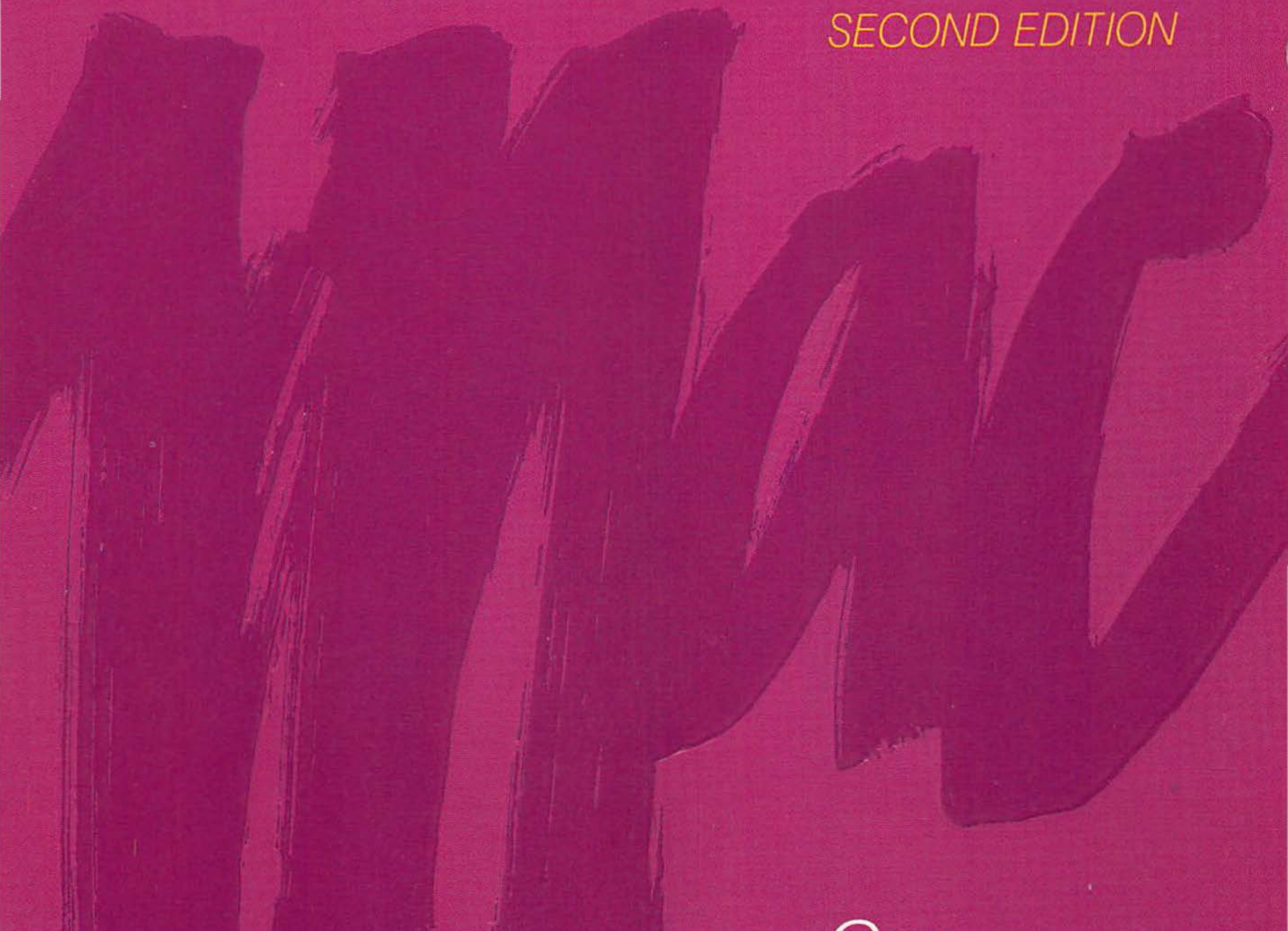
The Apple® Macintosh™ Book

SECOND EDITION



Cary Lu

MICROSOFT®
PRESS

Mac

The Apple® Macintosh™ Book

SECOND EDITION

Cary Lu

MICROSOFT® PRESS

PUBLISHED BY
Microsoft Press
A Division of Microsoft Corporation
10700 Northup Way, Bellevue, Washington 98004

Library of Congress Cataloging in Publication Data
Lu, Cary, 1945–
Mac : The Apple Macintosh Book
Includes Index.
1. Macintosh (Computer) I. Title.
QA76.8.M3L8 1985 004.165 85-21432
ISBN 0-914845-66-7

Printed and bound in the United States of America

1 2 3 4 5 6 7 8 9 RRD RRD 8 9 0 9 8 7 6 5

Distributed to the book trade in the United States by Harper and Row.

Distributed to the book trade in Canada by General Publishing Company, Ltd.

Distributed to the book trade outside the United States and Canada by Penguin Books Ltd.

Penguin Books Ltd., Harmondsworth, Middlesex, England
Penguin Books Australia Ltd., Ringwood, Victoria, Australia
Penguin Books N.Z. Ltd., 182-190 Wairau Road, Auckland 10, New Zealand

British Cataloging in Publication Data available

Disclaimer

The opinions in this book are strictly those of the author and do not necessarily represent the views of Microsoft Corporation, Apple Computer, Inc., or any other company.

Introduction

Section One

Section Two

Section Three

Section Four

Preface to the Second Edition

In the 17 months since the first edition, a flood of new Macintosh products has appeared. To cope with the flood, this second edition has much new and greatly revised material:

- Software and hardware developments in all areas.
- A dot-matrix printer comparison, including speed and printing resolution measurements.
- How the LaserWriter and its PostScript graphics language processor works.
- A chapter on graphic arts, including information on scanners and typography.
- Expanded discussion of programming languages, including a comparative programming example in 16 languages.
- A comparison of the Macintosh design strategy versus the IBM PC-AT.
- Expanded coverage of communications.
- Networks and AppleTalk.
- The Technical Topics chapter, now with wiring information for all ports, internal display adjustments, an annotated list of system error messages, and an explanation of why the Macintosh should not be operated above 15,000 feet—and what to do about it.

The first edition included much introductory material for readers who might not have even seen a Macintosh. To save space in this second edition, I have shortened the introductory portions and eliminated most material that can be found in Apple's manuals.

Although I describe many specific products, my main goals are to explain how and why things work—or don't work—and to discuss future developments. Because the future will bring many improvements, I use stringent standards, standards that will not suffer from inflation. In printer resolution, for example, I consider the Imagewriter in its "high resolution" mode to be *low* resolution (*very low* resolution in standard mode), and the LaserWriter to be *medium* resolution. Everyone should understand that, at present, only typesetting equipment is capable of quality printing; everything else is a compromise. In graphics, as in all areas, computers should make things better, not merely easier.

Cary Lu
July 1985
Boston, Massachusetts

Acknowledgments

This book comes out of a conversation between Bill Gates of Microsoft and Steve Jobs of Apple. Jobs's development team was hard at work on the then-secret Macintosh, which was many months away from introduction. Microsoft was busy writing application programs and helping Apple with the Mac interface and many other design issues. Bill suggested that Microsoft's new publishing division prepare a book as well.

Nahum Stiskin, general manager of Microsoft Press, asked me to write the book shortly afterwards. I replied that I had never heard of any microcomputer that was sufficiently interesting to write a whole book about. Nahum insisted that I go to the west coast and see the Macintosh in action.

Here is the book.

Many people helped in its preparation. Among the staff of Apple Computer, I especially thank Chris Espinosa, Martin Haeberli, Mike Murray, Mike Boich, and Guy Kawasaki. Almost everyone at Microsoft seems to have helped at one time or another. Jeff Harbers, leader of the Macintosh software development group at Microsoft, played a key role, along with others in the group. Despite the extraordinary pressures of completing extremely complex products under an absolute deadline, everyone patiently answered questions and offered suggestions. These people really understand how the Macintosh and its software work; any errors are mine alone.

At Microsoft Press, Joyce Cox and Salley Oberlin contributed outstanding editing; Elaine Foster, Bonnie Dunham, and Marianne Moon put in long hours turning the manuscript into finished form. Art director Karen-Lynne de Robinson, along with Nick Gregoric, Sue Cook, and Chris Stern, turned out the visual portion of this book. I also thank Barry Preppernau and Larry Levitsky.

Fellow author Steve Lambert shared the pleasures and frustrations of working with hardware and software under development and provided much valuable advice and insight.

At a critical time, Ellen Chu came in and read several chapters of the book. She didn't say much; she just threw them out and rewrote them. She was right.

Cary Lu
February 1984
Boston, Massachusetts

For this second edition, virtually everyone who worked on the first edition at Microsoft Press helped out again, and I thank them all again. JoAnne Woodcock edited the second edition, and I also thank David Laraway, Dave Rygmyr, Lee Thomas, Debbie Kem, Stephanie Ideta, John Berry, Lia Matteson, Greg Hickman, Karen Meredith, and Lesley Link-Moore. Mark Alsop contributed many new line drawings.

Cary Lu
July 1985

Introd

Computers Should Work Like People

Computers are supposed to help us get work done quickly, easily, and effectively. So why have they become cloaked in mystique? Because most computers are difficult to use. So difficult, in fact, that we hear about "computer literacy" as if everybody must learn a new language. Computer enthusiasts haven't helped by talking computer jargon that obscures rather than clarifies. And so the mystique has grown: To work with a computer, we must think like a computer.

Nonsense. Computers should work the way we do.

More than any other computer, the Macintosh has changed people's minds about how these machines should work. It is the first widely used computer that does not force us to change our language and work habits for its sake. Learning to use the Macintosh does require some practice, but since most steps are analogous to the way we already work, learning is quick and easy.

The Macintosh is a visual computer, one operated as much with symbols and pictures as with words. The ideas behind it were originally developed at XEROX's Palo Alto Research Center and, in 1981, led to the landmark XEROX Star, the first commercial computer with a visual interface. Apple's Lisa, launched in 1983, brought the price of such machines below $10,000—much less than the Star, but still too high for most individuals and many businesses. Now the Macintosh offers a visual interface for considerably less, enabling many more people to enjoy its benefits.

In fact, the advantages of a visual interface are so compelling that all future microcomputers will have one. Macintosh imitations are already appearing and, by 1986, graphic Mac-style software will be available for the IBM Personal Computer AT. (The PC and PC-XT will also have such programs, but these older machines lack the computing horsepower to run visual software effectively.)

Before the Macintosh, even people who liked computers accepted an initial period of suffering before they became productive or had any fun. Not so with the Mac. Whether you're new to computers or an old hand, Mac's visual technology means those periods of frustration are over.

HOW TO USE THIS BOOK

This book is not intended to be read straight through. Instead, it is divided into four main sections. The brief opening section, Chapters 1 and 2, introduces you to the Mac and describes its basic operations. You can skip these two chapters if you already have a Macintosh.

The second section, Chapters 3 through 10, describes specific types of software. You need not read these chapters in order; just read about the programs that interest you. If you are already using a Mac, you may also want to pass over the introductory sections in some chapters and concentrate on the software comparisons. Although I have kept technical jargon to a minimum in these chapters, some terms are unavoidable. Check the glossary for definitions of any you don't understand, or read Chapter 11 for full explanations.

The third section, beginning with Chapter 11, takes up each part of the Macintosh in detail, explaining how each component works and interacts with the others. This section also describes the inner workings of Mac software. Other topics include graphic arts, maintenance, the Macintosh versus the IBM PC-AT, and future developments.

Section Four is for those interested in specialized topics, such as photography and advanced communications techniques, including AppleTalk.

A glossary and an index complete the book.

In my comments, I list the version number of the product that I looked at. Many products will change in time, adding new features, so some comments may no longer apply. Products that were substantially complete but not yet available for sale when I wrote about them are described as *pre-release.* Products in the planning stage but not necessarily operational are described as "The Acme Company *plans* to release. . . ," or "The Beta company has *announced*. . . ."

Some of the products described here may not be available and others will have been superseded by better products. A few companies mentioned here were technically operating in bankruptcy as I wrote this. Use this book as a guide, but check with magazines and user groups for the latest information.

Sectio

In the Beginning

one

What is the Macintosh? Is it right for you? This section introduces you to Macintosh and tells you where and how to set it up; it reviews Mac's major components and their functions.

Getting Acquainted

Chapter 1

Will you save time using a computer? Maybe. In many situations, the time savings with a computer are surprisingly hard to measure. Often the total amount of work remains much the same, because any potential time savings are lost in redoing and polishing your first effort. When you typed a memo on a typewriter, you probably didn't bother to retype it for small errors; you simply wrote a correction in the margin. With a computer, you will be tempted to edit the memo and reprint it—several times. The results are probably cleaner, but did you save time?

Sometimes a computer is too easy to use. You can fiddle with numbers so effortlessly that their underlying meaning can become obscured. In contrast, if you must work through the numbers by hand, on paper, you may start more carefully, choosing realistic figures for a financial projection.

For some tasks, computers do save time; performing many repetitive calculations or maintaining complex inventory lists, for example, are best done with a computer.

HOW MUCH DO YOU NEED TO LEARN?

Depending on your situation, you may only need to learn a single program—a word processor, perhaps. And you may not even need to learn everything about that program. As you become more comfortable with Mac, you'll probably want to learn more, but set your own pace. Learn only as much as you want, when you need to.

Whatever you do, though, don't just read this or other books; get a Mac and use it.

WHAT YOU NEED

When you buy a Macintosh, you need:

- The Macintosh hardware package.
- Some applications software; for many people, the MacPaint and MacWrite combination will be the best starting package.
- A box of blank disks.

A computer's physical parts—keyboard, screen, printer, wires, and so on—constitute the hardware. You can touch hardware.

Software, or programs, are the instructions you need to turn a collection of hardware into a word processor, a number cruncher, or a game machine. You can't touch software, although you can touch the hardware it is stored on. Hardware is useless without software and vice versa.

You can choose from hundreds of programs, each turning your Macintosh into something different. Some of the Macintosh programs vying for your attention may be exactly what you need.

WRITING YOUR OWN PROGRAMS

If you find programming interesting and want to learn to program the Macintosh, by all means do so. But don't feel that you have to. Most people who use computers will never write a program. After all, excellent programs have already been written, so why duplicate the effort? Good word-processing software can take one skilled person several years to prepare; you may well prefer to do other things with that time.

As microcomputers become more common, chances are that a program meeting your specific needs will appear. But if you have a special requirement for which no program exists, you can write the program or have someone else write it for you. (Chapter 10 discusses programming languages.)

IS MACINTOSH RIGHT FOR YOU?

No computer can be all things to all people, and the Macintosh will not suit everyone. You may not find it suitable if you must have:

- Complete compatibility with another microcomputer model.
- Specific software that runs on another microcomputer but not on Mac.
- Specialized accessories that only work on a different computer.
- A color display screen.

Some of these restrictions will change as more software and accessories appear for the Macintosh.

In addition, all computer designs reflect compromises. If you use a Macintosh, you will have to live with certain design decisions, some of them a little painful (see Chapter 21 for details).

Which Macintosh?

This book focuses on the 512 kilobyte Macintosh, and for most people, this model will be the best choice. Avoid the 128 KB Mac; if you have one, upgrade it. The cost of additional memory is dropping rapidly, and there is no point in suffering from memory limitations. If you need more than 512 KB, you can get your Mac modified (see Chapter 22).

If you have a choice, get double-sided disk drives; their greater storage capacity helps tremendously.

The next section shows you how to set up the Macintosh computer. If you have one already set up and are familiar with basic terms like keyboard, disk drive, and mouse, go to Chapter 2; if you have already used a Macintosh, skip to Section Two.

FIRST STEPS

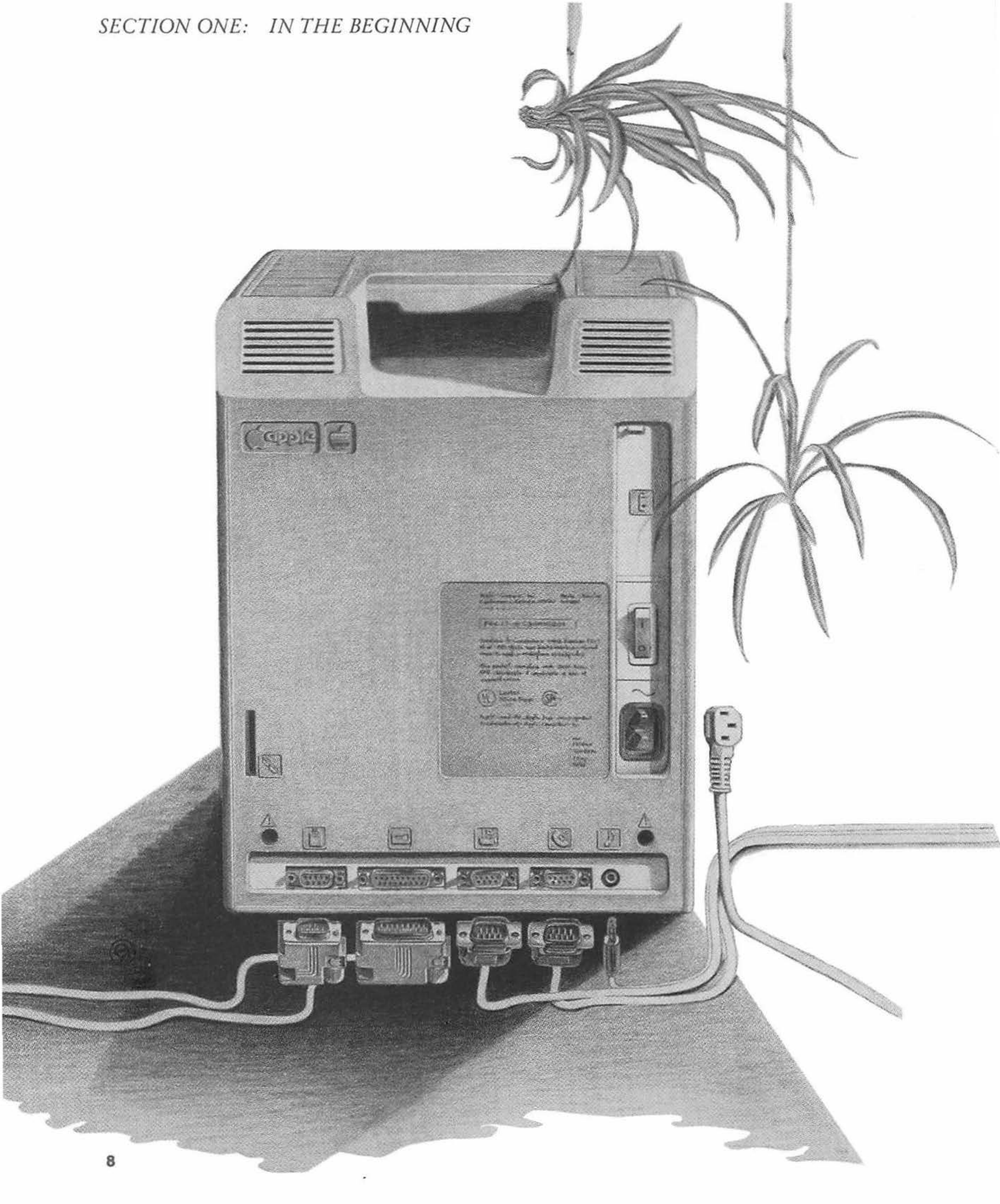
Apple supplies an excellent manual with the Macintosh that explains how to set it up, so this section will only present supplementary information.

When you set up a Macintosh, the first thing you will need is a comfortable, properly lit place for it. You will probably want the keyboard at typing height, a little lower than your desk or dining table. Adjust your chair so you're seated comfortably; the fatigue some people feel when working for long periods at a computer usually comes from poor lighting or an uncomfortable chair or desk setup. If two people of different heights share the computer, use an adjustable chair or keep a cushion handy.

Avoid placing your Macintosh next to a heating vent or radiator. Exposure to high temperatures might damage it or, more likely, its magnetic information-storage disks. Generally, if you can stand the temperature, so can Mac.

Don't put the computer in direct sunlight, and make sure the area behind it is neither brilliantly lit nor very dark; otherwise your eyes will have to readjust constantly for the difference. No lights should shine directly on the screen; position them to avoid glare. If you have bright overhead office lights, a shade or hood over the screen may help. (See Chapter 12 for more information about reducing glare.)

As you assemble the computer, the only problem you are likely to have is with the plugs. If you have trouble pushing a plug in, you have the wrong jack or the wrong orientation. Never force a plug! Small, knurled knobs secure all plugs to the computer.



If the power supply is irregular in your area or if you've had problems with other equipment, see Chapter 20 for suggestions. Outside of North America, make sure that the electrical supply is the correct voltage. The power-line frequency (50 or 60 Hz) doesn't matter, but the voltage must be correct. Any transformer you use should have a capacity of 150 watts or 150 VA.

There is a little plastic piece labeled INTERRUPT RESET. It fits on the left side of the computer. You won't need to install this piece for now, so you can put it away for another time.

If you ever need to move your Macintosh, the safest way is to disconnect all the cables and move each unit separately.

TAKING CARE OF THE MACINTOSH

Although you should give your computer the same care and respect that you'd give any other valuable object, it is no more fragile than a television set and fairly hard to damage.

You cannot harm Macintosh hardware by any combination of typing on the keyboard or handling of the mouse. Generally, you can only alter or erase software or lose data by deliberate choice. We'll go over simple safekeeping steps as they come up.

TAKING CARE OF YOU

Don't work steadily at any computer for hours on end; plan to stretch your legs at least every hour or so. Don't forget to eat, and remember to talk occasionally with your fellow workers or your family and friends.

USING NEW PRODUCTS

Whenever you work with hardware or software for the first time, take a little extra care. Don't invest a lot of time entering data at the beginning; make sure that the program does what you need first. Enter a little data and check the results; print from the program to make sure everything works. Only after you are confident of both the program and your ability to use it should you invest the time in intensive work.

KEEPING UP WITH NEW PRODUCTS

You may find yourself caught up in the craze to get the latest and greatest of each software or hardware accessory. If you enjoy doing this, fine; but remember, familiar products that work well may be enough. Glamorous features may be irrelevant to your needs. Remember, too, that every time you change programs, your work will inevitably be disrupted until you learn the new one. Although the Macintosh design makes this period short compared with other computers, you should weigh the potential benefits against the liabilities. In a business, be especially sure before changing programs and working procedures.

