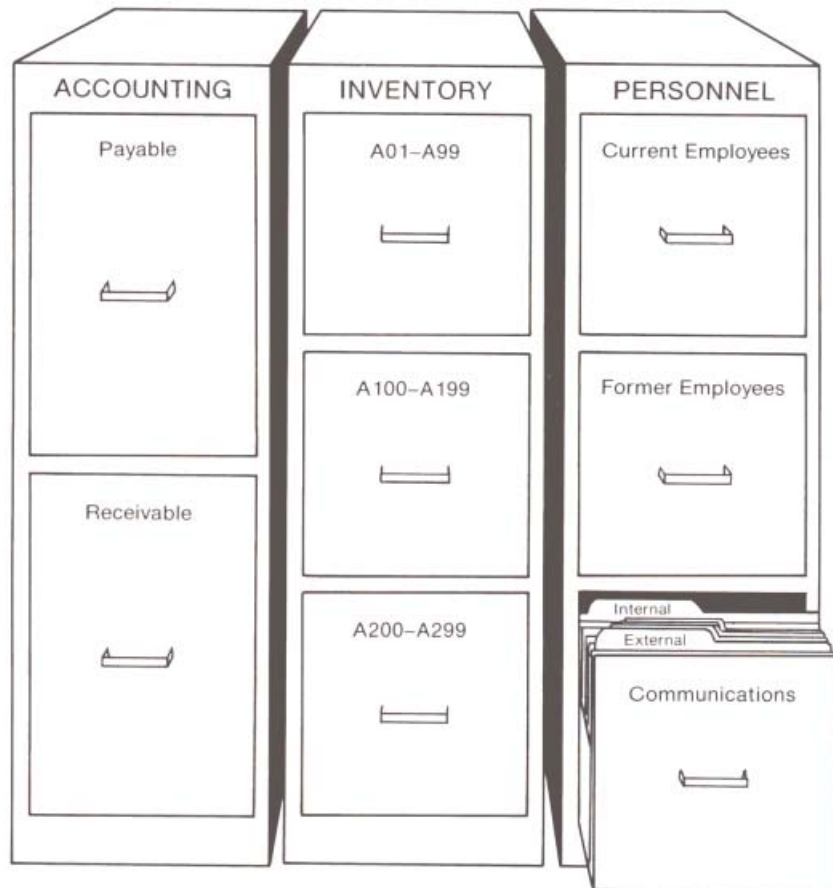
The ACCOUNTING file cabinet has one drawer marked PAYABLE and one drawer marked RECEIVABLE. In each drawer are a lot of file folders, one folder for each company that has an account with Widgets, Inc.

The PERSONNEL file has drawers marked CURRENT EMPLOYEES, FORMER EMPLOYEES, and COMMUNICATIONS. The first two contain individual file folders for each current or former employee. The last drawer has a divider down the middle, separating INTERNAL from EXTERNAL communications. The

INTERNAL side contains file folders for each person who's ever sent a memo, and each folder contains every memo that person has sent. The EXTERNAL side contains all the documentation of Widgets' products, in no particular order.

Figure 4-4 shows the Widgets filing system.

Figure 4-4. The Widgets, Inc. Filing System



You are in charge of maintaining these file cabinets. If somebody wanted to write a letter of recommendation for Sam Johnson and asked you for Sam's personnel file, you would go to the PERSONNEL filing cabinet, look in the FORMER EMPLOYEES drawer, and find the file for Sam Johnson.

If you wanted to find out how much was owed to the Quigley Supply Company, you would go to the ACCOUNTING cabinet and look in the PAYABLE drawer for the Quigley file folder.

If Steve Atkins gave you a copy of a memo he wrote, you would go back to the PERSONNEL cabinet, open the COMMUNICATIONS drawer, look in the INTERNAL side for Steve's file, and drop the memo in with all of the other internal memos Steve has sent.