A FEW WORDS ABOUT EACH COMPONENT

Here's a quick rundown on each component. You won't need more information to operate a Macintosh successfully, but you may want to learn more after you have some experience; you will find more detail starting in Chapter 11.

The Main Computer Unit

For now, you need to know only two things about the main unit: how to turn it on and how to adjust the screen brightness. You can leave the machine on constantly if you wish; just turn the brightness down whenever you're not going to use it for long periods.

Disks and the Disk Drive

The disks store information magnetically. They fit, one at a time, into the disk-drive slot just below the display screen. They can only go in right side up; the plastic disk jacket has a small arrow showing the correct orientation.

When you are finished working, you should use the Finder to eject all disks. I'll explain this shortly; for now, just note you should eject all disks before turning off the power.

Disks store several kinds of information in units called files. Functionally, a computer's files resemble the paper file folders in your filing cabinet. Stored information includes:

System and Finder information. When you first turn on the Macintosh, you must insert a start-up disk in the disk drive. The computer transfers System and Finder information from the disk into its memory and onto the screen. (We'll see what these terms mean in the next chapter.)

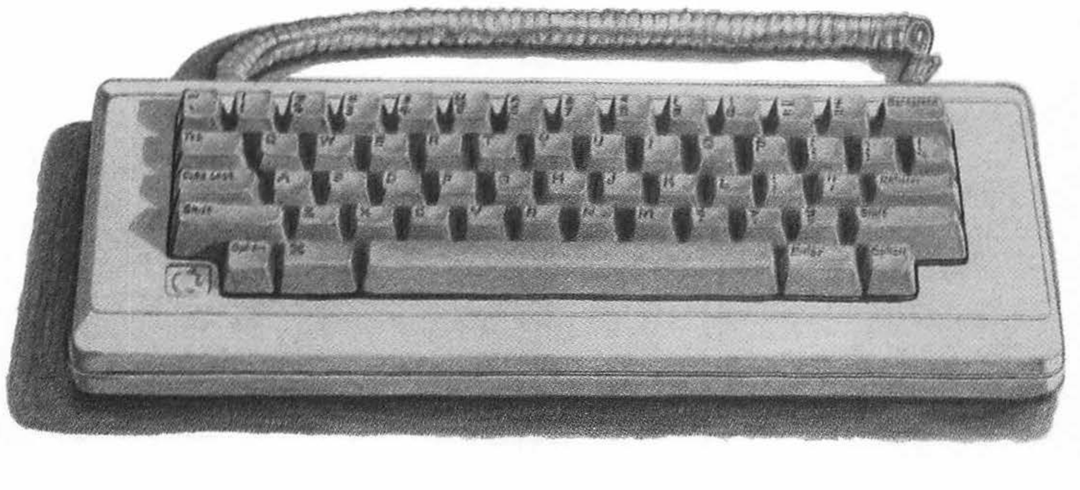
Application programs. In most cases, programs such as MacPaint and MacWrite that turn your Macintosh into a graphics tool or word processor will be on the same disk as the System and Finder information. Once the Finder is in memory, you'll need to indicate the program you want, and it, too, will be transferred into memory.

Data. Information you enter when using a program, and other information needed by a program, are stored in data files. Data files don't always have to be on the same disk with the program, although some programs will be easier to use if the data files are handy. You can keep these different kinds of information on the same disk, but if they must share space with application programs, as well as the System and Finder, you may run out of disk storage space. Chapter 9 explains how to set up disks to avoid this problem.

Don't be concerned if you don't follow all the subtleties of this discussion just yet. We'll take up these topics again, and you'll soon be fluent with disks and files.

The Keyboard

You'll use the keyboard (and mouse) to enter information and to tell the Macintosh what to do with that information. The Macintosh keyboard has all the familiar typewriter keys and a few extra ones. The typewriter-like keys produce all the familiar letters and symbols. If you hold a key down, the character will repeat automatically.



The Macintosh Shift key works just like a typewriter's; when you hold it down, you get capital letters and punctuation marks instead of lowercase letters and numbers. But the Caps Lock key works a little differently than a typewriter's shift lock: It stays engaged when you press it and releases on a second press. When the Caps Lock key is engaged, you get capital letters as you would expect, but you still get numbers, not the punctuation marks above them. For punctuation marks, you must press the Shift key *and* the number key at the same time, even if Caps Lock is engaged.

Two keys you won't find on a typewriter, the Option and Command (⌘) keys, work like two more shift keys. You press them at the same time as another key. The Option key serves several purposes, including generating special symbols for foreign languages or scientific equations. The Command key shouldn't concern you if you are just starting out; it is a shortcut for telling the computer to do something like eject the disk or delete a word. We'll cover its functions when we discuss software.

The last non-typewriter key is the Enter key. You'll use it most often to tell your Macintosh you've finished with some typing.

A last word of warning: If you are in the habit of typing the letter "l" for the number 1, or a letter "O" for zero, you'll have to change your ways. Sorry. The difference is very important to computers.

The Mouse

Along with the keyboard, the mouse lets you work with the Macintosh. We'll cover it in Chapters 2 and 15.

ADDING ON: HARDWARE ACCESSORIES

There are four add-on accessories you should consider buying. All are simple plug-in units; two are probably essential:

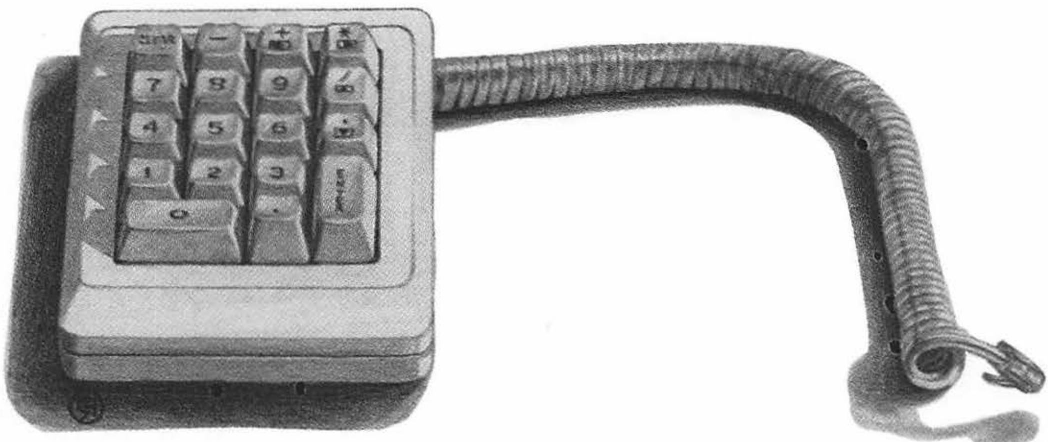
A second disk drive makes computing much easier by providing more storage for programs and data. Although you can limp along with a single disk drive, a second one makes many tasks so much easier that you should consider it a necessity rather than an option. For example, with two disk drives you can copy disks or move information from disk to disk without constantly swapping disks during the operation. The Macintosh will accept only one additional microfloppy disk drive.

Many users should consider a hard disk drive, which gives greatly increased storage and much faster operation.

A printer produces text and graphics on paper, and nearly every Macintosh owner will need one.

Two other accessories are useful but may not be essential.

A numeric keypad provides an adding machine-like set of keys, handy if you work with numbers. The keypad also improves some Macintosh operations by giving you more keyboard options but it is not necessary for most software.



A modem lets you communicate with other computers by telephone. It converts information from your computer into sounds that can travel by an ordinary telephone line; it also does the reverse, changing sounds generated by a distant computer and its modem back into information your computer can understand.

You are ready to start using the Macintosh.

Starting Up

Chapter 2

If you have used a Macintosh already, you should probably skip this chapter, which covers basic information but does not replace Apple's excellent manuals. If you are new to the Macintosh, I suggest you read the manuals and use this chapter as a supplement.

STARTING UP THE MACINTOSH

Any disk that will start up a Macintosh has all the features you need to perform the operations described in this chapter. A disk that can start up a Macintosh is called a system, or start-up, disk.

Insert a system disk into the disk-drive slot.

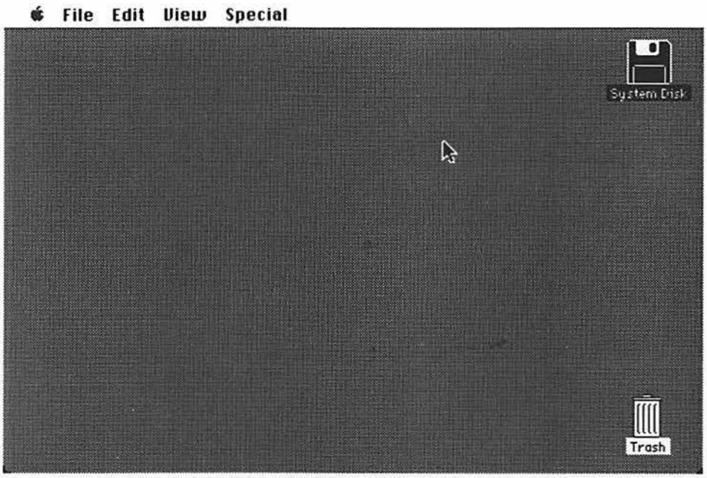
The disk drive whirs as it loads start-up information from the disk into the computer's electronic memory. A little wristwatch appears on the screen, telling you to wait.

LOOKING AT THE DESKTOP

After the disk drive stops, you see the Desktop on the screen; this is part of the Macintosh's visual interface. The icons, or pictures, on the Desktop represent places for finding, storing, and discarding information. The first icons you see are:

- The disk you've just inserted.
- A trash can.

A window may open as well. If so, try to ignore it for the next five minutes. If it gets in the way, scan ahead to "Closing a Window."



USING THE MOUSE

Move the mouse around your desk.

The pointer on the screen follows the mouse movements. You cannot move the pointer off the screen.

The pointer you see now is shaped like an arrow, but it can take on other shapes, as we'll see later.

Pointing

Move the pointer over an icon.

The exact spot you are pointing at depends on the pointer's shape. When the pointer is an arrow, position the arrow tip over the object.

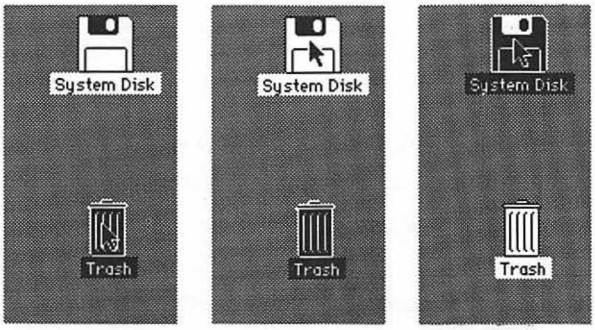
Clicking

Point at the Trash icon.

Press and release the mouse button once.

This action is called clicking. When you click the white Trash icon, it turns black. The black color indicates you have selected that icon. Selecting an icon means your next action will apply to that icon.

Move the mouse to the disk icon and click it.



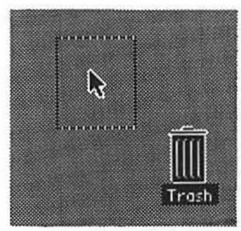
The Trash icon turns white again, and the disk icon turns black.

Click anywhere on the Desktop outside the icons.

This cancels your previous selection. Now none of the icons on the Desktop is selected.

Dragging

Move the pointer over the Trash icon.



Press and hold down the mouse button.

Move the mouse while holding the button down.

You use this essential operation, called dragging, in many ways. As you move the mouse, the icon's outline moves with it.

Release the mouse button.

When you release the button, the icon pops over to the location of its outline.

Pressing

Pressing means that you position the pointer and then press the mouse button, holding it until an action is complete. Pressing differs from dragging because you don't move the mouse while pressing the button. We will get to an operation that requires pressing shortly (in the section on scrolling).

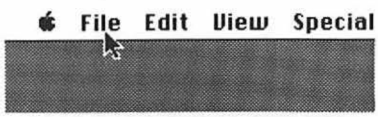
OPENING AN ICON

To find out what information an icon represents, you must open the icon; you can think of this as opening a drawer in a filing cabinet.

Click the disk icon (the system disk).

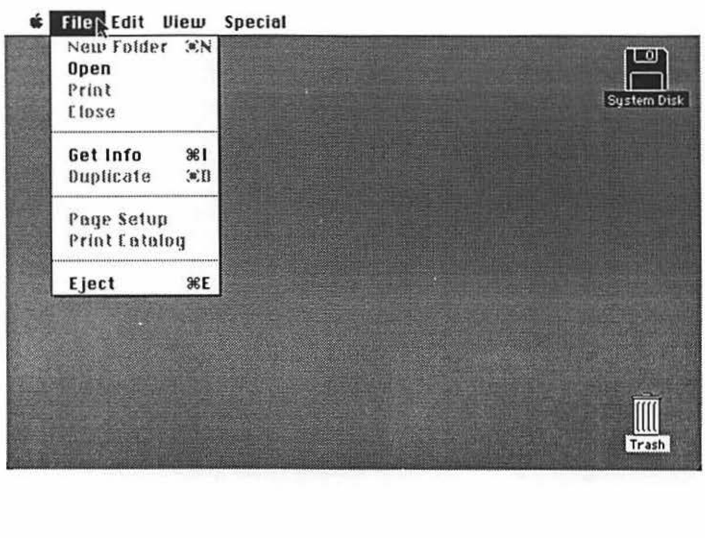
Move the pointer to the words (menu bar) at the top of the screen.

Point at **File.**



Press and hold down the mouse button without moving the mouse.

You've "pulled down" the File menu. The menu entries are things you can do with the icon you've selected—in this case, the system disk. Available menu choices appear in black letters; unavailable ones, such as Close, appear in dimmed or gray letters.

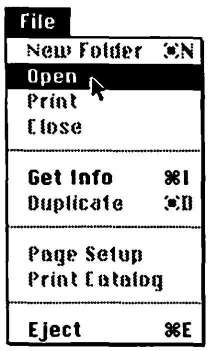


Still holding down the mouse button, move the pointer down the menu.

The available choices are highlighted in turn.

Position the pointer over **Open**.

The word Open is highlighted.



Release the mouse button.

You select a menu item by releasing the mouse button when the item is highlighted. In Macintosh terms, you have chosen Open from the File menu.

The disk window now opens up.

Double-Clicking

To open an icon faster, point at it and quickly click the mouse button twice; this action is called double-clicking.

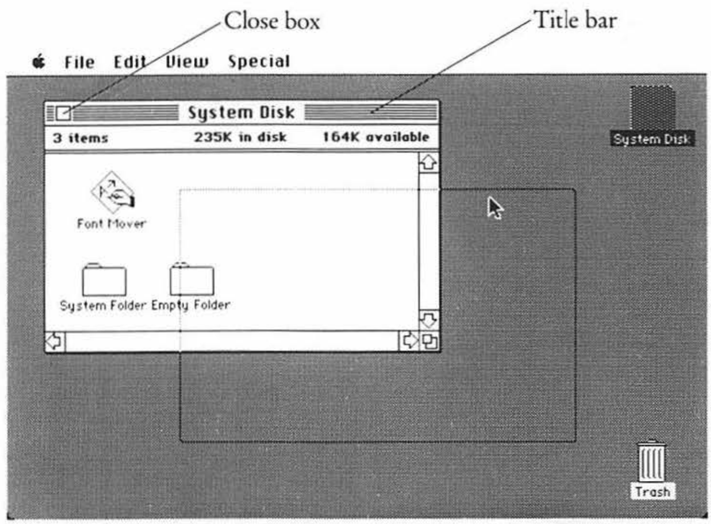
LOOKING AT THE DISK WINDOW

Each icon in the disk window represents a file on the disk. Some files may be application programs; others may be documents or drawings; still others may represent information needed by the Macintosh or used by a program. The operations described here apply to most Macintosh windows, not just the disk window.

Moving a Window

You may need to reorganize your Desktop by moving windows around on it.

Point at the black-banded bar at the top of the window and drag it with the mouse.



The area with the black bands is the window's title bar. Dragging the title bar moves the entire window.

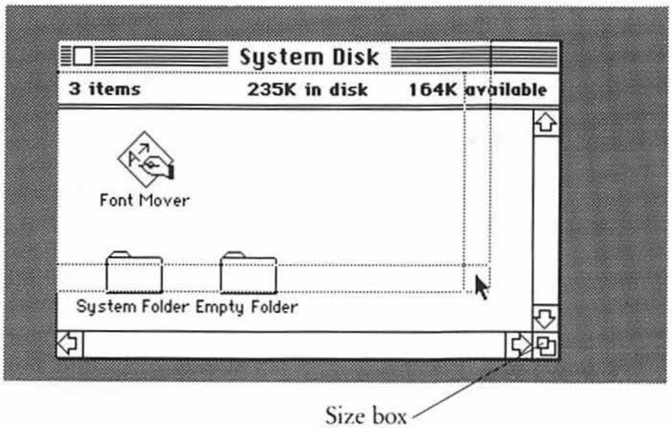
Release the mouse button.

The window moves to the location of its outline.

Changing a Window's Size

To reduce clutter on your Desktop, you can make a window smaller; if you want to see more of a disk's contents, you can make its window larger.

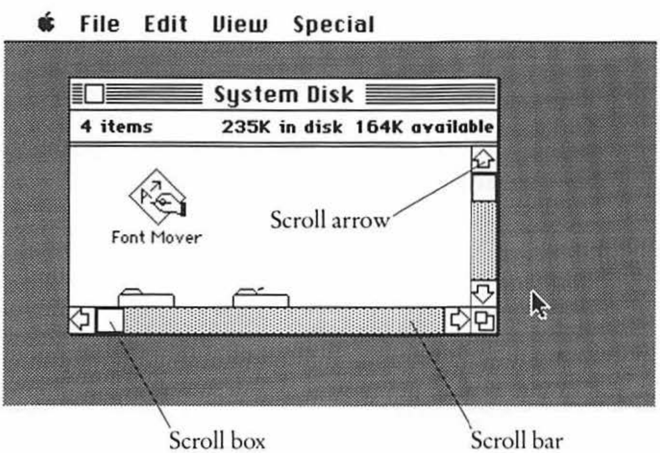
Point at the size box at the lower right corner of the window and drag it with the mouse.



Notice that some of the icons drop out of sight as the window is dragged smaller.

Scrolling the Disk Contents

Sometimes a disk may contain more files, represented by icons, than you can see in the window at one time; to move the invisible icons into view you need to scroll the disk contents past the window.



Click one of the scroll arrows.

Clicking the arrows moves the disk's contents past the window by a small increment (a line if you're looking at text).

Click in the gray area of one of the scroll bars.

Clicking in the gray area instead of the arrow moves a larger increment (a screenful if you're looking at text).

Position the pointer over a scroll arrow and press the mouse button without moving the mouse.

Remember, this action is called pressing; it's a fundamental mouse operation.

The disk contents scroll by the window until you release the button or until there is nothing more to scroll.

Drag the scroll box.

This action brings hidden icons into view. The scroll box indicates the relative location of a window with respect to the contents.

Some windows do not permit resizing or scrolling and therefore lack a size box or one or both scroll bars. If a disk window is big enough to display all the contents of a disk, the scroll bars turn solid white, and you can't scroll the window.

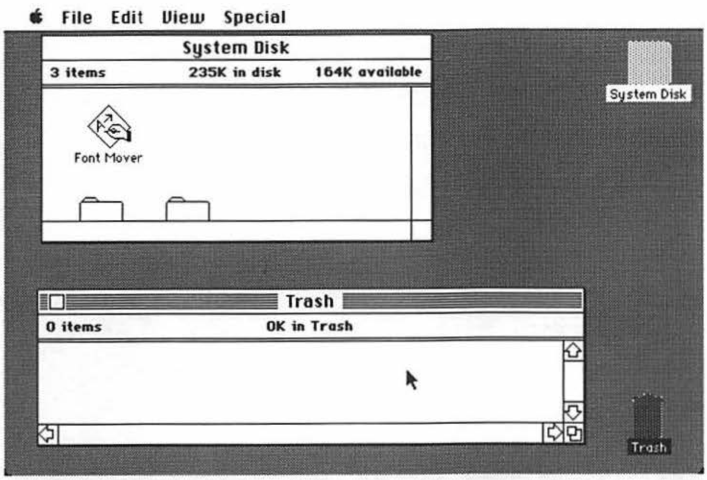
OPENING MORE THAN ONE WINDOW

You can have several windows open on the screen at the same time.

Double-click the Trash icon.

Double-clicking is a fast way to open an icon. (Remember, point at the icon and quickly click twice on the mouse button.)

This opens another window. It's empty because there is nothing in the trash.



The window with black bands across its title bar is the active window. The black bands have disappeared from the disk window's title bar, showing that it is no longer active. (You might have to hunt around for the disk window if the Trash window has covered it up. Move or resize the Trash window if necessary.)

Click anywhere in the disk window.

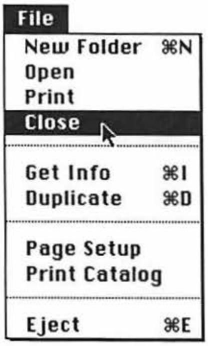
The black bands are back in the disk window's title bar, showing that the disk window is again active.

Drag the disk window's size box so that the window overlaps but does not obscure the Trash window.

Click anywhere in the Trash window.

The Trash window moves on top of the disk window and black bands appear in the Trash window's title bar. If you have several overlapping windows on your Desktop, the active window will always be in front.

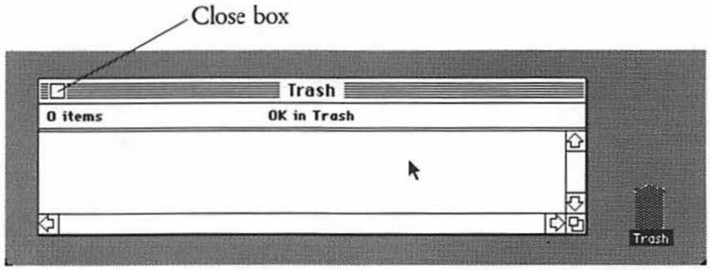
Closing a Window



You may close your windows in either of two ways:

Choose **Close** from the **File** menu; *or*

Click the close box in a window's upper left corner.



Close the Trash window; leave the disk window open.

WHAT THE DISK WINDOW TELLS YOU

Now that you can manipulate windows, let's take a closer look at the system disk window.

Click the disk window to make it active.

Drag its size box to make a large window.

The words and icons you see tell you which items you have stored on the system disk and how much space they take up.

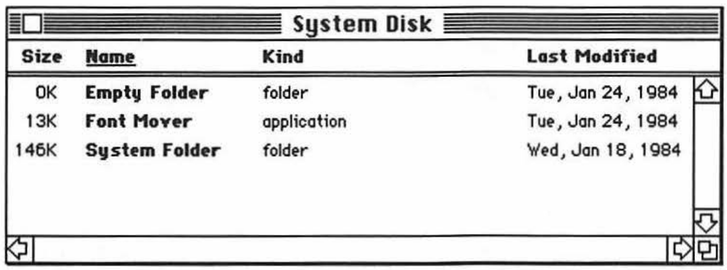
A disk can hold only a finite amount of information. The top line in a disk window tells you how many items are on the disk, the total storage taken up by the files on that disk, and how much space is still available for additional items. The units are thousands of characters, or kilobytes.

The icons represent individual files of information on the disk. Some of these files are application programs; others contain software the Macintosh itself needs to operate; still others contain resources that you will find useful once you start working with the application programs.

You can use the menus to find out more information about the files on the disk.

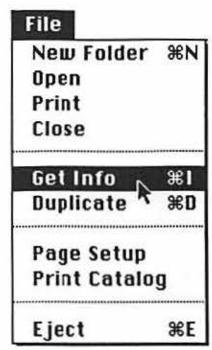
Choose **by Name** from the **View** menu.

(Point at View on the menu bar, drag the mouse to highlight by Name, and release the button.) The icons disappear, and a list of files appears in alphabetical order. Along with each file's name, you have its size and kind, and the date the file was last modified.



Pull down the **View** menu again, holding down the mouse button.

These choices represent ways you can organize the list of files. If you choose by Date, for example, the files are listed starting with the one most recently changed.

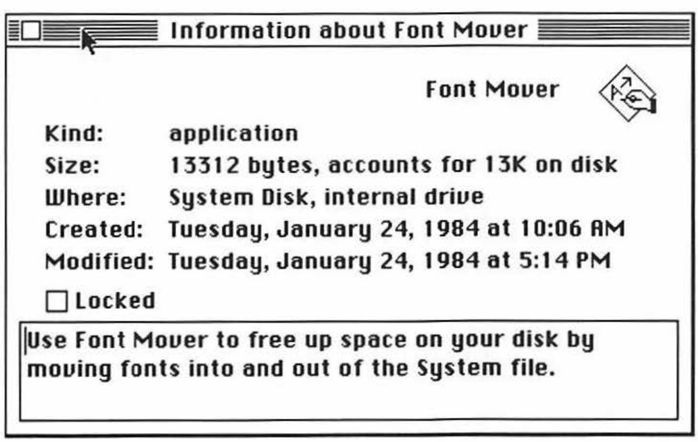


Drag to **by Icon** and release the mouse button.

Click any file icon.

Choose **Get Info** from the **File** menu.

A new window opens, showing information about the file.



To close the Get Info window, you can either:

Click the close box in the Get Info window; *or*

Choose **Close** from the **File** menu.

TAKING A BREAK

Choose **Close All** from the **File** menu.

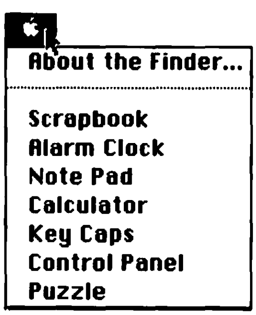
Choose **Eject** from the **File** menu.

After a few seconds, the Macintosh pushes the disk out of the disk drive. Turn off the power if you wish.

To restart, simply turn on the power and insert the disk again.

DESK ACCESSORIES

The Apple menu in the menu bar opens the way to the Macintosh desk accessories. You can use the desk accessories at any time, regardless of other activity—whether you are using a program or not, whether any windows are open or not.



Point at the Apple symbol at the far left of the menu bar.

Press and hold the mouse button.

Drag the mouse down to highlight **About the Finder...**

Release the button.

Some information about an important program called the Finder appears on the screen. The Finder creates your Desktop, shows you which files are on the disk, copies files from disk to disk, and much more. You've been using the Finder throughout this chapter.

Click anywhere to get rid of the Finder information.

Go back to the Apple symbol and pull down its menu again.

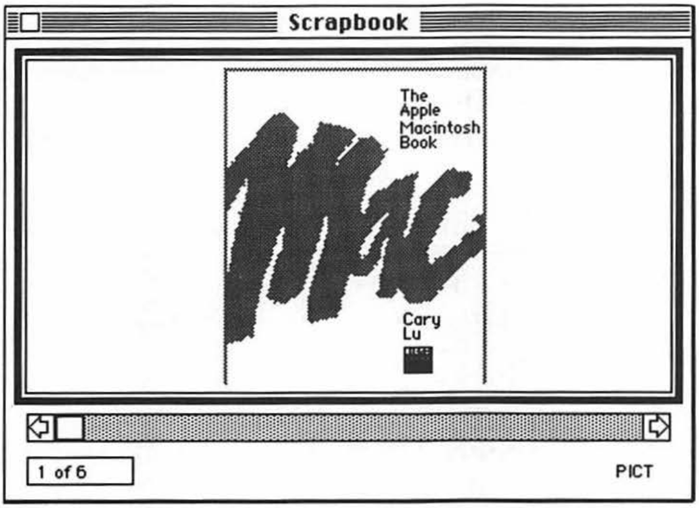
The rest of the items in the menu are desk accessories. (Your menu may not be in the order shown.)

Choose each accessory in turn.

Click the accessory's close box to put it away.

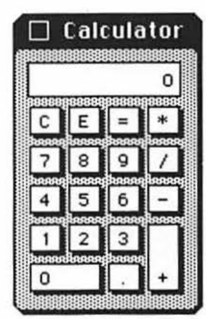
What Do the Accessories Do?

Scrapbook saves material from one program and lets you move the information to another program or within the same program.



Alarm Clock displays a small box with the current time. Click the small symbol on the right to add a calendar. Now click the small alarm-clock icon. You can set the alarm by moving the cross-hair pointer over either the hours or minutes and clicking; then click the up or down arrow to set the time. Turn the alarm on by clicking the bell icon on the left.

Note Pad gives you a place to type text notes. You can tear off sheets (the Note Pad gives you eight) by clicking the upturned corner. Clicking the lower left corner below the diagonal line flips the Note Pad back a page. Mac beeps if you try to put too much text on a sheet. We'll use the Note Pad in the next section.



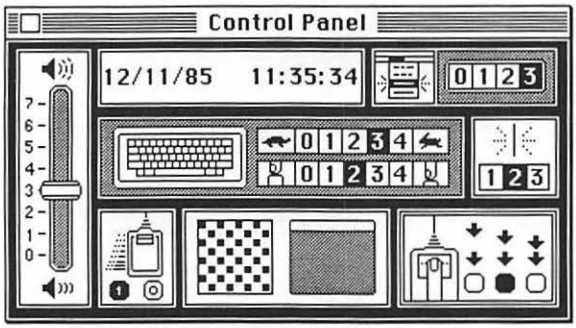
Calculator is a general-purpose, four-function calculator with exponentiation. It uses standard algebraic entry. Use the mouse to click the numbers or functions you want; the results appear in the calculator's display panel. You can also enter numbers from Mac's keyboard. The * symbol denotes multiplication.

Key Caps displays a miniature Macintosh keyboard that tells you which characters your real keyboard generates. Press a few keys on the real keyboard or click them on the Key Caps display with the mouse and watch what happens; the keys are highlighted on the Key Caps display and appear in the blank bar above it.

Hold down Shift, Option, Caps Lock, or a combination of these keys on the real keyboard and watch what happens; the display keyboard is instantly transformed into a variety of symbols and graphics characters.

Puzzle will look familiar; you can play with it when the boss isn't looking.

Control Panel lets you adjust the speaker volume (drag the sliding control up or down), set the repeat rate of the keys (turtle for slow, hare for fast), adjust the rate the pointer flashes, set the clock/calendar, and other features; see your manual for details.



In this section we've looked at standard accessories supplied by Apple with the Macintosh. Many others are available from other software sources; these are described in Section Two.

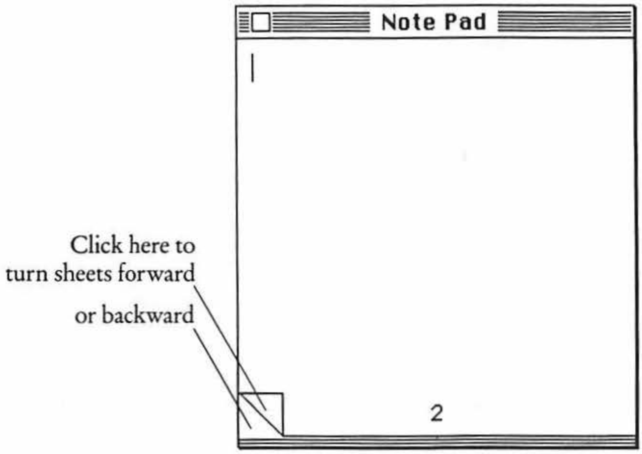
Again, you may want to close everything to take a break before we move on to practice a major operation: editing text.

EDITING TEXT

For most Macintosh application programs, you will use the same general methods for entering and editing text. We'll practice these operations using the Note Pad, since all system disks have one.

Choose **Note Pad** from the Apple menu.

If the first page has writing on it, turn to the next clean sheet by clicking the upturned left corner.



You should now see a blinking vertical bar at the top left of the Note Pad page; this bar marks the text insertion point. Anything you type appears at this point. This insertion point is used by all application programs for text entry.

Type two or three lines.

If you make a mistake, hit Backspace and retype.

Your text appears on the Note Pad, pushing the insertion point ahead of it. Notice how the words wrap around to the next line when you get to the edge of the page. You don't have to hit Return at the end of each line as you do on a typewriter; the Macintosh software does it for you.

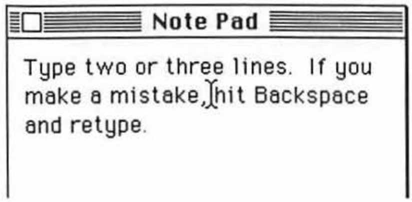
Inserting Text

Suppose you have left out a phrase in the second line of the text you just typed. You want to go back and insert the phrase where it belongs.

Move the mouse so the pointer lies somewhere within the text.

Notice how the pointer changes shape as it moves onto the Note Pad, becoming an I-beam. You can use the I-beam pointer to mark where in the text you wish to insert the phrase you left out.

Move the I-beam pointer to anywhere on your second line of text and click.



The blinking insertion point, which was at the end of your text, moves to the location of the I-beam pointer. If you now type in the missing phrase, it will be inserted in that location, pushing the insertion point and the existing text ahead of it.

Practice moving the insertion point around your text.

Move the I-beam pointer to a blank area at the end of a line and click.

The insertion point appears at the end of the text in that line.

Move the I-beam pointer to the blank area following all the text you have typed and click.

The insertion point appears at the end of the last line.

Deleting Text

Click so the insertion point appears somewhere within the text.

Press the Backspace key.

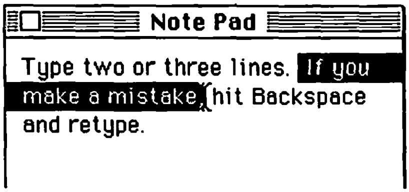
Using the Backspace key is the simplest way to delete something; backspacing removes the character to the left of the insertion point.

You can also use Backspace to delete more than one character at a time. But first, you have to use the I-beam pointer to select what you want to delete.

Move the I-beam pointer to the beginning of the text you wish to delete.

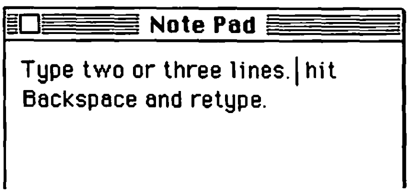
Drag the I-beam pointer to the end of the text you want to delete and release the button.

Note that the text is now highlighted.



Hit Backspace.

Gone!



Once you've selected text with the I-beam pointer, you also have three other ways of deleting it:

Choose **Cut** from the **Edit** menu; *or*

Choose **Clear** from the **Edit** menu; *or*

Hold down the Command key (⌘) while typing x.

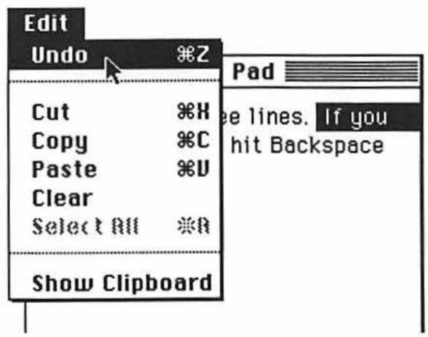
Using the Command key is a shortcut to an operation; you simply hold down the Command key and type a letter instead of using the mouse to pull down and choose from a menu. The Command key equivalents for menu items are shown next to each item. From now on, this kind of instruction will be given as: Type Command-x.

Changing Your Mind

If you delete the wrong thing, the Macintosh gives you two ways to repair your mistake:

Choose **Undo** from the **Edit** menu; *or*

Type Command-z.



Either operation restores your last deletion.

Choose **Undo** again.

The selection is deleted again. Undo reverses whatever your last move was; if your last move was to restore a deletion, then Undo deletes the restoration. You cannot undo more than one previous step.

In some programs, the Undo menu item changes to reflect your last action. It might say Restore Row if you've just deleted a row, but the item will always be in the same menu location.

Erasing the Note Pad

Select the entire text by dragging the mouse down over all the lines.

Choose **Cut** or **Clear** from the **Edit** menu.

These actions erase the entire practice sheet.

Closing the Note Pad

Click the Note Pad's close box.

If you don't erase what you've written, your text is stored in a file on the disk named Note Pad when the window closes. The next time you use the Note Pad, you'll find your notes on the page where you typed them. Try it. To end this practice session:

Close the Note Pad window.

Close any other windows you might have opened.

Close the system disk window.

You can close windows one at a time by clicking each close box or all together by choosing Close All from the File menu.

Choose **Eject** from the **File** menu.

The disk drive ejects the disk.

TURNING OFF THE MACINTOSH

Do not turn Mac off before using the File menu to eject your disk. If you turn off the computer before ejecting the disk, you may lose some information that Mac has not yet stored on the disk (see Chapter 14 for details).

A NOTE ABOUT PHILOSOPHY

As you learn more about the Macintosh, you will find that the best programs use a consistent philosophy. Such programs will have no unpleasant surprises.

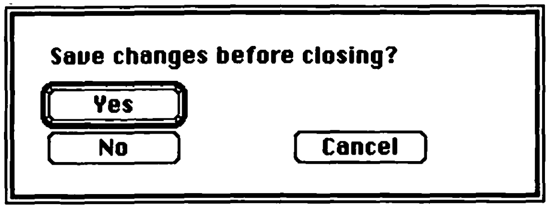
For most Macintosh operations, you always select something first, then choose what you want the computer to do with it. Thus you select a file, then choose Get Info; you select a portion of text, then choose Cut or Copy.

A few programs, by their nature, don't work quite in this fashion. In the MacPaint program (Chapter 4), you select a tool first, and *then* you do something with it. Yet selections of an area within MacPaint follow the normal rule: Select first, then choose what to do with it.

Most application programs will also use the same two ways of finding out more specifically what you want them to do: conversational dialog boxes and more urgent alert boxes.

Dialog Boxes

As you work with the Macintosh, application programs will at times put dialog boxes on the screen in response to your actions. For example, a dialog box might ask you the size of paper in your printer. Some dialog boxes merely ask a yes/no question.

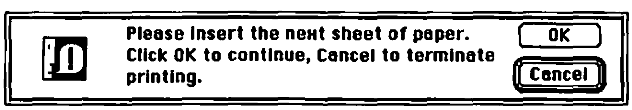


In most dialog boxes, you make choices by clicking buttons or items on a list. The next step depends on the dialog box. Often, you click a button labeled Yes or OK when you are satisfied with your choices, or you click a button labeled Cancel to close the box and return to the previous step. You cannot proceed past a dialog box unless you click an appropriate button.

One special form of dialog box is a mini-finder—a listing of the files on the disk that were created by the application program you are using. A related form is a mini-list, which might, for example, contain the font choices available for a particular program.

Alert Boxes

An alert box is similar to a dialog box, but is used to warn you of a potentially serious problem. An alert box might tell you that your word-processing document is getting too long to store on the disk, for example. In most cases, you must acknowledge the alert box by clicking an OK button.



WHAT TO READ NEXT

The next seven chapters describe some major software types. Most Macintosh users will want to read the sections on MacPaint and MacWrite in the next two chapters. Read the other chapters if you are interested in the software.

Since this chapter has covered only the essential operations to get started, you may want to skip to Chapter 9, which covers disk organization. The information in Chapter 9 is important for any productive work on a Macintosh.


Sectio

A Macintosh Software Survey

Working with Macintosh: This section describes how to perform basic operations with the Macintosh interface and introduces you to software available for the Macintosh.

Chapter 3
Word Processing

Almost everyone who buys or uses a computer eventually does some form of word processing on it. In fact, many people use their machines for nothing but word processing. With a computer, writing and editing anything from a letter to a book becomes easier, faster, and far less messy.

Instead of scrawling out drafts by hand or banging out innumerable versions on a typewriter, you type your text into the computer only once, changing words and phrases on the screen as you type. The computer allows you to move paragraphs, delete sections, and add new sections—all without having to retype pages over and over. When you like what you've written, you can print a precisely formatted copy immediately.

At least that's the way it should be. A major challenge for word-processing program developers has been to give you on paper exactly what you see on the screen—not an easy task. With traditional word processing, you could rarely be sure exactly where new pages of text began, much less see italics or headlines on the screen.

The Macintosh has changed all that. At last, what you see *really is* what you get.

MACWRITE

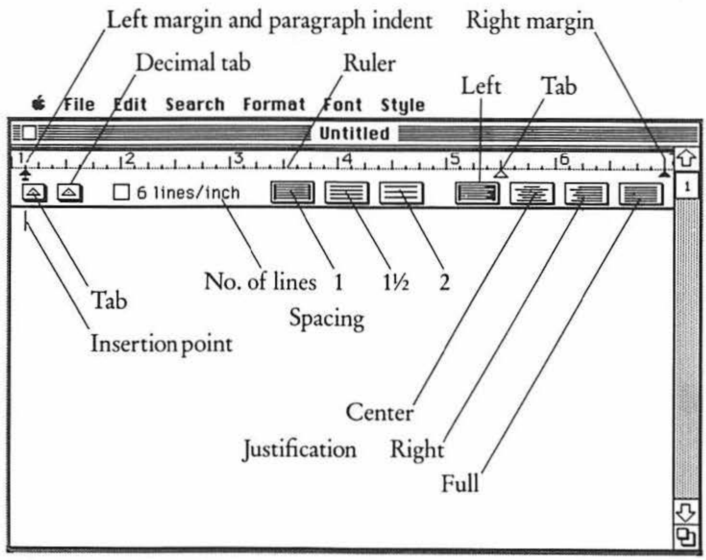
This chapter begins with MacWrite. If you are already familiar with this program, skip ahead to later sections, which describe and compare other word-processing software for the Macintosh.

Starting MacWrite

For this practice session, you'll need any start-up or system disk containing the MacWrite program.

Double-click the MacWrite icon to open it.

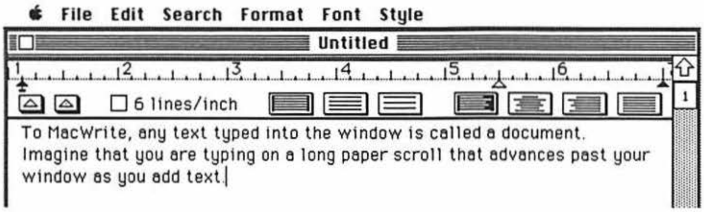
The Desktop clears and the wristwatch appears, telling you to wait for the MacWrite document window to open. After a few seconds the wristwatch disappears and the window, labeled Untitled, opens.



Notice the blinking vertical bar at the top left margin; this marks the insertion point (equivalent to the cursor in other word processors) where the text you type will appear.

Type a few sentences—anything.

Imagine that you are typing on a long paper scroll that advances past your window as you add text. Notice what happens when you get to the end of a line: MacWrite automatically bumps words to the next line without your having to press Return; this is called word wrap. The only time you have to press Return is at the end of a paragraph.



Inserting and Adding Text

With the mouse, you can move the pointer anywhere within the text. You use it to position the insertion point so you can add text.

The insertion point will only appear within or at the end of your text; you cannot put it in the white space following the text or in the margins.

Move the pointer into the text area.

The pointer changes shape when it's in the text, becoming an I-beam pointer. Whenever you're working with text on a Macintosh, you will see the I-beam pointer and can use the same basic operations described here.

Position the I-beam pointer between two words and click.

typing on a long paper scroll

The blinking bar marking the insertion point appears wherever you click within the text.

Type a few words.

typing on a long papyrus or paper

The extra text you add is inserted at the insertion point, between the two words.

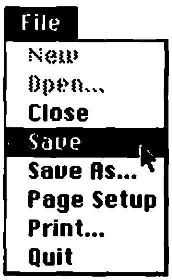
What if you change your mind? You can undo your last step.

Choose **Undo Typing** from the **Edit** menu; *or*

Type Command-z.

Undo reverses your last action, back to the last click of the mouse.

Saving a Document

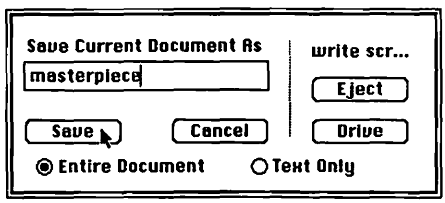


Choose **Save** from the **File** menu.

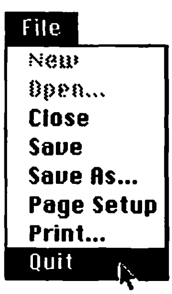
A dialog box comes on screen, asking you to name the file.

Type a file name.

Click the Save button.



The disk drive turns on, and your document is saved on the disk. After the file is stored, you can resume working on your document. If you want to leave the MacWrite program completely:



Choose **Quit** from the **File** menu.

If you have made any changes since the last Save, a dialog box asks if you want to save your current work. If you are just experimenting, click No, and nothing will be saved. But if you want to save the latest version:

Click Save.

If the file has no name ("untitled"), a new dialog box asks you for a file name.

Type in a file name and click Save.

After you save the document, the Desktop with the disk window reappears.

Reading in an Existing Document

In the disk window, locate a file you want to work with. Make sure that the file you select is a MacWrite document.