If you wanted to store all of Widgets' records on the Apple II, you could set up the filing system the same way, only instead of filing cabinets, you'd use disks.

Each disk would take the name of the file cabinet it was replacing. This becomes the volume name of that disk. The disk's volume directory is also identified by this volume name. So, the ACCOUNTING disk's volume directory would contain the files named PAYABLE and RECEIVABLE, which would correspond to the drawers in the filing cabinet. These files would be directories. The PAYABLE directory would contain files named QUIGLEY, ACME, and so forth, which would contain the same information that was previously in the individual file folders.

The structure is the same throughout the rest of the filing system. The PERSONNEL disk contains three directory files: CURRENT, FORMER, and COMMUNICATIONS. The COMMUNICATIONS directory contains directory files called INTERNAL and EXTERNAL. INTERNAL contains directories for ADAMS, SMITH, JOHNSON, ATKINS, and all the rest of the memo senders. Finally, the directory ATKINS contains a file for each memo Steve Atkins has sent.

If you wanted to get that employee file for Sam Johnson, you would tell ProDOS how to locate that file by supplying a pathname. The pathname of Sam Johnson's employee file, which has the filename JOHNSON, is

```
/PERSONNEL/FORMER/JOHNSON
```

That's like saying go to the PERSONNEL cabinet, look in the FORMER employees drawer, and get the JOHNSON file. Only now you're telling ProDOS to go to the PERSONNEL volume, look in the FORMER directory, and get the JOHNSON file.

The pathname of the Quigley accounts payable file is

```
/ACCOUNTING/PAYABLE/QUIGLEY
```

That's like saying go to the ACCOUNTING cabinet, look in the PAYABLE drawer, and get the QUIGLEY file. Or to put it in language that ProDOS can understand: go to the ACCOUNTING volume, look in the PAYABLE directory, and get the QUIGLEY file.

The pathname of the individual memo that Steve Atkins sent on March 15 might be

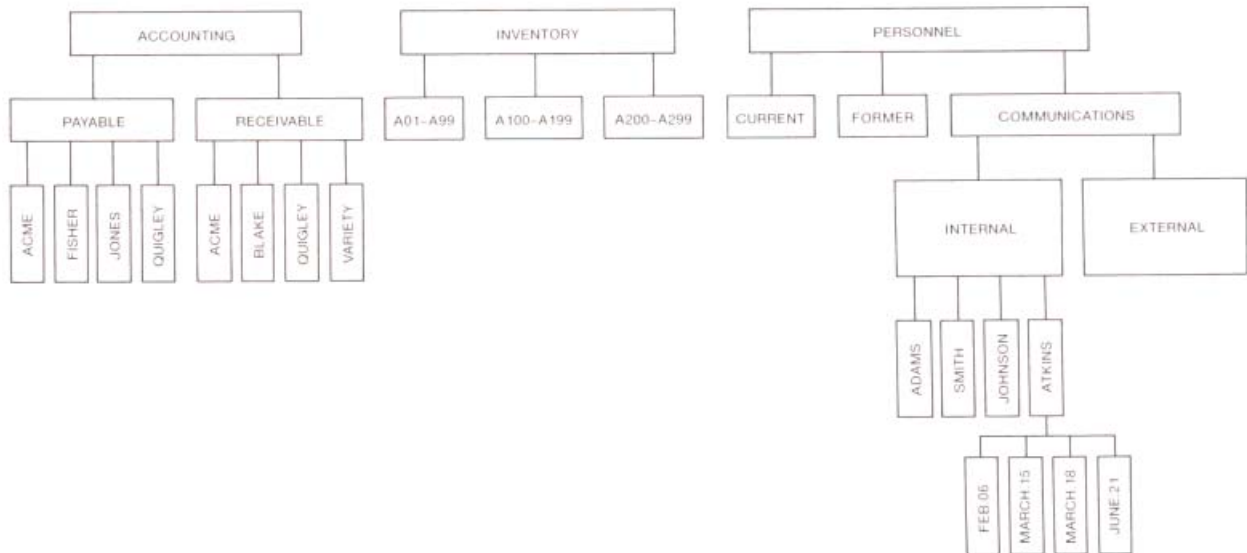
```
/PERSONNEL/COMMUNICATIONS/INTERNAL/ATKINS/MARCH.15
```