If you open a file icon, the system will automatically start the program that generated the file and read in the file information.

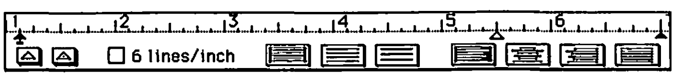
Double-click the file icon.

The contents of the file you have chosen now appear in the document window.

Formatting with MacWrite

With conventional word processors, setting the format of your document—the margins, tabs, line spacing, and alignment—often takes more effort than typing the text. MacWrite uses visible rulers with special markers to make setting formats easy and elegant.

When you start a document, MacWrite supplies a ruler with a preset format that you can keep or change. The ruler corresponds to the text area on a standard 8½- by 11-inch sheet of paper; the full margins do not appear in Mac's document window.

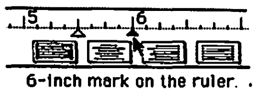


Notice the format of the text on your screen. Changing this format simply requires moving the margin or indentation markers and selecting the spacing and justification, or alignment, icons.

Margins and paragraphs

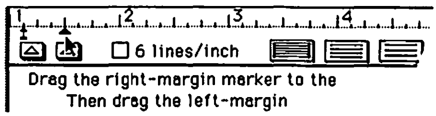
There are two markers at the left edge of the ruler. They are superimposed in the preceding illustration, but can move independently, as you'll see in a moment. The triangle marks the left margin and affects any word-wrapped text; the arrow marks the paragraph indent and affects only the first line of each paragraph (any line immediately following a Return). The triangle at the right edge of the ruler marks the right margin.

To drag a margin marker, position the pointer carefully on the triangle and drag.



Drag the right-margin marker to the 6-inch mark on the ruler.

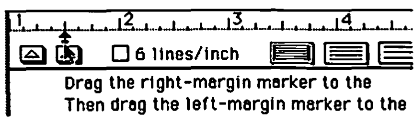
Drag the left-margin marker to the 1½-inch mark on the ruler.

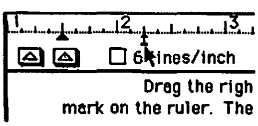


Notice what happens to your text.

Drag the paragraph-indent arrow so that it is on top of the left-margin marker.

If you drag both the left-margin and indentation markers to the same position, all your text lines up on the left margin with no paragraph indents.

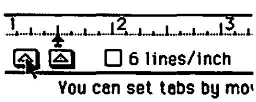




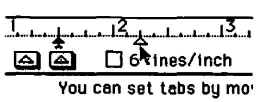
Drag the indent arrow first to the left and then to the right of the left-margin marker.

Dragging and placing the indentation marker to the right of the left-margin marker gives you normal paragraph indents; dragging it to the left gives you hanging indents—the first line of each paragraph extends farther left than the rest of the text.

Tabs



You set tabs by dragging the little triangles at the left, under the ruler. The triangle on the left is a standard tab; the other triangle (with the dot) is a decimal tab. The decimal tab is used for entering numbers in a column; decimal points will line up on the tab position.



Drag a tab triangle to any position on the ruler.

You must drag a tab triangle to the ruler; the tab icon disappears if you drag it to any other location.

If you press the Tab key while typing text, the insertion point will move to the next tab position, just as on a typewriter. You can drag as many tabs as you need from their boxes.

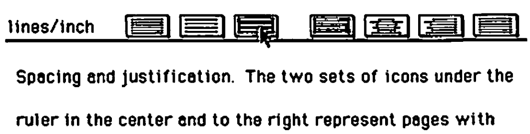
Drag the tab triangle away from the ruler and release the mouse button.

The tab icon disappears and the tabs are cleared.

Spacing and justification

The two sets of icons under the ruler in the center and to the right represent pages with different spacing and justification. The choice of 6 lines/inch forces the standard vertical spacing used on mailing labels and many forms.

Position the pointer in the double-spaced page icon and click.

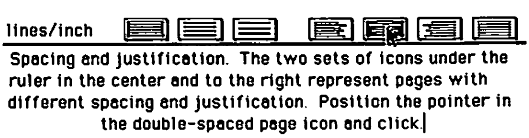


Your document is double spaced.

Click the single-spaced page icon.

Your document is single spaced.

Click the page icon with center alignment.



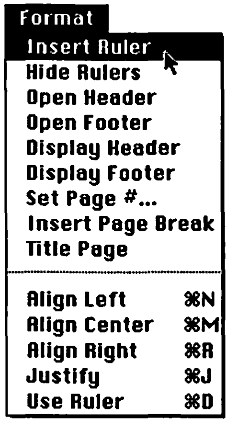
Click the other page icons in turn.

Your text lines change justification instantly on screen: left justified, centered, right justified, and both margins justified.

Changing the format within the text

A ruler sets the format for any text that follows it, until you insert another ruler. You can insert a new ruler anywhere.

Choose **Insert Ruler** from the **Format** menu

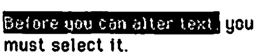


You can make any changes you wish to the new ruler. If the rulers are too intrusive, you can make them invisible by choosing Hide Rulers from the Format menu.

Editing by Selecting

Editing means changing—anything from correcting a typo to doing a wholesale cut-and-paste operation. Before you can alter text, you must select it. To select text:

Position the pointer just before the text.



Drag the pointer across the text you wish to select.

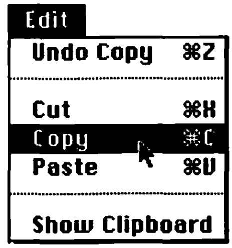
Release the mouse button.

The selected text is highlighted in reverse color. Your next command will affect only the selected text.

You can select an individual word quickly by pointing at it and double-clicking.

You can change the selected text in many ways. Three fundamental editing operations will copy, move, and delete it.

Choose **Copy** from the **Edit** menu.



The selected text is copied into the Clipboard, which is an electronic scratch pad.

Move the pointer to another part of the text and click.

The blinking insertion point appears in the new location.

Choose **Paste** from the **Edit** menu.

The selected text is copied into the new location; it also remains intact in its original location.

Select another portion of text.

Before you can alter text, you must select it. Before you can alter text,

Choose **Cut** from the **Edit** menu.

The selected text is deleted and disappears into the Clipboard.

Move the pointer to another part of the text and click.

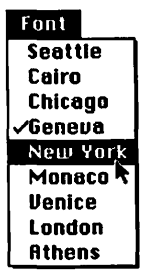
Choose **Paste** from the **Edit** menu.

The text is moved from the Clipboard into the new location. The Clipboard can hold one item at a time. Any time you choose Cut or Copy from the Edit menu, you replace the previous Clipboard contents with the new selection. So if you want to get rid of some text entirely, you can simply cut it to the Clipboard and leave it there until it is replaced by something new.

Choosing Typefaces and Type Sizes

MacWrite lets you choose the typeface, or font, in which your text is displayed and printed; you can also choose the size and style (bold, italic, outlined, shadowed, and so on).

Font choices are stored with the text. Whenever you move the insertion point and add new text, MacWrite automatically uses the font of the adjoining text.



Select some text by dragging through it so it is highlighted.

Choose any font from the **Font** menu.

The highlighted text is now displayed in that font.

CHOOSING A WORD PROCESSOR

You can choose from several different word processors for the Macintosh:

- MacWrite is a simple, handy word processor. Versions 4.2 and later probably have all the functions that the average user needs for straightforward tasks, and as long as it comes with the Macintosh, the price is right. MacWrite can now deal with long documents.
- Microsoft Word (version 1.05) offers a full complement of features that rival and surpass those of dedicated word processors. For major editing projects, Word supplies much more power than MacWrite or Jazz.
- Jazz (Release 1), from Lotus Development Corporation, includes a word processor that is essentially the same as MacWrite, but it works smoothly with the Jazz spreadsheet and business graphics. See Chapter 7 for more information about Jazz.

How They Compare

All these programs are fine for ordinary word-processing tasks. Their basic functions are similar enough that you can switch from one to another with only minor difficulty. But since they use different file formats, you will be better off choosing one as your principal word processor. Here's how they compare (the specifics may change as features are added; for an explanation of technical terms, see the glossary or Chapters 11 and 14):

Size of file

MacWrite: Limited by disk capacity.

Word: Limited by disk capacity.

Jazz: Limited to RAM available; normally not a problem but depends on other documents in RAM.

Windows

MacWrite: Single window, cannot be split or scrolled horizontally.

Word: Up to four windows with horizontal scrolling. Any window can be split so you can see two parts of the same file.

Jazz: Up to eight windows with horizontal scrolling.

Setting margins and tabs

MacWrite: Ruler on screen shows indents, margins, and two types of tab (left-aligned and decimal).

Word: Ruler displayed temporarily on screen upon request; shows indents, margins, and four types of tab (left-, center-, and right-aligned, as well as decimal). Can automatically fill spaces between tabs with hyphens, underlines, or periods.

Jazz: Same as MacWrite.

Pictures and graphics

MacWrite: Can include MacPaint images or other graphics in text, but not on the same line as text.

Word: Same as MacWrite.

Jazz: Same as MacWrite; includes live links to spreadsheets and graphics from within Jazz, so if you update information in a spreadsheet or graph, the document is automatically updated as well.

Visual page display

MacWrite: Nearly complete display, includes running headers and footers at the top and bottom of each page. Page breaks always automatically appear in the correct position.

Word: Does not show running headers and footers on each page; page breaks displayed only on command and manual re-display is necessary after changes.

Jazz: Same as MacWrite.

File types

MacWrite: Uses its own compressed file type to save disk storage space. Can read and write standard ASCII files.

Word: Uses its own file type. Can read and write ASCII files as well as MacWrite files, including formatting information.

Jazz: Uses its own file type. Cannot read ASCII files directly, only through the Clipboard and Scrapbook.

Learning time

MacWrite: About an hour for basic functions.

Word: Two or three hours for basic functions; if you know MacWrite, 15 minutes. Complex functions take more time.

Jazz: Starting from scratch, about an hour; if you know MacWrite, two minutes.

Additional features

Microsoft Word includes many features not in MacWrite or Jazz:

- A glossary function that lets you type short codes in your text in place of frequently occurring words or phrases. You can, for example, define a glossary entry so that every time you type *MC* and Command-Backspace, Word expands it to *Macintosh.*
- Footnotes that can be placed automatically at the end of text or along the bottom of the page.
- You can set any font size from 4 to 127 points; you are not limited to the sizes installed in the System file.
- Multiple printing formats, including multicolumn printing and adjustable gutter margins (text on odd- and even-numbered pages shifts to the right and left, respectively, to allow for binding). You can format at three different levels in Word: individual characters, paragraphs, and divisions. You can define a division to be any size; for example, the introduction, body, and appendices of a book can each have a distinct format.
- Form letters can be generated with a separate mailing list or Microsoft File; the function can perform conditional assembly-checking for a name or numeric value before printing a particular sentence.
- Additional keyboard support: Word has many functions defined as keystroke combinations; it also makes use of the numeric keypad for cursor-control keys and function keys.
- Word can drive common daisy-wheel printers directly without special driver software.
- With a conversion utility program, Word can read files, including formattting information, produced with versions of Word on MS-DOS and UNIX/XENIX computers.

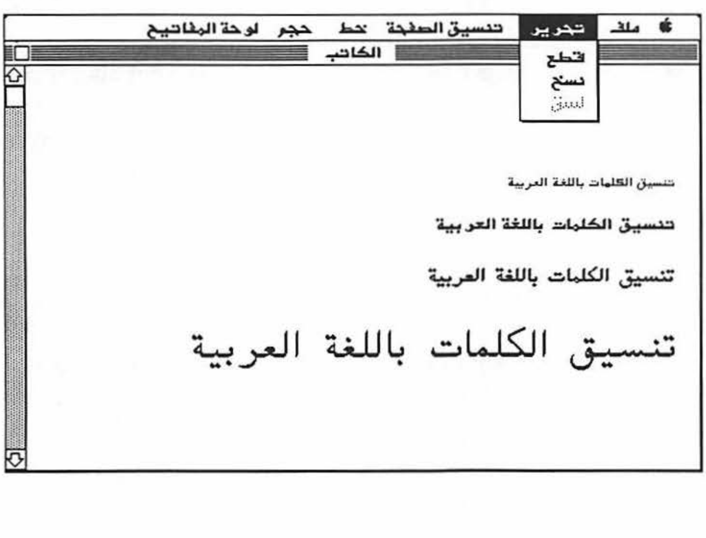
OTHER WORD PROCESSORS AND RELATED PROGRAMS

MockWrite (version 4 from CE Software; available from user groups) is a simple text editor that runs as a desk accessory; you can use it at any time, while using another program. It's much more useful than the Note Pad accessory that comes with the Macintosh; files created by MockWrite can be read as ASCII text files by other word-processing programs.

Foreign-Language Word Processors

For European languages and other alphabetic languages rendered in left-to-right, top-to-bottom form, any Macintosh word processor will work. The standard fonts include the diacritical marks necessary for many European languages. Some languages will need a slightly modified font (you can add them yourself with a font editor, see Chapter 15); Russian and Bulgarian will need a Cyrillic font, which is available from several companies. Fonts for other scripts will appear or you can create your own if you have the skill and patience.

For languages that read right-to-left, the standard word processors are nearly impossible to use. AlKaatib (Arabic Software Associates, Inc., Provo, UT) not only does right-to-left word processing but handles the Arabic character set appropriately, changing a letter's shape according to its position at the beginning, middle, or end of a word, and forming ligatures (connected letters) as needed. Although supplied with a font for Arabic and Persian, AlKaatib can process such languages as Hebrew, Urdu, Pashto, and Sindhi with suitable fonts.



Ergo Soft (Tokyo) has announced a Japanese kanji word processor, EgWord. You enter the words on the keyboard in kana or romaji and the software offers the likely matching kanji characters, which you select with the keyboard or the mouse.

Scientific Word Processors

Conventional programs cannot cope with the specialized needs of scientific or mathematical word processing. You need to be able to pick up and move individual characters, create special symbols, and overlap those symbols to build formulas and equations. You should also be able to define a set of symbols as a block and move all of them together.

MacAuthor (pre-release; from Icon Technology, Leicester, England) is a full-function word processor with special features for scientific and academic use. According to announced specifications, you can design your own symbols, up to a one-inch square. A nondestructive backspace option lets you overlap two or more characters. Text and pictures can be on the same line; the program will run (or wrap) the text around the picture. You can define formats for any portion of text, recalling them with command keys. The MacAuthor file format is not compatible with other word processors but the program can read and write standard ASCII text.

Fonts with scientific, mathematical, and other specialized symbols are widely available from many sources including Apple; you should invest in one if you use these symbols often. The subscript and superscript functions in MacWrite and Word come in handy but you can't put a superscript and a subscript on top of one another (example: x_i^2). You could adjust the characters with a font-editor program (for example, put accent marks over letters), but this is not very flexible. MacWrite is somewhat less convenient than Word, since it always puts at least one line of pixels between lines of characters. You can also create the occasional equation with MacPaint or MacDraw and then move it to the word processor via the Clipboard or Scrapbook.

Outline Processors

Think Tank, from Living Videotext, deals not with long, linear documents but with outlines.

Any word processor can deal with an outline, of course. But word processors can only show a screenful at a time; most of the time you can see details in only one part of the outline—only the trees in the midst of a forest.

Think Tank tracks the relative importance of outline entries by the indentation level. The program preferentially keeps the main headings—the most important entries—on the screen. When you look at an outline, you first see only main headings. You can then select any heading and expand it, showing its subheadings. And, in turn, you can select a subheading and expand it to show the sub-subheadings, and so on. Because only the entries you select are expanded, the major entries remain on screen. In effect, you can always see both the forest and the nearby trees.

Think Tank comes in two versions, one for the 128-KB and one for the 512-KB Macintosh. The 512-KB version adds the ability to include pictures and graphics in the outline and has more traditional word-processing functions. Think Tank uses nonstandard window scrolling, a nuisance.

Spelling Checkers

Spelling programs compare your text against a stored word list and flag all mismatches. Spellers generally come in two forms. Literal spellers use a word list containing only complete words; singular and plural forms are stored separately. Root-word spellers combine root words with many standard prefixes and suffixes; plurals are constructed by adding -s or -es to the root words. Root-word spellers take up much less storage space but will not catch many wrong words, such as *sheeps.*

Spellers cannot check for homonym errors (*peace of cake* for *piece of cake*) or for grammar, so you should never trust a speller to find all errors; you must proofread the final text. Spellers ought to find doubled words (*the the*) but most can't.

Spellers are designed to work with specific word-processing programs; the one you choose must not disturb or be disturbed by the formatting information in your files. Most spellers will let you add words—such as a specialized vocabulary or names you use often—to the spelling list. Some will help you with corrections, displaying close matches to mismatched words and saving you the trouble of looking up possible spellings. In general, the larger the spelling list, the more useful—but only up to a point: Immense spelling lists will include so many archaic and obscure words that some typos will be considered valid words.

The early Macintosh spelling programs are not impressive. Hayden: Speller (version 1.00) can check both MacWrite and Microsoft Word files and you can edit the dictionary with the word processor. The speller works as a freestanding program, separate from the word processor, so you cannot switch quickly between editing and checking spelling; during checking you see the suspect word and the previous line. Assimilation's Mac Spell Right (first released version) is a root-word speller specific to the word-processing program and includes a simple thesaurus function. Mac Spell Right works within MacWrite so you can always see the full context and switch to normal editing at any time, but its spelling list is inferior to Hayden's.

Grammar Checkers

Grammar-checking programs are still primitive. They look for a fixed set of common problems—stock phrases indicating wordy prose, doubled words, or sexist phrases. If you write *in view of the fact that,* the program might suggest *because.* Although simple-minded in design, such programs can help clean up poor writing. A perceptive user can even extract value from a simple word-frequency analysis. In typical business writing, for example, *of* is more common than *and*; good writing reverses the order.

Much good writing bends the rules, however; grammar programs won't improve Shakespeare. Nevertheless, combining a grammar program's analysis with common sense will usually help. You'll soon learn to avoid many simple grammatical errors and can concentrate on making complicated errors that no software can detect.

Language-Translation Programs

Programs that assist translation from one language to another are beginning to appear. At this stage, these programs can only look up word equivalents and do rudimentary sentence analysis; all require the services of a skilled human translator to produce a finished product. Nevertheless, these programs can speed up a translator's work, particularly when specialized vocabularies are involved. True translating programs that can give intelligible results on their own through syntactic and semantic analysis are still some years away, though.

Graphics Software Chapter 4

MacPaint and MacDraw, Apple's two main graphics programs, take different approaches to creating graphics and are best suited to different tasks. MacPaint is unstructured, creating and storing the picture as a collection of pixels (or picture elements); a line in MacPaint is a group of independent pixels, so you can change one pixel without affecting the others. MacDraw, in contrast, is structured and stores attributes. In other words, it stores information that places a line of a particular length and thickness at a certain location with a certain orientation; when you make a change, you are changing the line as a whole. In MacDraw one picture component can overlie another and you can alter one without affecting the other. Thus MacPaint is excellent for free-form drawing, but MacDraw is better for precise design work such as planning a room layout.

Objects created with either MacPaint or MacDraw can be incorporated via the Clipboard or Scrapbook in pictures created with the other. On a 512 KB Mac you can even run both programs at the same time with Apple's Switcher utility (see Chapter 9). This chapter is based on MacPaint version 1.5 and MacDraw version 1.7.

Note: For business graphs like pie charts and plotting data, see Chapter 5; for graphic arts, see Chapter 17.

MACPAINT

The superb MacPaint program by Bill Atkinson is the essence of Macintosh. You can use it to create designs—your company logo, for example—that you can incorporate into documents created by other programs. You can print your logo along with the text you write with your word-processing program or create an illustration to dress up a business chart. Conversely, with suitable techniques or accessories, you can turn a graphics image from virtually any source into a MacPaint drawing.

What's more, MacPaint is fun. Even people with no tolerance for computer games find MacPaint absorbing and rewarding.

Starting the Program

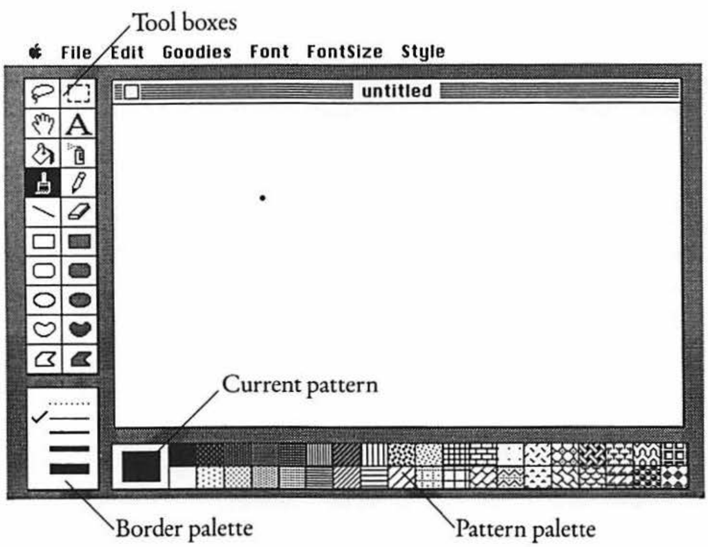
Insert the MacPaint disk.

Double-click the disk icon to open the disk window (if necessary).

Double-click the MacPaint icon; *or*

Click the MacPaint icon and choose **Open** from the **File** menu.

The screen turns grey, with the word MacPaint replacing the desktop menu selections. After a few seconds, MacPaint appears and gives you a blank drawing window labeled Untitled. You'll give it a title when you are ready to save your drawing on your disk.



The boxes on the left contain several drawing tools—a pencil, a paintbrush, an eraser, a can of paint, and so on, plus a number of open and filled shapes.

The border palette at the lower left shows lines of different thicknesses; you select these to control the border width of the open and filled shapes.

The boxes along the bottom make up the pattern palette. The current pattern selected from the palette is shown in the large leftmost box. We'll see how these patterns are used in a moment.

Starting Out

The best way to learn what each tool does is to experiment. Some tools are self-explanatory: The pencil lets you draw free-form; the paintbrush paints with lines and patterns; and the eraser erases.

Click the pencil to select it from the tools.

You can select any tool by clicking it.

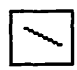
Move the pointer to the drawing window.

The pointer changes shape in the drawing window depending on which tool you're using. Now it's a pencil.

Drag the pencil to draw something—anything.

When you've finished drawing, release the button. Drag the pencil again to draw another line.

Click the straight line in the boxes to the left of the drawing window.



Move the pointer into the window.

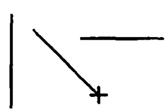
The pointer changes shape to a crosshair.

Drag anywhere in the window and release the mouse button.

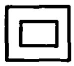
The line tool permits you to draw only straight lines (at some angles the line is a little bumpy because each line is made up of discrete points).

Press Shift and drag another line.

The Shift key is a "constrain" key; when it is depressed, you can only draw vertical, horizontal, and diagonal (45-degree angle) lines. Any line you draw will automatically be adjusted to one of these three orientations.



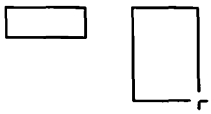
Drawing predefined shapes



Click the open rectangle in the center left tool box.

Point anywhere in the drawing window and drag to another point.

Release the mouse button.



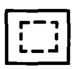
The starting and finishing points of the dragging action mark the diagonal corners of the rectangle.

Similarly, you can select other open shapes and indicate what size they should be by dragging.

If you press the Shift key while you drag the rectangle, it will be drawn as a square; pressing the Shift key while dragging the ellipse will draw a circle.

Moving part of the drawing

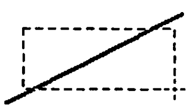
You can move any part of your drawing if you select it first.



Click the dotted-line rectangle in the top tool box.

The dotted-line rectangle is called the selection rectangle.

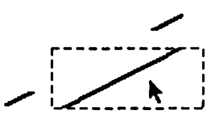
Move the pointer into the drawing window.



Drag the selection rectangle so it surrounds the part of your drawing you want to move.

Position the pointer inside the selection rectangle.

Drag the rectangle and its contents to the new location.



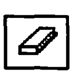
Release the mouse button.

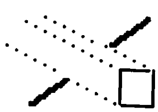
This action deposits the rectangle's contents in the new location.

Click in a blank area, or choose another tool to get rid of the selection rectangle.

Erasing part of the drawing

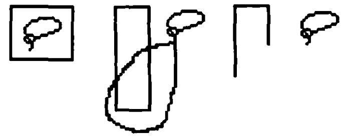
There are four ways to erase part of a drawing.





Click the eraser icon, and drag it over what you want to erase; *or*

Select an area using the lasso (in the top left tool box) or the selection rectangle and hit Backspace; *or*



Select an area with the lasso or selection rectangle and choose **Clear** from the **Edit** menu; *or*

Choose the white pattern and use the paintbrush to paint over what you wish to erase.

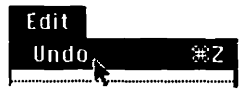
Erasing the whole drawing

Getting rid of your doodles is easy; simply:

Double-click the eraser.

Undoing the changes

You can always undo your last step—but only the last step.



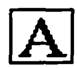
Choose **Undo** from the **Edit** menu; *or*

Type the ` (top left) key.

Adding text to the drawing

You can add titles, labels, and captions to your drawings simply and quickly with MacPaint.

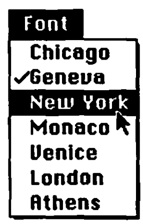
Click the letter A to select it from the tools.

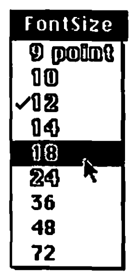


Choose **New York** from the **Font** menu.

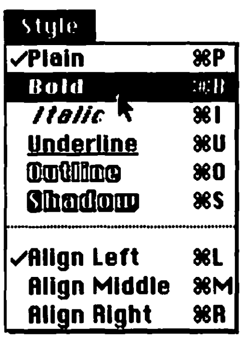
Choose **18 point** from the **FontSize** menu.

Points are a measure of type size; ordinary printed text is generally 10 or 12 point.





Choose **Bold** from the **Style** menu.



New York
18 point
bold

Click an open area in the drawing window.

Type a title or a label.

You can change the font and size by making a different choice from the menus.

Moving to Another Part of the Page

The drawing window shows you only part of the whole drawing page, which is 7⅝- by 9½ inches. If you fill the drawing window and want to move to another portion of the page:

Click the hand icon.

Drag the hand in the drawing window.

If you drag the hand upward, the drawing moves upward, exposing a fresh portion of the page below it; if you drag the hand downward, the window moves downward, exposing a fresh portion above it; and so on.

On large drawings, it works best to move the drawing half a window at a time.

On a 128K Mac it takes a little time to bring a fresh portion of the page into view, so the wristwatch icon appears to tell you to be patient while Mac finishes its task.

Learning to Paint

MacPaint's tool boxes contain three tools for painting: a paintbrush, a regular can of paint, and a spray can of paint. We'll experiment with them here.

Click the paintbrush.

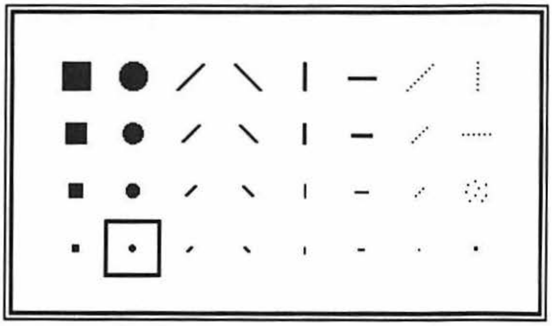
Move the pointer to the drawing window and drag the paintbrush.

To select a brush of a particular size and shape:

Double-click the paintbrush; *or*

Choose **Brush Shape** from the **Goodies** menu.

A window appears with the size and shape of the current brush outlined by a box.



Click any brush size or shape.

You cannot continue painting until you make a choice; if you don't want to change the brush, you must click the size and shape already selected in the box.

When you click a size and shape, the window disappears.

Drag the paintbrush in the drawing window.

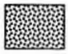
The brush paints with the current pattern.

Using the MacPaint palettes

Choose another pattern from the palette along the bottom of the screen.

Choose the filled rectangle from the tools to the left.

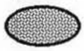
Move the pointer to the drawing window.



Drag the pointer diagonally across the window and admire the results.

Choose a different border width from the border palette.

Drag across the drawing window again.



Choose a different shape (for example, the filled oval) and again drag the pointer.

Experiment with different tools and patterns; try the paintbrush and spray can, using different patterns.

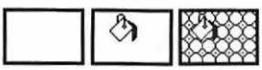
Filling an area with a pattern

Choose the open rectangle from the tools.

Draw a rectangle.

Choose the paint can.

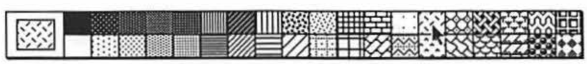
Move the paint can into the rectangle and click.

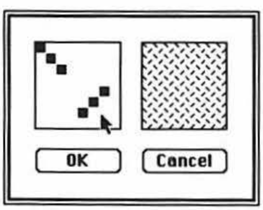


If the white pattern is selected, nothing will happen; click another pattern and try again.

Creating a new pattern

Double-click a pattern that is close to the one you'd like to create.





A dialog box comes up with the pattern detail greatly magnified on the left and the overall pattern on the right.

Click individual squares of the pattern in the left box.

Each individual square represents a dot on the screen. Clicking individual squares changes their color; black squares become white, white ones become black. The box on the right shows what the overall pattern looks like when you change individual dots. When you are satisfied with the new pattern:

Click the OK button.

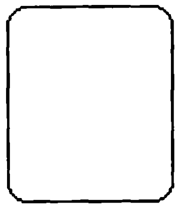
When you click OK, the modified pattern replaces the pattern you originally double-clicked. The pattern change is saved with the drawing on disk.

If you change your mind, click Cancel.

A Macintosh Self-Portrait

Let's put some of these techniques together and have the Macintosh draw a self-portrait. (Whether you, the hardware, or the software is actually doing the drawing is a semantic question I will ignore.)

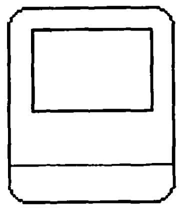
Double click the eraser to clear the window.



Choose the rounded-corner open rectangle from the tools, and the second border from the border palette.

Move the pointer to the drawing window and drag from top left to bottom right to form the body of the Macintosh.

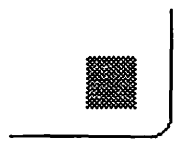
Draw the screen the same way, using a smaller, square-corner open rectangle.



Choose the straight-line icon from the tools.

Hold down the Shift key while dragging a horizontal line to indicate Mac's recessed lower front face.

Choose the filled rectangle.

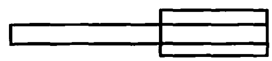


Choose a uniform gray dot pattern.

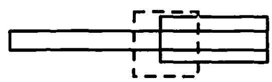
Drag to form the keyboard connector.

Choose the open rectangle.

Drag and build the disk-drive slot from two rectangles.



Using the selection rectangle (top right tool box), select the area where the two disk-drive rectangles overlap.

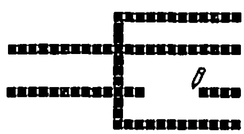


Choose **FatBits** from the **Goodies** menu.

FatBits shows you a magnified portion of the drawing, surrounded by the flickering selection rectangle. Each black square represents a dot on the screen. You need to remove some dots to create the shape you want.

Choose the pencil from the tool boxes.

Move the pencil into the drawing window and click on each dot you wish to remove; drag the pencil to remove several dots at a time.



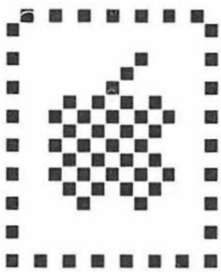
Clicking the pencil changes a square from black to white or white to black; if you click in a white area, you create dots. If you take out too many dots, use the pencil to put them back in.

The results appear in proper scale in the inset at the upper left corner of the drawing window.

Click in the inset to restore the original drawing window.

Choose a suitable pattern for Mac's screen—horizontal lines, perhaps.

Choose the paint can, move it to the rectangle that represents the screen, and click to fill in the screen.



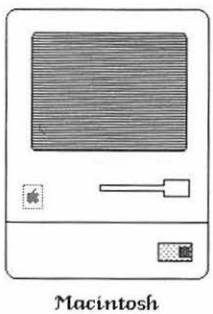
For some finishing touches:

Choose the filled rectangle and drag a small rectangle for Mac's Apple logo.

To make the Apple logo:

With the selection rectangle, surround the Apple logo and choose **FatBits** from the **Goodies** menu.

Use the pencil to remove dots so you end up with something that looks like an apple.



Label the drawing; try the Venice font at 14 point:

Choose **Venice** from the **Fonts** menu.

Choose **14** from the **FontSize** menu.

Choose **Plain** from the **Style** menu.

Select a location and type in the label.

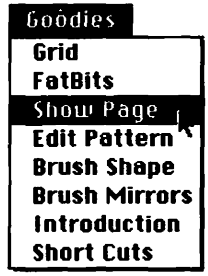
One last touch. If something isn't positioned just right:

Choose the selection rectangle from the tool boxes.

Surround the misplaced part of your drawing.

Use the pointer to drag the surrounded part of your drawing to precisely where you want it.

Click in a blank area to clear the selection rectangle.



Choose **Show Page** from the **Goodies** menu to see where your drawing will appear on the 8½- by 11-inch page.

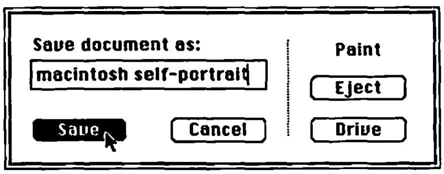
The dotted rectangle represents your drawing window.

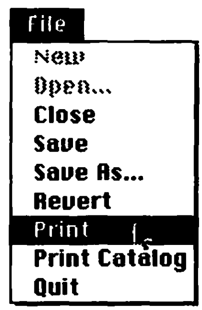
Press the mouse button inside the drawing window and drag to move it anywhere you want it on the page.

Dragging anywhere else will reposition your drawing on the page. The printed version will match your final choice.

Choose **Save As...** from the **File** menu to name and save your handiwork.

Type in an appropriate name and click the Save box.





Your Macintosh self-portrait will be saved as a file and will be given a special icon that will appear in your disk window on the desktop.

Choose **Print** from the **File** menu to print your picture.

What you see in the MacPaint drawing window is only about a third of a single 8½- by 11-inch page. What you get when you print is the whole page. If you've been experimenting, you may have miscellaneous doodling on parts of the page outside the drawing window.

Notes on MacPaint

The active area of a MacPaint drawing window is 3⅜ inches high by 5½ inches wide (14.2 by 8.6 cm). The active area can be moved anywhere in a 7⅝- by 9½-inch area on an 8½- by 11-inch page, so a full page of artwork can be created in separate sections but printed out as a complete page.

One limitation in MacPaint is the size of the image you can select—only as large as a MacPaint active window. Sometimes you need to select a larger area, such as for a wide letterhead. The supporting software for the ThunderScan image scanner and Paint Cutter (Silicon Beach Software) lets you make such large selections.

For the best results, draw MacPaint images in the correct size from the beginning. MacPaint lets you resize an object along the horizontal or vertical axis, or both, but the resizing process distorts the image. If you are making an object larger, the program at times doubles the number of pixels to maintain the same relative weight; conversely, it must delete dots if you reduce the size.

When you save pictures in MacPaint, each file includes some overhead information. To save disk space, combine pictures into a single file when possible.

For some enhancements to MacPaint, see Table 4-1.

TABLE 4-1. GRAPHICS-SOFTWARE PRODUCTS

MacPaint Accessories

ClickArt Effects: From T/Maker; a clever desk accessory, works from within MacPaint and lets you rotate any selected object through an angle of your choice (MacPaint only allows 90-degree rotations); can also create the appearance of a perspective view and can distort objects as if they were on a rubber sheet.

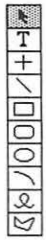
ColorPrint (version 1.1): From Esoft Enterprises; lets you print color pictures from MacPaint in multiple passes through the printer with color ribbons; first, create the complete drawing, then set up each pass by selectively deleting portions of the image, leaving just the portion for each specific color; change ribbons for each pass; ColorPrint backs up the paper in the Imagewriter when required.

ColorMate (pre-release): From SoftStyle; initially sold by NEC for its color Pinwriter printer; works with color printers, which use multiple-part color ribbons, and can print the finished page in one pass, without fuss. The program works separately from MacPaint; create the complete image first with MacPaint, then assign colors with the SoftStyle program, which lets you selectively display each color as black or white on the screen.

Iron-on transfers: Put MacPaint images onto T-shirts with ribbons that print iron-on transfer images on paper; available from Diversions, Inc. (Belmont, CA) and I/O Design (Rumson, NJ).

MACDRAW

If you have experience with MacPaint, you won't have much trouble learning MacDraw. The tools are at the left side of the screen, but the line weights are under the Lines menu and the patterns are found under the Fill menu.



You select objects rather than regions, and you can overlap objects. Bring to Front and Send to Back under the Arrange menu let you choose which objects are on top and visible. If you have trouble selecting an object, you can move obscuring objects temporarily. You can treat a group of objects as a unit by selecting each one individually (hold the Shift key down as you select them) and choosing Group from the Arrange menu; you can also Ungroup the objects. To change the position of any object, you click on the object so it is marked by small black squares, and then drag the object to where you want it.

The MacDraw window scrolls in the usual way; you can have up to four windows open at a time and can cut and paste between the windows.

Here is how to draw a Macintosh with MacDraw:

Choose Turn Grid Off from the Layout menu.

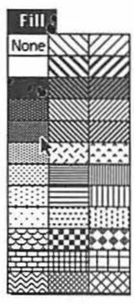
This gives you more freedom in positioning objects.

Choose the rounded-corner rectangle tool and draw out a suitable shape.

The selected tool automatically switches back to the selection arrow after you use other tools.

Choose the square-corner rectangle tool and draw the screen.

Choose the straight-line tool and drag a horizontal line to indicate Mac's recessed lower front face.

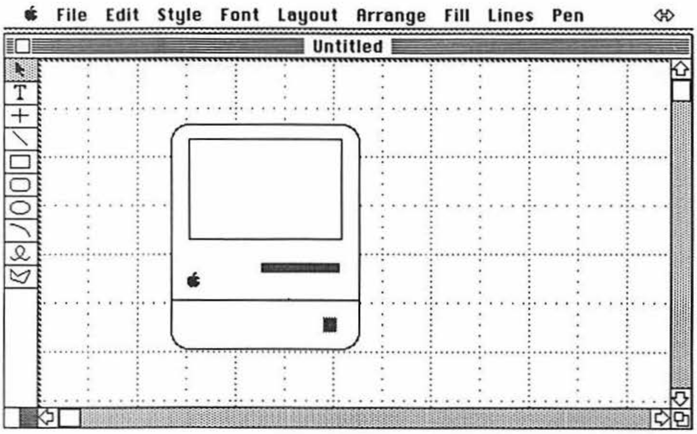


Choose the square-corner rectangle tool to draw the keyboard-connector outline and then choose a suitable pattern from the Fill menu.

To get rid of the connector outline, choose the dotted line from the Lines menu.

Creating a disk-drive slot that looks the same as the one created with MacPaint is difficult. You have to create each line individually and then group them, so fudge the slot with a single rectangle and fill it with a dark pattern; you could also create the slot in MacPaint and move it to MacDraw.

MacDraw has no FatBits function, so creating details like the Apple logo is impossible. But you can go back to the MacPaint picture, copy the logo to the Clipboard, and paste it into the MacDraw picture.



Notes on MacDraw

Unlike MacPaint, MacDraw can scale drawings smoothly because it recalculates each element from basic attributes. MacDraw can also create large, multiple-page drawings.

Although MacDraw deals with objects effectively, it lacks some features necessary for precise design work; supplementary software may add these features to MacDraw and other programs under development will include them.

Structured graphics, such as graphs from other programs, can usually be transferred to MacDraw via the Clipboard or Scrapbook. Generally the pasted image appears with all objects selected. You can immediately group all the objects into a single entity or you can group them selectively.

MacDraw objects moved to MacPaint lose their structures and become collections of pixels. If you move a MacPaint picture to MacDraw, you will no longer be able to manipulate picture components; the entire picture becomes a single MacDraw object. (If created by early versions of MacPaint, large objects may be split into several bands; if so, choose Group from the Arrange menu to keep the image together.)

For more professional design work, MacDraft (pre-release, from Innovative Data Designs) is similar to MacDraw but contains many additional features for precision design—you can zoom in and out, rotate objects by 1-degree increments on-screen, and attach numerical values to an object's size.

PRINTING FROM MACPAINT AND MACDRAW

In standard-resolution printing on the Imagewriter, both MacPaint (Print Draft) and MacDraw generate exact pixel-for-pixel correspondence between the screen and printer. In the high-resolution mode on the Imagewriter, MacPaint (Print Final) interpolates pixels with an algorithm (group of instructions). The algorithm examines adjacent pixels; if they are in a vertical or horizontal line, additional dots are filled in. Since this won't always be what you want, you will have to experiment.

When you print MacPaint images on the LaserWriter, Print Final performs 45-degree interpolation to smooth the corners between diagonally adjacent pixels. If the MacPaint picture is included in a MacWrite or other program document, selecting Smoothing in the Page Setup dialog box adds the interpolation. MacDraw documents can be printed smoothly at higher than screen resolution, since interpolation of the pixels is based on the drawing attributes.

Spreadsheets and Graphs

Chapter 5

Spreadsheet programs are the most popular business application for microcomputers. They are powerful tools for calculation and projection. Graphing programs take numeric input, either from another program, such as a spreadsheet, or from the keyboard, and generate charts in standard formats—bar graphs, pie charts, line graphs, and so on.

USING SPREADSHEETS

Suppose you want to calculate the difference in cost between two new cars. One model has better fuel economy but a higher price. Is it cheaper in the long run to buy this model than one that consumes more fuel but costs less? To do this calculation carefully without a computer, you would take out a sheet of ledger paper and systematically enter each cost item. To get a total, you must make many assumptions about the cost of gasoline, how many miles you expect to drive each year, the interest rate on the car loan, and so on. Since you can't be sure how many miles a year you will drive, you have to repeat the entire calculation several times for high, medium, and low estimates. The process is so laborious that even an accountant would stop calculating and start guessing after one or two repetitions.

You can set up the same analysis with a spreadsheet. A spreadsheet works like a ledger sheet; it is organized in rows and columns. You can link the value of any entry to that of any other entry with a mathematical or logical formula. Once you have entered the values, the computer performs all the calculations. For a single calculation, a spreadsheet may not be any faster than a calculator. But for repeated calculations with changing values, called variables, the spreadsheet shines. If you change any number—the number of miles you drive each year, for example—the rest of the calculation is performed automatically. The process is so much easier with a spreadsheet that you can run dozens or hundreds of calculations painlessly. In the process, you will come to learn the true cost of owning a car—where the money really goes and which costs are essentially irrelevant to your comparison.

Spreadsheets give you the power you need to plan ahead in your business, too. For example, you can construct a spreadsheet model of your income and expenses, using formulas to add tax and shipping automatically to each purchased item and to calculate discounts for volume purchases.

Spreadsheets aren't just for calculations, though. If you need to produce an organization chart or other textual material that is arranged in columns and rows, a spreadsheet works much better than a word processor.

Macintosh Spreadsheets

On the Macintosh, you can choose among a half dozen spreadsheet programs. First, I will illustrate a spreadsheet application using Microsoft's Multiplan spreadsheet. With minor changes, these instructions also apply to the other spreadsheets.

If you have used a spreadsheet program on another computer, you will find that the Macintosh spreadsheets are a revelation. To begin with, the screen is far easier to read. Instead of typing endless keystrokes to perform such simple tasks as changing a column width, you simply point at the line dividing two columns, grab it with your mouse, and drag the column wider or narrower. If you want to have two windows showing different parts of the spreadsheet, you grab a split bar, and drag the column until the windows are set to your satisfaction. It's a lot easier to do than to describe. A conventional spreadsheet on an ordinary computer is archaic and clumsy by comparison.