That's like saying go to the PERSONNEL cabinet, look in the COMMUNICATIONS drawer on the INTERNAL side, find the ATKINS folder, and get the MARCH 15 memo. Or to put it more precisely ProDOS-wise: go to the PERSONNEL volume, look in the COMMUNICATIONS directory, then look in the INTERNAL directory, then look in the ATKINS directory, and get the MARCH.15 file.

ProDOS doesn't know from cabinets, drawers, folders, or memos, but it can maneuver from directory-to-directory-to-file faster than you can say "COFFEE.BREAK."

Figure 4-5. The Hierarchy of Widgets' Records

If you're having trouble visualizing all of this, take a minute to study the diagram in Figure 4-5.



Each file has one pathname, and each pathname refers to only one file. Even if you have two files with the same local name, you can still tell them apart by pathname. For instance, one could be named

```
/ACCOUNTS/PAYABLE/QUIGLEY
```

The other could be named

```
/ACCOUNTS/RECEIVABLE/QUIGLEY
```

Even though these two files have the same volume name and the same filename, they are different because they have different pathnames.

Don't let the complexity of Widgets' files scare you. You may rarely build hierarchies of such complexity on the Apple II. If you are using flexible disks, you might have different disks for different purposes, and maybe a few directories on each disk, but you'll find that you usually won't need more than one or two levels of directories on any one disk. The important thing is that the capability is there.

Wildcards

A wildcard is a symbol that makes life a whole lot easier when you are using file commands. A wildcard can represent any character or group of characters that you want it to represent—just as the wildcard in a poker game can represent the ace of spades, the king of hearts, or any other card you want it to represent.

You can use wildcards with these file commands: List ProDOS Directory, Copy Files, Rename Files, Delete Files, and Alter Write-Protection.

The Equal Sign Wildcard

One of the two wildcards you can use with file commands is the equal sign (=). When it is used as a part of a pathname, it does strange and wonderful things.

Here's how it works. If you type the filename `FR=D`, the command you happen to be working with at the time selects all the names starting with `FR` and ending with `D`, regardless of the letters in between. The filenames `FRIEND`, `FRAUD`, and `FRESH.SQUID` all share this file pattern (though it's unlikely they'd share the same directory).

The Question Mark Wildcard

The other wildcard is the question mark (`?`). It works just like the equal sign wildcard character except that you get a chance to evaluate each file before it is deleted, copied, or whatever.

Here's how it works. Once you indicate that you want to copy or delete a given file pattern, you'll see the name of the first file that fits the pattern with the cursor positioned to the right of the filename. And you'll see this prompt:

```
FOR EACH FILE, ENTER <Y/N> OR <ESC>
```

If you press `[N]`, the operation won't be performed on that particular file, and you'll see the message:

```
CANCELED
```

If you press `[Y]`, the operation will be performed, and you'll see a message like

```
COPIED
```

or

```
DELETED
```

or whatever is appropriate for the command you're using.

In either case, you'll be asked the same question about each file that fits that pattern.

If you press `[ESC]`, the operation is canceled for the rest of the files.