Microsoft Multiplan

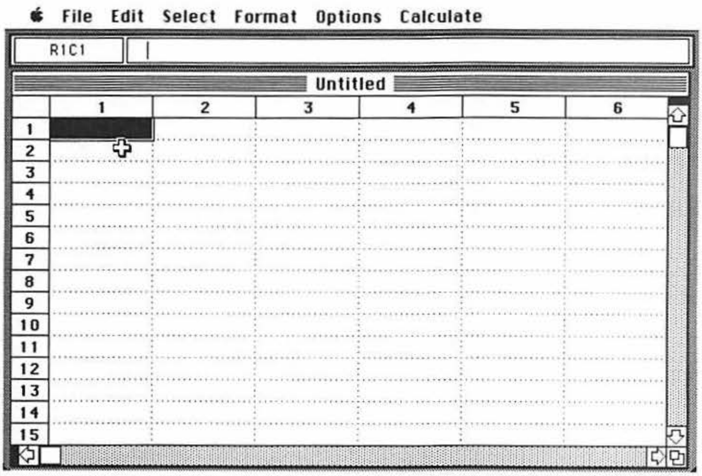
In this Multiplan profile, we will build a spreadsheet model to analyze the cost of buying and owning a car. Although building the model takes many steps, you should be able to have it running in 20 minutes.

I've included the complete model because it shows the power of a spreadsheet, and because the results are interesting; most car owners don't really know where their money goes.

Starting Multiplan

Put the Multiplan disk into the Macintosh.

Double-click the Multiplan icon.



An Untitled window opens up, displaying an array of cells. The pointer is shaped like a fat cross when moving over the cells.

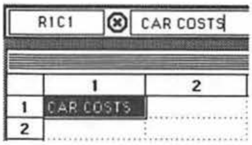
Click any cell.

The cell color reverses. Clicking a cell selects that cell.

Each cell has an address: a row number down the left side and a column number across the top; R1C1, for example, is the address of the cell at the intersection of row 1 and column 1. The selected cell's address is displayed in the formula bar along the top of the window. As you click on different cells, the cell address in the formula bar changes.

The formula bar displays not only the cell address but also the contents of the current cell.

Entering labels



Click the first cell at the upper left (address: R1C1).

Type CAR COSTS.

The label appears in the first cell as you type it. If you make a mistake, backspace and retype.

Press the Return key.

The next cell down (R2C1) becomes the active cell.

Press the Return key again.

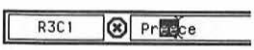
This step moves the active cell down one more (R3C1). You could also have just clicked the next cell down.

Type Price and press Return.

If you make a mistake but don't notice it until after you have pressed Return, you can go back and correct the mistake in the usual Macintosh way:

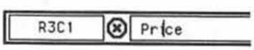
Click the cell with the mistake.

The formula bar at the top shows the active cell's contents.



Drag through all or part of the contents to select it.

Delete the selection by pressing the Backspace key; *or*



Type a new entry.

Another method: Undo your last step.

Choose **Undo Typing** from the **Edit** menu.

If necessary, you can always retrace your steps. For a more complete guide, see the Multiplan manual.

And now, back to entering labels:

Click cell R4C1.

Type Down Payment and press Return.

Continue with:

Loan Value

Monthly Payment

1st Yr Cost

2nd-3rd Yr Cost

Fixed Costs

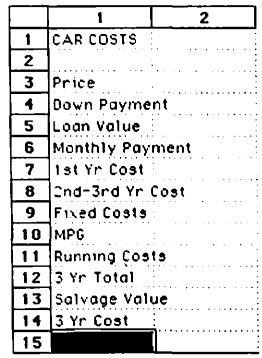
MPG

Running Costs

3 Yr Total

Salvage Value

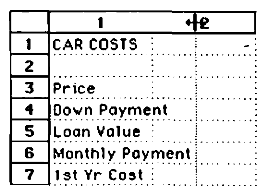
3 Yr Cost



After each entry, press Return.

Some of the labels may be a little wider than the column. Until you enter something in the cell to the right, you can still read the entire cell's contents; however, entries in column 2 may partially obscure long labels in column 1, so let's widen column 1 now:

Point at the line just above the first row dividing columns 1 and 2.



The pointer changes to a line with arrows.

Drag the pointer to the right until the column division clears all the labels.

Dragging to the right makes the column wider; dragging to the left makes it narrower. All of the other columns move right or left, their widths unchanged.

Click the first cell in the second column (R1C2).

Type Gas Cost and press the Tab key.

If you press Tab rather than Return after the entry, the active cell moves to the right.

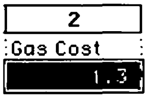
Type Miles Yr and press Tab.

Type Interest and press Tab.

Type Loan Term and press Return.

Entering variables

Click the cell under Gas Cost (R2C2).

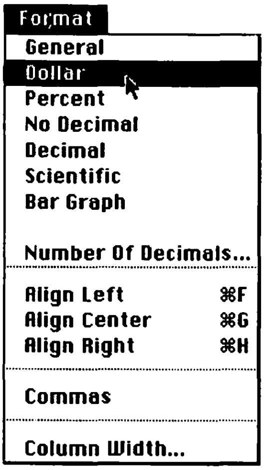


Type the cost of gasoline, say 1.30, and press Enter.

The entry appears as 1.3. Since most of this spreadsheet will show dollar values, you can change the format of column 2 to display the amount in dollars:

Select the entire column of cells by clicking the column heading.

The entire column is highlighted.



Choose **Dollar** from the **Format** menu.

Now all numeric values in the second column will appear with a dollar sign and two decimal places.

Click the cell under Miles Yr (R2C3).

Type 12000 and press Tab.

We'll use 12,000 miles driven in a year as a starting point. Because this cell was not selected for the dollar format, there is no dollar sign.

Pressing Tab moved the selection to R2C4, the cell under Interest. To enter the interest rate:

Type 0.14 and press Enter.

0.14 is 14 percent interest. As with the other variables, we can change the loan interest later. To display it in percent format, make sure that R2C4 is still the selected cell (pressing Enter doesn't move the selected cell):

Choose **Percent** from the **Format** menu.

The entry now reads 14.00%.

Click the cell under Loan Term (R2C5).

Type 36 and press Enter.

This sets a three-year, or 36-month, loan term.

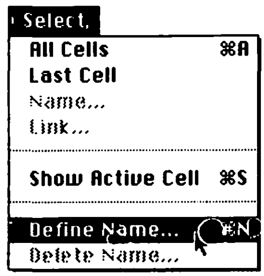
	1	2	3	4	5	6
1	CAR COSTS	Gas Cost	Miles Yr	Interest	Loan Term	
2		$1.30	12000	14.00%	36	
3	Price					

Naming variables

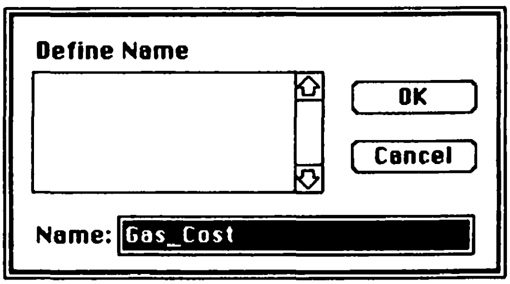
For convenience, name the four variables you have just entered. You can then build up formulas with easy-to-understand names instead of cell addresses like R2C2.

Click the cell containing $1.30 (R2C2).

Choose **Define Name...** from the **Select** menu.



A dialog box asks you for the name of this cell. Multiplan proposes the name Gas_Cost, so you don't have to type it. When you choose Define Name for a particular cell, the program will automatically look at adjacent cells for possible names.



Press Enter.

This names the cell value. Pressing Enter does the same thing as clicking the OK button in the dialog box.

Follow the same procedure to define names for the cells containing 12000 (R2C3), 14.00% (R2C4,) and 36 (R2C5). Now you can start building up the formulas for the main calculation.

Entering formulas

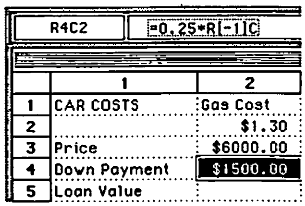
Click the cell to the right of Price (R3C2).

Type 6000 and press Return.

$6000 is the price of an economy car.

The active cell is now R4C2, the cell to the right of Down Payment. The down payment is a quarter of the price. In the following instructions, use the mouse to perform the action given in [brackets].

Type =.25*[click R3C2, the cell above] and press Enter.



You have told Multiplan to multiply (computers use * instead of × to avoid confusion with the letter X) the price in R3C2 ($6000) by 0.25 and put the result ($1500) in R4C2.

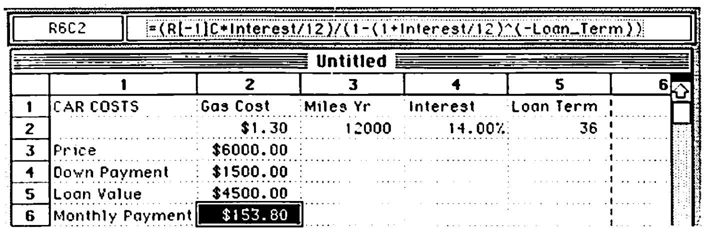
Click the cell to the right of Loan Value (R5C2).

For the loan value, subtract the down payment from the price:

Type =[click R3C2, price]−[click R4C2, down payment] and press Return.

The monthly payment formula is the most complex:

Type =([click R5C2, loan value]*Interest/12)/(1−(1+interest /12)^(−Loan_Term)) and press Enter.



This is the standard formula for monthly loan payments; your bank may use a slightly different one. The ^ symbol indicates an exponent.

Check the entry for accuracy.

The formula bar should show:

=(R[−1]C*Interest/12)/(1−(1+Interest/12)^(−Loan_Term))

If you put in spaces around the symbols, Multiplan removes them. The R[−1]C is the address of the loan value cell you clicked relative to the address of the cell where you were entering the formula (up one row, same column).

Edit the line if necessary to correct mistakes.

The first-year cost of the loan is the down payment plus 12 monthly payments:

Click the cell to the right of 1st Yr Cost (R7C2).

Type =[click R4C2, down payment]+12*[click R6C2, monthly payment] and press Return.

If you see ##### in the cell, the column is too narrow. Open it up as described earlier, dragging the divider between columns 2 and 3 to the right.

The selected cell (R8C2) is now the one to the right of the label, 2nd-3rd Yr Cost.

Type =24*[click R6C2, monthly payment] and press Return.

The next item, Fixed Costs, includes car registration, insurance, and maintenance.

Type 800 and press Return.

The next item is MPG—miles per gallon.

Enter 30 and press Enter.

Since miles per gallon isn't a dollar value, you need to delete the $ sign. Make sure cell R10C2 is still selected:

Choose **General** from the **Format** menu.

The cell contents change from $30 to 30.

For Running Costs, click R11C2, then type =Gas_Cost*Miles_Yr/[click R10C2, mpg] and press Return.

For 3 Yr Total, type =[click R7C2, 1st yr cost]+[click R8C2, 2nd-3rd yr cost]+3*[click R9C2, fixed costs]+3*[click R11C2, running costs] and press Return.

	1	2
1	CAR COSTS	Gas Cost
2		$1.30
3	Price	$6000.00
4	Down Payment	$1500.00
5	Loan Value	$4500.00
6	Monthly Payment	$153.80
7	1st Yr Cost	$3345.59
8	2nd-3rd Yr Cost	$3691.18
9	Fixed Costs	$800.00
10	MPG	30
11	Running Costs	$520.00
12	3 Yr Total	$10996.78
13	Salvage Value	$4500.00
14	3 Yr Cost	$6496.78
15		

The next item, Salvage Value, is the expected selling price of the car after three years. A $6000 car might sell for $4500 after three years.

Type 4500 and press Return.

For 3 Yr Cost, type =[click R12C2, 3 yr total]−[click R13C2, salvage value] and press Return.

Now you're ready to see the power of a spreadsheet. You can change any variable—the price of gasoline, the price of the car, the interest rate—and all the affected numbers will change. What if the price of gasoline were to decrease or, more likely, increase dramatically? Click R2C2, gas cost; vary the gas cost from $1.30 to $4.00 and watch how the 3 Yr Cost changes. The total cash outlay for a car at the end of three years (3 Yr Total) is sobering; think how many Macintoshes you could buy.

Duplicating the model

If you want to compare several cars, you can change the price and other entries, but then you won't be able to compare two cars directly because the new entries will erase the old.

The better way to make a comparison is to build up a second column. You could retype everything, but Multiplan lets you duplicate a column quickly.

	1	2	3
1	CAR COSTS	Gas Cost	Miles Yr
2		$4.00	12000
3	Price	$6000.00	
4	Down Payment	$1500.00	
5	Loan Value	$4500.00	
6	Monthly Payment	$153.80	
7	1st Yr Cost	$3345.59	
8	2nd-3rd Yr Cost	$3691.18	
9	Fixed Costs	$800.00	
10	MPG	30	
11	Running Costs	$1600.00	
12	3 Yr Total	$14236.78	
13	Salvage Value	$4500.00	
14	3 Yr Cost	$9736.78	
15			

Drag in the second and third columns starting at R3C2 (price) and going down to R14C3 (3 yr cost).

This selects the cells in column 2 that you want to copy and tells Multiplan where to put them. If you wanted to copy column 2 more than once, you would simply select more columns.

Choose **Fill Right** from the **Edit** menu.

2	3
Gas Cost	Miles Yr
$4.00	12000
$6000.00	$6000.00
$1500.00	$1500.00
$4500.00	$4500.00
$153.80	$153.80
$3345.59	$3345.59
$3691.18	$3691.18
$800.00	$800.00
30	30
$1600.00	$1600.00
$14236.78	$14236.78
$4500.00	$4500.00
$9736.78	$9736.78

Voila! The contents of column 2 are cloned into column 3.

Now change the appropriate figures—price, fixed costs, mpg, and salvage value—for a second car.

You can repeat this duplication for as many columns as you like. You can then see what happens if the price of gasoline goes up to $5.00 per gallon. Or if you drive 30,000 miles a year. Or the interest rate drops to 10 percent. If you are interested in what your present car is costing you, you might as well treat its current value as its price. This strategy adds interest charges to the cost and thus reflects true costs, since if you had the money instead of the car, you could invest the money and earn interest on it.

With the model you've created, you can calculate the real difference between a $6000 car with 30 miles to the gallon and a $9000 car with 40 miles to the gallon. The model will tell you what the break-even point is for holding on to an older gas guzzler and buying a new econobox.

	1	2	3
1	CAR COSTS	Gas Cost	Miles Yr
2		$1.30	12000
3	Price	$6000.00	$9000.00
4	Down Payment	$1500.00	$2250.00
5	Loan Value	$4500.00	$6750.00
6	Monthly Payment	$153.80	$230.70
7	1st Yr Cost	$3345.59	$5018.39
8	2nd-3rd Yr Cost	$3691.18	$5536.78
9	Fixed Costs	$800.00	$1150.00
10	MPG	30	40
11	Running Costs	$520.00	$390.00
12	3 Yr Total	$10996.78	$15175.16
13	Salvage Value	$4500.00	$7000.00
14	3 Yr Cost	$6496.78	$8175.16
15			

To make this model easy to enter, I've kept it all on one screen. To enlarge your model, read the Multiplan manual about how to freeze the labels (titles) and make a split window. You can then elaborate the fixed costs to include license fees, use taxes, and insurance, and have the maintenance costs go up year by year.

CHARTS AND GRAPHS

A spreadsheet generates tables of numbers, the kind we've all seen in annual reports and statistical summaries. But the meanings of numbers are all too often obscured when they are printed in tables.

We can recognize and interpret graphics much more easily than numbers. For the same reason that the icons in the Macintosh Finder work more effectively than the massed text that passes for file directories on other computers, spotting trends or comparing two numbers is much quicker on a graph than it is in a gray column of numbers.

You could use programs like MacPaint or MacDraw to create charts, but those programs cannot take numeric input and automatically turn it into charts. Graphing programs can.

On the Macintosh, most graphing programs are a part of a spreadsheet program; they plot numbers from the spreadsheet. Whether you need its computational power or not, the spreadsheet provides a useful vehicle for entering and checking your numbers before graphing.

The quality of the charts is not only a matter of personal taste. A presentation-quality chart not only pleases the eye but is easier to understand. Different data values must be easy to discriminate; you should be able to add numerical values to a graph for convenient reference. A good graphing program gives you many choices; programs with limited choices force you to accept a few rigid notions of what a graph should look like.

Many spreadsheet programs can accept data from several sources. Microsoft Chart is a freestanding program, able to take information either directly or from Multiplan and some other programs.

COMPARING SPREADSHEETS

The first half-dozen spreadsheet programs for the Macintosh all do at least an adequate job. They are listed here, beginning with the largest spreadsheet size, and summarized in Table 5-1.

Microsoft Excel (version 1). This is among the most powerful spreadsheets developed for computers so far and takes exceptional advantage of the Macintosh interface; it includes comprehensive graphing capabilities and a simple database function. You can define your own functions and consolidate multiple spreadsheets. Excel is the only Macintosh spreadsheet thus far to include a keyboard macro feature, which means you can assign a series of commands and spreadsheet entries to a simple keystroke combination. (A much more limited keyboard macro function is possible for the other spreadsheets if you use an accessory utility program, but there may be compatibility problems; see Chapter 9.)

TABLE 5-1. SPREADSHEET PROGRAMS

Product	Nominal size	Business graphs	Spreadsheet windows	Split window	Linked sheets	Text editor	Number of functions
Excel	16,384 × 256	Good	>64	Yes	Yes	No	85
Crunch	9999 × 250	Fair	6	Yes	Yes	Limited	74
Jazz	8192 × 256	Fair-Good	4	No	No	Yes	93
Quartet	999 × 62	Fair	1	Yes	No	Limited	25
Multiplan	255 × 63	None	1	Yes	Yes	No	42
ClickOn	40 × 25	Limited	1	See note	No	No	28

Nominal size gives the maximum spreadsheet area in rows and columns. You cannot actually fill most spreadsheets with real data up to their nominal size; you will run out of memory first.

Business graphs are linked to the spreadsheet. Multiplan requires Microsoft Chart for graphs (rating: good).

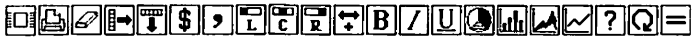
Spreadsheet windows tells how many windows can be open at a time.

Split window is the ability to look at two or more parts of the spreadsheet at the same time.

Linked sheets can be created independently and then merged; programs unable to link sheets can usually accept static information from the Clipboard.

Number of functions tells how many mathematical, statistical, financial, and other functions are available; these numbers are only roughly comparable.

Crunch (version 1.0 from Paladin Software). This program includes graphing, a simple data base, and a 2½-page text editor. Twenty-one icons give you direct access to common operations including formatting and graphing. You can define your own functions, change the text style (plain, italic, bold, underlined) cell by cell, and consolidate spreadsheets. Crunch does not have an Undo command. The graphing function creates the most common types of graph, but you cannot add annotations to your graphs.



The spreadsheet component of Jazz (release 1) from Lotus. Although not as powerful a spreadsheet as Excel, Jazz includes word processing and communications, as well as graphing and a simple data base. Jazz's most outstanding feature is live links to the word processor, but Jazz cannot split a spreadsheet window, an annoying oversight. See Chapter 7 for more information and a comparison with Microsoft products for multiple tasks.

Quartet (version 1.0). This program from Haba Systems includes business graphics, a simple data base, and a rudimentary but useful text editor. The text editor can handle a maximum of 1000 characters at a time—good for annotations but not for word processing. Quartet employs an unusual system with arrow icons for moving around the spreadsheet; this frees some space on screen for the data but makes rapid movement around large spreadsheets awkward. Quartet lacks an Undo command and does not automatically figure out whether you are entering text or a number/formula; you must preface text entries with an apostrophe.

Microsoft Multiplan (version 1.02). The first spreadsheet for the Macintosh, Multiplan still works well even compared to newer competitors. Multiplan's main limitations are its maximum spreadsheet size of 255 rows by 63 columns (although you can consolidate separate files) and the necessity for a separate program, Microsoft Chart, to create graphs.

ClickOn Worksheet (first released version). This is a spreadsheet desk accessory from T/Maker Graphics. Despite the necessary restrictions of a desk accessory, ClickOn manages surprisingly complete features, including simple graphs. For some people, ClickOn may be all the spreadsheet they need; others might use it in addition to a conventional spreadsheet. You cannot, however, move ClickOn spreadsheet formulas to another program. You cannot scroll or split the spreadsheet window or split the screen, but you can overlap columns. ClickOn has manual calculation only and it lacks an Undo command.

Notes on Spreadsheets

All spreadsheets can perform the common functions. The two Microsoft products work in similar ways, but other spreadsheets differ enough in details that you probably should select one program and stick with it. You can move information between some programs, but many specific features will be lost. If you need to move data to or from an IBM PC, see Chapter 27.

Although ordinarily used for numeric calculations, spreadsheets work much more effectively than a word processor for creating tables of text. Moving an entry from one portion of a table to another is difficult with a word processor, but is easy with a spreadsheet. Most spreadsheets distinguish between text and values by the characters you type, but you can force them to treat a number as text; check the manual for details. When you have finished, you can copy the table to the Clipboard and then paste it into a word-processing document.

Business Programs Chapter 6

Microcomputers have been much more successful and useful in business than in the home. This chapter surveys a wide range of programs aimed at business and professional users.

DATA BASES

A data base is an organized collection of information, or data; for example, an address book is a data base. A specific database file is made up of records (such as a complete address), each of which is in turn made up of fields (name, street address, zip code, and so on). You index a data base by a field; most address books are indexed by name, as are telephone directories.

One reason computerized data bases are so powerful is that you can change the index field; you can index on zip codes instead of names when you are preparing a mailing. You can also index on more than one field. And you could select only those records with fields that meet specific criteria; for example, you can request all the Jacksons who live in Chicago.

A database program lets you enter, sort, update, find, and print your data quickly and easily. Although the idea of a data base is simple, database programs are the most complex of the common microcomputer applications. The most powerful go far beyond simple functions, letting you define relationships between data entries, perform statistical and accounting calculations, extract and reorganize the data, and much more.

Traditionally, the problem for the user has been deciding how much power to buy; the more powerful the database program, the more difficult it has been to use. The Macintosh interface has helped, and the best database programs for the Mac are easier to use than their equivalents on other computers, but programs boasting great power are thus far still troublesome and difficult to learn.

Certain database programs have been the poorest of all Macintosh programs so far. Several were converted from earlier versions designed to run on other microcomputers, and these take minimal or no advantage of the Macintosh interface. You should examine a program with special care if it runs on the Macintosh in the same way that it runs on older computers.

Selecting a Data Base

First, the essential requirements; a database program should offer all of these features:

- Field types. The very simplest data bases use only text fields. But if a data base only recognizes text, not numbers, the program cannot perform any calculations on fields. Nearly all programs let you define text and numeric fields, and many also include date fields for time-dependent financial calculations and logic fields for true-false information.
- Data entry. You should be able to edit any text entry just as you do in MacWrite—double-click to select a word, shift-click to extend a selection, and so on. You should be able to select a portion of any entry and where valid cut, copy, or paste it to any other entry (pasting a text entry into a purely numeric field doesn't make sense, of course). For convenience, you should also be able to create a custom form on-screen for data entry.
- Sorting. The program should let you select a field and sort on it. For most alphabetic applications, the sorting should be a dictionary sort, sorting the letters regardless of whether they are upper- or lowercase. For either alphabetic or numeric sorts, you should be able to choose whether you want to sort from lowest to highest or vice versa.
- Queries and reports. You should be able to ask for all records meeting some criteria—for example, every salesperson who sold over 500 widgets in a month.

- Output forms. All database programs should let you organize and print out the information any way you want. You should be able to do the positioning and sizing with the mouse, so that you can change quickly from an address directory to mailing labels. The reports should have an optional six-lines-per-inch mode for printing onto standard mailing labels.
- Print merge option. Information in the data base should be easily combined with text from a word processor for form letters. And more generally, you should be able to use the Clipboard or Scrapbook to move information between the data base and other programs.

Other features are useful to look for but, depending on your needs, they are arguably not essential:

- Graphics. Some data bases can store pictures and graphs as a field; the pictures might come from MacPaint or another program. (Microsoft File, Ensemble, and Helix are examples.)
- Illustrations. You might want to use pictures in reports, so you can add artwork to a printed report; the most flexible can mimic existing printed forms or create new ones. (MegaForm and FileMaker can mimic forms.)
- Calculate on fields. Although unnecessary for address books, calculations are necessary for many applications, such as managing sales information and inventories.
- Variable-length fields. Most traditional data bases only accept fixed-length fields, so you can't squeeze in a little extra information or annotate specific records. Variable-length fields let you add additional notes anywhere you want, somewhat like the margins in an address book. (Filevision, Helix, and Microsoft File offer this.)
- Search all fields. If you can't always remember where some specific information was stored, an all-field search will find it.
- Index all words. If you routinely create records with multiple-word fields, you might want to index all words in a field for later retrieval. (FileMaker and, to a lesser extent, FactFinder have this feature.)

- Sloppy-search function. Suppose you can't remember the spelling of an entry. A sloppy-search function lets you type in anything resembling what you want and will find close matches. You can often do this with a fragmentary search, using the letter you can remember, for example, but many database programs only search for your search string at the beginning of a field. A sloppy-search system should automatically search for homophonic sounds; if you enter "Brown," the program will also look for "Braun." (No one does this well yet.)
- Information interchange. Some database information is structured much like a spreadsheet. If you use both types of program, you may want a data base that works smoothly with a spreadsheet and lets you move information back and forth with a minimum of fuss. (1st Base, Microsoft File, and others do so.) Integrated packages can bind the spreadsheet and data base together tightly. (Jazz makes a start at this, but does not link the two tasks.)
- Power. Will the program grow with your needs? In general, you should buy a program with a little more power than you think you need, rather than one that you might outgrow quickly. Some programs will create compatible data files that you can use with more advanced programs. To do a simple task quickly, consider a file-card data base.
- Networks. Several companies have been preparing sophisticated database programs for network operation; such programs permit several users to work with a file simultaneously (see Chapter 26). Omnis 3 is the first Macintosh database program for networks.

DATA BASE ORGANIZATION

Database programs organize their data along three major models:

- Flat-file data bases store all data in a simple sequence for each record; the data can be represented in a grid with the records as rows and the fields as columns.
- Hierarchical data bases establish fields or, sometimes, distinct files in a tree-like structure where access to any particular field follows a branching path.

- Relational data bases use fields that are common to several files to establish relationships among the files. The most powerful of these programs can establish multiple relationships simultaneously.

Notes on Database Programs

There are so many database programs for the Macintosh that Table 6-1 mentions only some representative programs. They are grouped by power or function for easier comparison.

TABLE 6-1. REPRESENTATIVE DATA BASES

High-powered data bases

MacLion (version 3.1): A programmable, relational database package centered around the LEO programming language (a variant of Forth); LEO is hard to write and read; although you can generate the necessary code for some common functions via menu selections, most of MacLion's power is difficult to get at; suitable for programmers only.

Omnis 3 (version 3.10): From Blyth Software; a traditional data base modernized with Macintosh features; can organize files in hierarchical or relational form; includes useful application-development tools; can work on networks with multiple users.

Medium-powered data bases

Ensemble (version 1.0): From Hayden; emphasizes graphics; can store pictures as a field and create graphs from the data; suffers from cryptic icons and awkward organization. Hayden bills Ensemble as an integrated package, but it is really a data base with some graphics extensions.

1st Base (version 1.01): A simple relational data base, not particularly flexible. Two accessory programs help out 1st Base: 1st Port converts files to and from other common programs; 1st Merge creates form letters with MacWrite.

Microsoft File (version 1.0): Good use of the Macintosh interface; you can select and change fonts, as well as store pictures as a field; data can be merged with Microsoft Word documents.

OverVUE (version 1.0a): From ProVUE Development: handles data in grid format, somewhat like a spreadsheet; can create summaries of data quickly; relatively easy to learn and use, but a little rigid in design.

Forms-generating data bases

FileMaker (version 1.0): From Forethought; can combine text and graphics into a printed form; good use of Macintosh interface; compared to Mega Form, has a more powerful data base with free-form searches, but the printing tools are not quite as complete.

MegaForm (version 1.00) with MegaFiler (version 1.01): From Megahaus; MegaFiler is undistinguished by itself, but works with MegaForm, an excellent forms-generating program. MegaForm with the LaserWriter printer can re-create any standard printed form, complete with logos, fine print, and shaded areas; entries within the form are treated as fields, and calculations can be performed on the fields.

File-card data bases

FactFinder: From Forethought; indexes all selected words in the data base for quick retrieval.

I Know It's Here Somewhere: From Hayden; can handle text searches better than most programs.

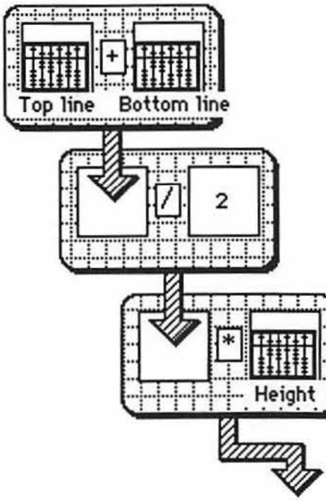
TABLE 6-1 (cont'd).

Unusual data bases

Filevision (version 1.0): From Telos; a visually oriented data base. You can create a picture and point at objects within the picture to retrieve information; thus, you could create a floor plan and then point at rooms to bring up a room inventory. Filevision is a terrific idea that doesn't quite work yet; creating pictures is tedious; a new version with an ability to accept pictures from MacDraw would help greatly.

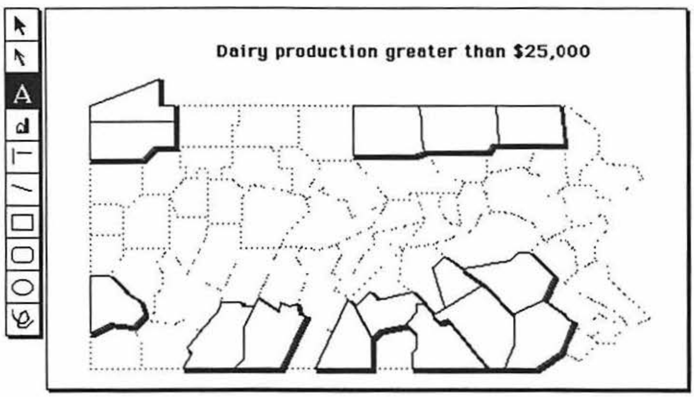
Helix (version 1.5): From Odesta; a distinctive data base with a library of icons that represent logical decisions. The idea is powerful but Helix also doesn't quite work yet; the first version is often hard to understand and is unusually difficult to learn. A redesigned version could be exceptionally valuable for teaching programming and computer concepts.

If you are accustomed to using text-only data bases, the following illustrations show how Mac's graphics capability is used by two products, Helix and Filevision.



Helix creates distinctive icons that represent logical calculations and steps. The icons illustrated here show the steps in calculating the area of a trapezoid (the sum of the parallel sides divided by two, times the height).

The Filevision data base can show calculated results graphically (here, Pennsylvania counties in an agricultural data base designed by Graham Bell). You can click on specific areas of the picture for more information.



POTPOURRI

The remaining sections of this chapter will cover an assortment of programs designed with the professional user in mind. They run the gamut from established (accounting) to rather exotic ("expert" systems) functions.

ACCOUNTING PACKAGES

If you are considering accounting programs, you should examine the available packages with the advice of a qualified accountant. Many general-purpose accounting programs are too general, unable to meet the needs of particular businesses. An increasing number of accounting programs serve specific businesses, such as medical offices or auto dealerships.

Many accounting programs, including specialized ones, can be modified to meet your needs. Most often you will need a programmer to assist in making the changes.

Besides your accounting needs, see if the program will work with the other software you use. Can you generate graphs from the data? Or move the data to a spreadsheet to make financial projections? Can the program grow with the company? Can your accountants use your files on their computer to perform audits?

Except perhaps for the very smallest business, any computer used for accounting will need a hard disk to store accounting programs and related information.

FINANCIAL MODEL BUILDERS

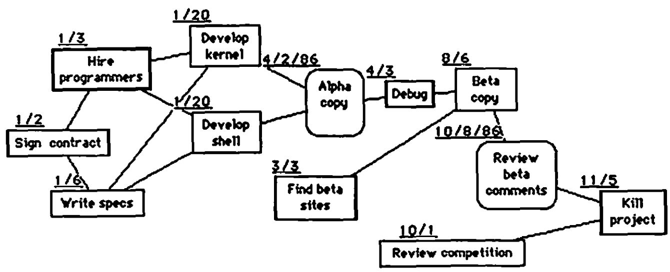
For many business computations, the formulas are well established. Everyone uses the same formulas to calculate depreciation or financial ratios, for example. A computer program can therefore speed up analysis of common situations by building a financial model for you. All you have to do is answer a series of questions, and the model builder creates a file. The file can then be read by a spreadsheet or other program and you can devote your time to analyzing the results rather than reinventing the wheel.

Many financial-modeling programs are appearing; some of them are free-standing and others are designed for a specific spreadsheet program. Several programs have already appeared for Multiplan, and companies will release programs for other spreadsheets as well.

The Dow Jones Spreadsheet Link in combination with Straight Talk extracts financial market information from the Dow Jones dial-up data base and presents the data formatted for Multiplan (and, in time, presumably other spreadsheets). It automates many steps and will appeal to people who want to check up on their stock portfolios regularly.

PROJECT MANAGEMENT

MacProject (version 1.0, from Apple) deals with time and resource allocations. You can examine time, personnel, budget, and material needs for a complex project and see which steps are the most critical to the project's success.



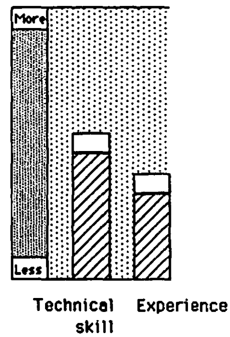
MacProject and related programs are mainly useful for fairly well-defined projects for which you can specify the parameters with some precision. For the chaos that dominates over systematic planning in most organizations, these programs are less effective; they don't handle fuzzy information and goals well. The more enthusiastic business-school graduates argue that a project-management program should impose order on the chaos; let the master plan, as expressed by the software, tell everyone what they must do and when they must do it. The B-school crowd may be right or merely naive, depending on the situation—or your point of view.

DECISION SUPPORT

A new class of decision-support programs tries to deal with fuzzy issues. Lightyear (pre-release, from Lightyear, Santa Clara, CA) and Decision Map (pre-release, from SoftStyle, Honolulu, HI) work in roughly similar ways. You begin by constructing a model listing the factors in a decision; these factors will usually be added together or multiplied for the overall result. Comparing several candidates for a job, for example, you would include experience, training, salary expectations, and so on. The programs try to make subjective evaluations easier by giving you a sliding, non-numeric scale calibrated with phrases such as "very important" at one end and "not important" at the other. You can change the weighting for each factor selectively and see how the comparison changes as a result.

Of course, the subjective scales are merely converted to a numeric scale within the program, so any spreadsheet can do the same analysis—you set up the same model, enter numbers corresponding to your subjective assessment, and multiply the results with weighting factors. However, spreadsheets are slightly less convenient for entering values.

These decision-making models are probably only useful for highly structured problems, such as analyzing mortgage-loan applications or filling many identical jobs at the same time. For the more common problem of selecting one candidate for a single job, the models are no better than the weighting you assign to each factor. Many people will fiddle the weights until they get the desired result, and then later use the "computer" model to justify a bad decision.



Subjective scales in Decision Map register weighting factors (here, factors in a personnel decision).

TIME SCHEDULING

Nearly all of the many calendar programs for the Macintosh are impractical, for reasons explained in Chapter 19.

For major scheduling tasks, such as in a medical or legal office, Front Desk (version 1.0, from Layered, Boston, MA) offers comprehensive functions. It can track the schedules for up to 15 persons as it records billing time and canceled, missed, or rescheduled appointments; you can set up standard appointment lengths, block out holidays, and view a week's worth of appointments for five people at a time on one screen. Front Desk even copes with common scheduling abuses, such as double-booking in a doctor's office.

"EXPERT" SYSTEM PROGRAMS

In the years to come a valid class of "expert" system programs will appear—programs that can store knowledge bases on specific, well-defined subjects. These programs will be able to analyze a particular situation and offer suggestions or predict potential problems.

Some programs already available claim to offer expert advice. For example, several try to tell you how to negotiate or close a sale; you answer a series of questions about the person you are dealing with and the program generates several pages of advice. But there's much less to these programs than meets the eye. You must know your "adversary" well to fill out the questionnaire accurately, and if you know the person that well, you probably don't need the program to begin with, even assuming a valid analysis—a dubious proposition.

Integrated Software

Chapter 7

Traditional application software performs a single task; a program might process words or numbers, but not both. Yet often the information you create in one task, such as putting quarterly results in a spreadsheet, needs to be included in another task, such as a report, from a word processor. On the Macintosh, most programs can exchange information via the Clipboard and the Scrapbook, but such exchanges have a serious limitation: Once moved, the information is static, or "dead." If the numbers in the spreadsheet change, you must move them again to the memo.

The key to truly integrated software is "live" links: Once transferred, the information "remembers" where it came from, so if you change the original information, all places that you have used it change automatically.

JAZZ

Lotus Development Corporation's Jazz (release 1) is the first microcomputer program to handle live links well. The program includes a word processor, spreadsheet, and data base, plus business graphics and telecommunications.

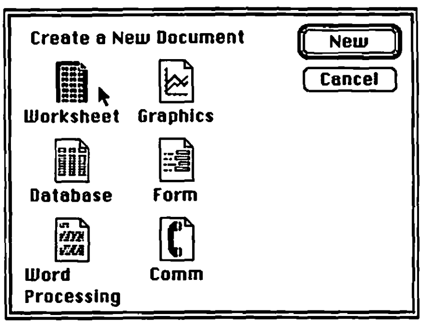
Jazz lets you move live information from spreadsheet, graphing, and database tasks to the word processor. If, for example, you create a spreadsheet and then create a graph based on that spreadsheet, you can incorporate both in a memo. If you later update numbers on the original spreadsheet, not only does the graph change automatically, so do the parts of your spreadsheet and graph you included in the memo.

Jazz requires a 512 KB Macintosh and two disk drives. It is a large program—380 KB plus help messages—and it won't fit, along with the system files, on a standard single-sided start-up disk. Instead, you put Jazz in the external disk drive and use the start-up disk for data. A hard disk drive improves matters greatly; some hard disk drives, however, connect through the modem port and will disable Jazz's communications features.

Even the 512 KB Macintosh doesn't have enough RAM for Jazz to load completely, so some program components are swapped between the disk drive and memory as required. In operation, about half the memory is allocated to the program and the other half (256 KB) to your data (the exact amount depends on font sizes, desk accessories, and other factors). All active data must be in RAM. Nevertheless, with Apple's Switcher utility you can run small programs such as MacPaint at the same time as Jazz, provided that you keep your data files small.

Starting Jazz

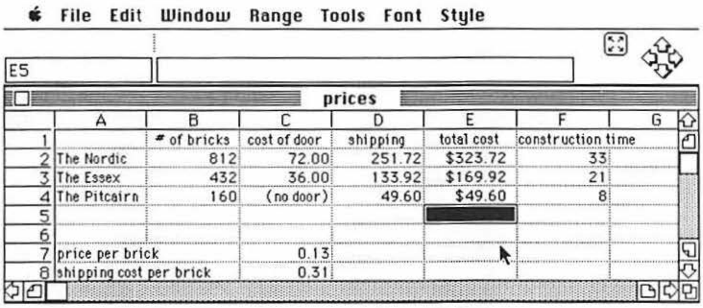
You start by pointing at the Jazz icon and clicking the mouse button twice. The New... entry in the File menu displays tasks you can choose from.



If you select Worksheet, Jazz loads the spreadsheet and displays a window with the familiar grid of rows and columns. (See Chapter 5 for a general comparison of spreadsheet features.) If you are familiar with Lotus 1-2-3 for MS-DOS microcomputers, Jazz has roughly the same power and functions except for the split screens and keyboard macros. The absolute spreadsheet size is 8192 rows by 256 columns; the usable size will depend on the data you enter and how many other documents you have open at once. You can have several spreadsheet windows open at the same time and you can cut and paste static information between windows.

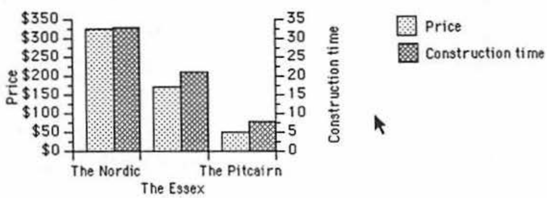
Using Jazz

The Jazz example here prepares a price-quote letter. The first step is a spreadsheet quoting the essential costs. A graph based on calculations in the spreadsheet, plus a fragment of the spreadsheet, are included in the finished letter.



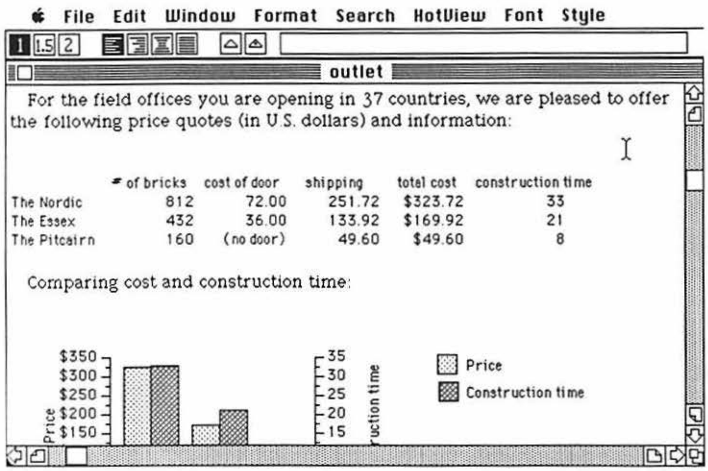
	A	B	C	D	E	F	G
1		# of bricks	cost of door	shipping	total cost	construction time	
2	The Nordic	812	72.00	251.72	$323.72	33	
3	The Essex	432	36.00	133.92	$169.92	21	
4	The Pitcairn	160	(no door)	49.60	$49.60	8	
5							
6							
7	price per brick		0.13				
8	shipping cost per brick		0.31				

Once you have numbers in the spreadsheet, you can create graphs quickly. You select a range of cells with the mouse and open a Graphics window (unlike 1-2-3, there is no disk swapping). Selections from the Type and Plot menus include standard bar, line, and pie charts. For automatic labeling of each plotted point, you can enter text in the spreadsheet in a cell next to the number or you can point anywhere on the graph with the mouse, plunk down some text, and add arrows to highlight what's most important.



For elaborate work, you can transfer the entire graph to MacPaint or MacDraw and add a diagram or a logo.

Jazz's word-processing window looks like MacWrite's with the icons rearranged. Operation and functional power are virtually identical to MacWrite, but you can open several windows at a time and can cut and paste static text from one window to another. See Chapter 3 for a general comparison of word-processing features.



The Jazz data base is adequate for straightforward file management but suffers from a stiff operational style. The data base creates simple flat files—files whose elements can be represented in a rectangular, spreadsheet-like matrix with the fields in columns and the records in rows (see Chapter 6). You enter formulas and conditional tests for database queries and reports by typing them on the keyboard (for example, =FSUM*field* means add all the values in the field) rather than selecting the options from a special display. This arrangement, also used in the spreadsheet, is a little harder for beginners and a little faster for experienced users. The report generator that sets up printed reports doesn't make effective use of the Macintosh interface. The essential capabilities are there, but you can't pick up and move or resize items with the mouse. And the data base is traditionally conceived: It will not store pictures in a field; you cannot store a little graph that might show a person's sales performance.

Jazz's communications component works well and has all the necessary options. You can set an answer-back for automatic acknowledgment of the remote computer and you can emulate a Digital Equipment Corporation VT100 terminal. You can specify how tabs, carriage returns, and other special characters are to be treated, and so organize information from many sources into files Jazz can use.

Because integrated software uses linked files, you should be especially careful when you rename files generated by Jazz. Files are referenced by name, so if you change the name the link can be broken. To rename a file, load it into Jazz and use the Save as... function from within the program instead.

Other Integrated Packages

Other integrated packages will appear. Many will not be true integrated packages, which I define here as including word processing, a spreadsheet, and a data base. Some companies use a looser definition whereby any program providing two or more functions is labeled integrated. Ensemble from Hayden is really a data base with graphs, and Quartet from Haba Systems is a spreadsheet with graphs; they are described in Chapters 5 and 6, respectively. Both lack real word processing; neither can handle type fonts or styles, and both are limited to documents of less than 1000 characters (less than half a page).

INTEGRATION STRATEGIES

If you do several tasks with your Macintosh, you can either choose a single, fully integrated package such as Jazz, or build up a collection of separate programs. A collection of programs will usually run much more slowly, particularly if you must save, quit, and reload files on a disk drive. Whether you use Jazz or a suite of programs, you will probably find a hard disk essential.

Apple's Switcher utility program allows you to partition memory into as many as four parts and load a different program into each partition. You can then change instantly between programs. You can transfer static Clipboard and Scrapbook information across the partitions. Switcher works very well subject to memory limits; 512 KB won't go very far.