A Wildcard Example

Here's an example of how you might use wildcards in real life. Say you're writing a cookbook, and you've got a volume filled with recipes. The `/DESSERTS` volume directory looks, conceptually, like this:

Examples of using wildcards are included in Chapter 5: "Copying With a Wildcard" and "Deleting With a Wildcard."


```
/DESSERTS/  
PECAN.PIE  
BANANA.BREAD  
BAKED.ALASKA  
CREPE.SUZETTE  
APPLE.CRISP  
APPLE.PIE  
CHERRY.PIE  
GINGER.BREAD  
APPLE.PANDOWDY  
SUGAR.COOKIE
```

You decide you'd like to copy all the pie recipes into their own subdirectory named PIES. (Before you can copy files into a new directory, you have to create that directory using the Make Directory command. It's easy, and you'll learn about it in Chapter 5.)

If it weren't for the wildcard, you'd have to use the Copy Files command three times: once for APPLE.PIE, once for PECAN.PIE, and once for CHERRY.PIE.

But you can replace CHERRY, PECAN, and APPLE with a wildcard character, as in =PIE, and the Copy Files command will copy every file in the directory ending in *PIE*.

Here's what your volume would look like after the copy files operation:

```
/DESSERTS/  
PECAN.PIE  
BANANA.BREAD  
BAKED.ALASKA  
CREPE.SUZETTE  
APPLE.CRISP  
APPLE.PIE  
CHERRY.PIE  
GINGER.BREAD  
APPLE.PANDOWDY  
SUGAR.COOKIE  
PIES/  
  PECAN.PIE  
  APPLE.PIE  
  CHERRY.PIE
```

But you don't want two copies of your pie recipes on the same volume. So now you can use the Delete Files command with a wildcard and clean up your disk.

Deleting is always riskier than copying, so you'll probably want to use the question mark wildcard—it lets you confirm that you really want to delete the file before doing it. This is a good idea in case a file you don't really want to delete fits the pattern formed with the wildcard.

So this time use `/DESSERTS/?PIE` with the Delete Files command. You will see the name of the first file that fits the pattern

```
/DESSERTS/PECAN.PIE
```

and the message

```
FOR EACH FILE, ENTER <Y/N> OR <ESC>
```

Press `(Y)` because you really do want to delete the file. You will have to confirm the command for each file that fits the pattern.

After deleting the extra files, your volume will look like this:

```
/DESSERTS/  
BANANA.BREAD  
BAKED.ALASKA  
CREPE.SUZETTE  
APPLE.CRISP  
GINGER.BREAD  
APPLE.PANDOWDY  
SUGAR.COOKIE  
PIES/  
  PECAN.PIE  
  APPLE.PIE  
  CHERRY.PIE
```

You can use a wildcard character in the middle of a filename (`FR=D`), at the beginning of a filename (`=PIE`), or at the end of a filename (`APPLE=`).

You can even use the wildcard character to represent all the files in a directory. Let's say the test kitchen needs a copy of every recipe in your dessert directory. All you have to do is type

```
CD /DESSERTS/=
```

and all the files in that directory will be copied.

The only real restriction on your use of wildcards is that you can only use one wildcard and it can only be in the last name (filename) of the pathname. If you try to use more than one wildcard in a pathname, you'll get this error message:

```
ILLEGAL WILDCARD
```

The Filer won't let you type pathnames with illegal wildcards.

Summary of Chapter 4

File: An orderly collection of information on a disk accessed by a filename.

Directory File: A file that contains the names of other files.

Filename: A filename can be the name of a data file or the name of a directory file. Filenames can be up to 15 characters long. The first character must be a letter. The rest of the characters can be letters, numbers, or periods.

Pathname: A pathname is a volume name followed by a series of filenames. The whole pathname is preceded by a slash, and the component names within the pathname are separated by slashes, which act as delimiters. The pathname tells ProDOS what path to take to a given file.

Prefix: A stored pathname (ending with a slash) that specifies part of the pathname. Once you set a prefix, you can refer to any file in the named directory or subdirectory by filename alone.

Wildcard: A character (= or ?) that represents other characters in a pathname. It can be used with the List ProDOS Directory, Copy Files, Rename Files, Delete Files, and Alter Write-Protection file commands.