Another way to get more than one task running at a time is with desk accessories (see Chapter 9). Several desk-accessory communications programs are available, with sufficient power for ordinary tasks. Similarly, the MockWrite desk accessory lets you run an effective, if bare-bones, word processor along with any other program. A few programs may not work with certain desk accessories; test before embarking on a complex job.

COMPARISONS

The obvious comparison pits Jazz against the Microsoft programs—Jazz versus Excel or Multiplan/Chart, Word, and File. The missing component in the Microsoft series is telecommunications, which could be done with a desk-accessory program. The Microsoft set works together better than a collection of programs from different publishers since they share an operational style, use a common set of command-key assignments, and can exchange files fairly readily.

Looking at each individual task in isolation, the Microsoft programs are generally more powerful and better designed than the components of Jazz. The greatest difference is in the data base; File is far superior in every way to the Jazz data base. Word is much more powerful than the Jazz word processor. Excel is much more powerful than the Jazz spreadsheet, boasting more features for the experienced user. Overall, Jazz makes less effective use of the Macintosh interface than Excel or Multiplan; Jazz does not have split windows; it uses minimal command keys, and cutting and pasting in both the spreadsheet and data base are limited to complete entries—you can't simply pick up a fragment and move it.

On the other hand, the Microsoft series is a set of separate programs, and you cannot run them all at the same time unless you have a megabyte of memory. Any information transfer among them is static, except for live link among the spreadsheets and graphs in Excel. Jazz features superb integration from the spreadsheet, business graphics, and data base to the word processor (although not in the reverse direction or between two files of the same task).

If you need to complete a complex memo under a deadline, Jazz is the clear choice. If you don't often move information to text files and you want operational power, then the Microsoft set is a better choice. Jazz gives you speedy results with minimal fuss; the Microsoft set gives you power at the expense of extra steps.

Could you use Jazz simultaneously with the Microsoft programs? Yes, but file-format conversion is a nuisance and some features will be lost in the transfer. Also, the working styles of the programs conflict in a mild way; Command-W changes window size in Jazz, but closes the active window in the Microsoft set. The conflicts aren't usually fatal but they can get in your way when you are fluent on one program but not the other.

Chapter 8

Introduction to Communications

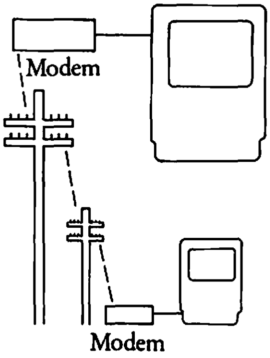
Someday, fast and efficient communication between computers will give us instant access to each other, to the world's libraries, and to the latest political gossip. But today, communication can be the most frustrating operation you attempt to perform with your computer.

When we use the Macintosh to communicate with other computers, we enter a chaotic world, for unfortunately the lack of industry-wide standards also afflicts the Macintosh. And the situation is likely to get worse before it gets better as the number of incompatible products increases.

Despite the problems, though, with a telephone line and the proper hardware and software, you can connect your Macintosh with public information services and data bases, electronic mail services, and anyone else who also has the proper equipment.

This chapter deals mainly with communications software and only covers the information essential to getting you started. For more information about modems, see Chapter 18. For a complete discussion of communications protocols, see Chapter 25.

WHAT YOU NEED



To use your Macintosh to talk to another computer, you need:

- Communications software—MacTerminal from Apple, Smartcom II from Hayes, or a similar program.
- A modem. Many modems will work with the Macintosh; make sure that the communications software you choose is compatible with the modem hardware.
- A telephone line (not a party line).

You will also need some information, called communications parameters, about the computer you want to communicate with and its modem:

- Speed—the modem's operating speed (300, 1200, or 2400 bits per second in most cases, corresponding to roughly 30, 120, or 240 characters transmitted per second). The term *baud* is commonly used (inaccurately) in place of bits per second; see Chapter 25.
- Character width—how many bits make up each of the characters being transmitted (seven or eight); also called data bits.
- Stop bits—how many bits (one or two) mark the end of each character, so that the computer knows when one character stops and another begins. This parameter is often unnecessary.
- Parity—a simple form of error detection (even, odd, or no parity).
- Handshake—a way for the receiving computer to tell the sending computer to pause (XON/XOFF, clear to send, or no handshake). This parameter is not always needed for short messages or files.

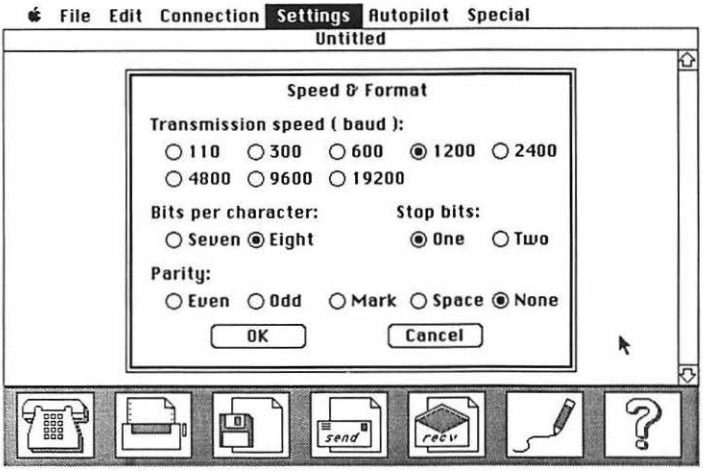
GETTING READY TO COMMUNICATE

Connect your modem to the Macintosh and to a telephone jack according to the instructions in the modem manual. Insert a disk with the communications program into the disk-drive slot.

Specific set-up instructions will vary among different communications programs, although general procedures are similar. The following instructions apply to Smartcom II from Hayes. (This program will not work in its original form with an Apple modem.)

Choose **Speed & Format...** from the **Settings** menu.

A dialog box appears.



Click the appropriate Transmission speed ("baud"), Bits per character (character width), Stop bits, and Parity.

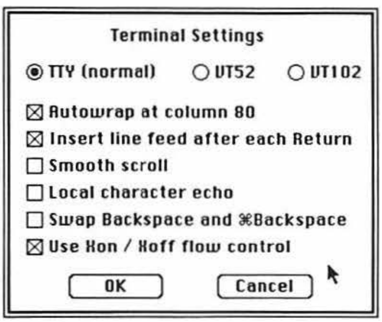
If you have control over the setting of both the sending and receiving machines, the choices of character width, parity, and so on aren't important as long as the two computers agree.

If you are trying an information service for the first time and you don't have its parameters, start with:

8 bits per character
1 stop bit
no parity

This combination often works.

Choose **Terminal...** from the **Settings** menu.



If you don't know the terminal type, start with TTY. The wrong choice will lead to problems only if the other computer expects you to have full-screen editing or other features not commonly used in the average microcomputer communications link.

Next, click the boxes for Autowrap at column 80, Insert line feed after each Return, and Use XON/XOFF flow control.

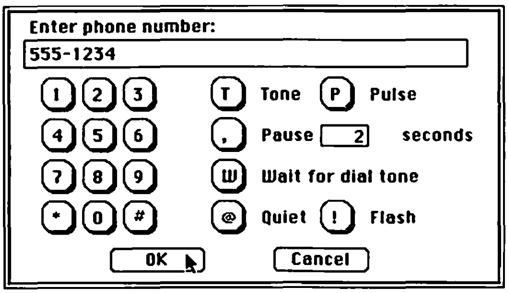
When your Macintosh talks to another computer, one computer/ modem must be set to originate and the other set to answer. By convention, the person who makes the phone call originates. If you are the calling party:

Choose **Originate** from the **Connection** menu.

The person who receives the call should set the receiving modem to answer the call automatically. If you are being called:

Choose **Answer** from the **Connection** menu.

Choose **Phone number...** from the **Connection** menu to set the telephone number.



Enter the number—say 555-1234—on the line or click the number buttons. If your phone line requires that you dial 9 first, enter 9555-1234. If you must usually wait a moment to get a second dial tone after you dial 9, enter a comma after the 9 (9,555-1234); the comma inserts a two-second pause in the automatic dialing sequence of an Apple or Hayes modem; for longer delays, use more commas.

You can put in hyphens to make your number more legible; the modem will ignore them.

Make sure no one else is using your phone line.

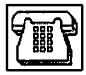
Most modems connect directly to your computer and to the phone jack so they don't need a telephone for operation. You may, however, want to keep a telephone connected to the line for voice operations.

Make sure all your phones are on the hook (that is, hung up) before making the connection.

MAKING THE CONNECTION

With all that preparation out of the way, you are ready to make the connection with the other computer and start transmitting your data.

Click the telephone icon, and then click OK in the dialog box.



The Hayes modem contains a speaker; you will hear the modem dialing the number and then the shrill answering tone from the other computer's modem. Your modem then switches automatically from voice mode into data mode and silences its speaker. You can proceed with transmitting your data.

If you hear a busy signal on the line instead of the other modem's answering tone:

Click Cancel in the Call Progress box.

If you hear a person's voice:

Pick up a telephone before clicking Cancel.

Once the two computers are connected, whatever you type at Mac's keyboard will be transmitted to the remote computer.

If you cannot see what you are typing on the screen, turn on local character echo in the Terminal... dialog box under the Settings menu. If you see doubled characters, turn off local echo.

When you are done:

Click the telephone icon.

With luck, everything will work and you can enjoy computer communications.

Making It Work

For many computer systems, you must send a carriage Return or Command-c (Control-c)—sometimes more than one—to elicit a response. (The remote computer can use these characters to determine what speed you are using, among other things.)

Under some circumstances, a few programs change the serial-port settings in the clock/calendar's memory (see Chapters 15 and 29). If a port does not respond correctly or you get garbage on your screen when you type with a communications program running before you are connected to another computer, turn off everything, remove the battery on Mac's back panel momentarily, and put it back. Then start over again.

COMMUNICATING BETWEEN MACINTOSHES

Moving information between identical computers is usually easy: If the computers are within walking distance, you simply swap disks. Between computers, you can also use a direct wire connection (called a "null-modem" because the wire replaces two modems), or you can use telephone lines.

In any case, you should use the same communications software at each end, set up with identical parameters. On a telephone line with modems, one Mac must be set to originate, the other to answer.

SO MUCH FOR THE EASY PART...

Because communications involves so many steps, finding the exact source of a problem can be difficult. You can test your own hardware and software by calling another computer, but even a successful connection won't rule out subtle problems in data transfer.

For more information, including an explanation of what happens at each step, read Chapter 25. You'll probably find that chapter easier to read after you have had some experience with communications.

SOFTWARE FEATURES

All general-purpose communications programs include essential features, such as setting the communications parameters, sending a file from a disk, and storing an incoming file to disk. Here is a list of features that are not found in all programs but that you may find valuable. For full explanations of technical issues, see Chapter 25; for a discussion of security issues, see Chapter 24.

- File-transfer protocols. Virtually all communications programs can handle standard text. Most of those for the Macintosh can also transfer programs and graphics by using a special protocol that insures error-free transmission of any file. The most common such protocol is Macintosh Xmodem.

- File-transfer time and status. As you begin transmitting a file, many programs tell you how long it will take and show you a progress report.
- Auto logon. When you dial up a remote computer, you usually have to supply a logon name and password to gain access. Some programs let you automate this procedure. *Caution*: In some situations you should not use an auto-logon feature for security reasons; anyone can read your password off the disk.
- Interactive communications. A recent feature, interactive communications lets you and another computer user (normally, but not always, on another Macintosh) work simultaneously on a document. The first such Macintosh program is Smartcom II from Hayes, which includes interactive drawing. Two Macintoshes connected together can share the palette and drawing window; anything drawn on one Macintosh is mirrored on the other. Interactive word processors should also appear in the coming months; with this feature you and an associate across the country will be able to edit a memo together.
- Terminal emulation. Many programs can make the Macintosh behave like a Digital Equipment Corporation VT52 or VT100 terminal. Some programs also allow Mac to emulate a Data General Dasher or Tektronix graphics terminal, or an IBM terminal. Terminal emulation means that the remote computer can send short commands to erase a line, move the cursor around, and so on. In many cases the emulation is not complete, although the missing features usually aren't critical.
- Session editing. After you are through communicating, some programs let you go back to edit the entire session and select only the interesting material to save on disk. The length of the session recorded depends on the available memory; a few programs will record the session on disk.
- Filters. Many computers send a variety of special characters during communications. In some cases these characters interfere with your screen display. Inward filters can be set to remove these characters before they reach your screen. Outward filters do the reverse, removing characters that might disturb the remote computer. In Macintosh files, the special characters created with the Option key will sometimes cause such trouble.

COMPARING COMMUNICATIONS PROGRAMS

You can choose from many programs. Some are listed and described in Table 8-1; they are grouped as follows:

- General-purpose communications programs. In addition to the customary features, all these programs can send complete files, including programs and pictures, to another Macintosh, using a modified Xmodem protocol (sometimes called Mac Xmodem).
- Specialized communications programs. These programs cannot send complete files to another Macintosh.
- Other communications programs.

When you compare products, check the version numbers, and compare the features of the latest versions.

TABLE 8-1. COMMUNICATIONS PROGRAMS

General-purpose programs

MacTerminal (version 1.1): From Apple; performs essential functions but does not let you set answerback, so you cannot log on automatically to the remote computer; offers adequate but somewhat limited control over communications parameters. Can emulate an IBM 3278 terminal with the AppleLine protocol converter; the Microsoft MacEnhancer includes a version of MacTerminal, but leaves out IBM terminal emulation.

MITE (version 2.92): From Mycroft Labs; gives unusually comprehensive control over communications parameters; a particularly good choice if you have a nonstandard modem or you link up with a reluctant remote computer. Error-correcting protocols include Clink (compatible with Crosstalk) and Hayes, as well as Xmodem.

VersaTerm (version 1.42): Distributed by Peripherals Computers and Supplies, Inc; emulates VT100 and Tektronix 4014 terminals; also, partial emulation of Data General D200 terminals; can send predefined strings to remote computer, but does not have auto logon; Tektronix emulation not as complete as with Tekalike (see Specialized programs).

Smartcom II (version 2.1B): From Hayes; easier to use than most; driven by icons, rather than pull-down menus; outstanding (if you need to draw pictures in this fashion) interactive drawing mode lets you draw on your screen and a remote Mac when both are equipped with SmartCom. Does not work with Apple modems.

Specialized programs

MacModem (version 2.0): From Microcom; has particularly complete programming functions; can be set to perform many functions automatically; unfortunately, only works with Microcom modems, a nonstandard form; supports MNP, a sophisticated but not widely used error-correction protocol; does not support Xmodem.

Straight Talk (version 1.1.1): From Dow Jones; specifically designed for connecting to dial-up data bases; not really suitable for general computer-to-computer connections, since it can transfer text only; has auto logon, and menus include several commands specifically for the Dow Jones data base; works with Dow Jones Spreadsheet Link for extracting financial information from Dow Jones and putting it directly into a spreadsheet; does not support Xmodem.

Tekalike (version 1.1.1): From Mesa Graphics; supports Tektronix 4010, 4012, 4014, and 4016 graphic terminal emulation; has direct support for several plotter models when running as a Tektronix terminal; pictures sent from the remote computer using the Tektronix protocol can be converted to MacPaint files; colors can be rendered as screen patterns; does not support Xmodem.

Griffin Terminal: Developed at Reed College and distributed by Metaresearch (Portland, OR); supports Tektronics 4012 terminal emulation, includes simple text editor.

Other programs

Desk accessories: Handy; you can use them simultaneously with any application program; programs must be small, however; many features will not fit in a desk accessory. CE Software (MockTerminal) and Dreams of the Phoenix (Jacksonville, FL) both have effective desk-accessory communications programs.

Through user groups: Include *MacTEP* by Dennis Brothers and *RedRyder*; provide essential features and serve for occasional communications on a tight budget.

MacLink, PC to Mac and Back, and 1st Port: All designed for moving files from an IBM PC to the Mac, although they can serve for general communications. See Chapter 27 for details.

Dealing with Disks Chapter 9

Let's deal with two key questions: Should you get a second microfloppy disk drive? Yes. Should you get a hard disk drive? Probably. Let's see why by taking a look at disks and how they're used. This chapter is based on Finder version 4.1.

SETTING UP YOUR DISKS

Disks come in two main types: system disks and document disks. System disks contain the information, or system files, necessary to start a Macintosh. You can start a Macintosh from the internal or an external disk drive. In most cases, system disks also contain one or more application programs. You can store documents on a system disk, if there is enough space.

Unlike system disks, document disks store only documents; you cannot use a document disk to start the computer.

If you have two disk drives, you can also use program disks, which contain application programs (rather than documents) but no system files. If a system disk is in the internal drive, a program disk can be inserted in an external disk drive. Since a program disk does not contain system files, it has much more room for documents.

Which Files Belong on Which Disks?

The amount of information contained in a file is measured in kilobytes, or KB (units of 1024 characters). Single-sided Macintosh disks hold 400 KB; for comparison, a double-spaced page of text is about 1.5 KB, or 1500 characters, long. (See Chapter 14 for more information about kilobytes and disk storage space.)

System files

The System and Finder files are essential for nearly all Macintosh operations, both to start the computer and to run an application program. In most cases on a single-drive system, these files must be on the same disk as the application program.

The System file takes considerable storage space—up to 140 KB. The Finder, a key program that manages the Macintosh Desktop with its menus and windows, is stored in a file about 42 KB long.

Printer files

The Imagewriter file (or an equivalent file for another printer—see Chapter 16) must be on the disk if you wish to print from the disk.

If you need several application programs on a single disk and don't need to print while actually using your applications, you can delete the Imagewriter file (or equivalent) to save space on that disk. When you want to print your work, save it, move the file to a disk that contains the relevant application program and the Imagewriter file, and print from this disk.

In a few cases such as MacPaint, a program may write directly to the printer; if so, you can remove the Imagewriter file.

COPING WITH A SINGLE DISK DRIVE

Long files take up a lot of disk space, so it's best to keep only essential files on any working system or program disk. If your Macintosh has only its one single-sided disk drive, you have to plan your disks carefully to avoid running out of disk space.

Creating Space on a Disk

Together, the System, Finder, and printer files take up half a single-sided disk, leaving about 190 KB for your application programs and their data. MacPaint and MacWrite are each 50 to 60 KB long. If you keep both on one disk, you will have only 70 to 80 KB left for your work. MacPaint won't even start unless there is enough space to save a complex image, so you can't have many images or much text stored on the disk in a single-sided, one-drive Macintosh. You can create more space in several ways.

Trimming the System file

A significant portion of the System file is taken up by the character fonts. To make more space available, some application programs come with a smaller, alternate System file containing only those fonts the Macintosh and the application need to operate.

If necessary, you can use the Font/DA Mover program to reduce the size of the System file yourself. With Font/DA Mover, you can copy any unneeded Desk Accessories and fonts from the System file to other files or disks. Later, you can restore the Desk Accessories and Fonts file to the System file if you wish. Open the Font/DA Mover program and select Help to find out exactly what you can do and how.

The minimum System file size is about 64 KB in the initial versions.

Font/DA Mover

Deleting the Finder file

You cannot reduce the size of the Finder file, but you can delete it if you want to dedicate a start-up disk to a single application program.

Choose **Set Startup** from the **Special** menu.

Select the application to which you want to dedicate the disk.

Eject the disk.

Restart the Macintosh with another system disk.

Eject the second disk and insert the original disk.

Move the Finder icon to the trash and empty the trash.

Now when you restart the computer with the original disk, you will go straight to the application. You will not be able to quit the program and go to the Desktop to perform such Desktop operations as copying a file, but these operations can be done with the aid of another disk that still contains the Finder file.

Deleting help files

You can save a little more space by deleting the help file from some programs. These files give you advice about using the program while you are running it, and once you have some experience, you may not need them. *Caution:* Some programs with a help feature do not have a distinct help file, and other programs will not start unless the help file is on the disk.

Trimming files created by the system

Depending on what you've put in them, the Clipboard and Note Pad files will use from about a few hundred to a few thousand bytes of space, the Scrapbook file potentially more. If you run out of space, you can move these files to another disk.

The Clipboard file is used to store transient Clipboard information if the information won't all fit in memory. You can erase a Clipboard file in the trash, but the next time you do any work with the Clipboard, the system will create the file again.

The Scrapbook file contains all the information you have pasted into the Scrapbook. You can move the Scrapbook file to another disk. If the second disk already has a Scrapbook file, an alert box notes the duplicate name but, as with any other file, you can change the name before moving the file. You will still need to put the original Scrapbook file in the trash and choose Empty Trash from the Special menu to increase the disk space. Later, you can move the old Scrapbook file back, and restore its name.

The Note Pad file contains entries you have made in the Note Pad desk accessory. This file can also be put in the trash and erased with Empty Trash.

In a two-drive system, the active Scrapbook and Note Pad files are the ones on the disk whose window is currently active.

The Best Solution...

The only satisfactory solution to disk-storage limitations is a second disk drive. A second disk drive improves storage dramatically, and would be valuable even with double-sided disk drives. The System, Finder, or printer files don't usually need to be in the second disk drive (a few programs will expect to find these files on the same disk).

MOVING INFORMATION BETWEEN PROGRAMS

When working with most Macintosh programs, you can save information in the Clipboard and the Scrapbook so another program can read and use the information.

The Clipboard

When you choose Cut or Copy from the Edit menu, the selected information goes into the Clipboard. You can see the Clipboard contents by choosing Show Clipboard from the Edit menu. To move the information into another program:

Cut or **Copy** the desired information in the first application program.

The information goes into the Clipboard.

Choose **Show Clipboard** from the **Edit** menu to see the Clipboard contents.

Not all programs have a Show Clipboard option.

Choose **Quit** from the **File** menu (Save the file if necessary).

Open the second program icon.

Find the place where you want to insert the information.

Choose **Paste** from the **Edit** menu.

The Clipboard contents reappear in the second program.

Note that the Clipboard can contain only one piece of information at a time. Any time you Cut or Copy, you replace whatever you might have put in the Clipboard earlier. To avoid changing what is in the Clipboard, some programs give you the option Clear in the Edit menu, which is similar to Cut, but does not change the Clipboard (Clear may not work exactly the same way as Cut, however).

The Scrapbook

For more flexible operation, put the Clipboard contents into the Scrapbook so they will not be erased by an interim Cut or Copy.

Cut or **Copy** the desired information.

Choose **Scrapbook** from the Apple menu (far left).

The Scrapbook window opens up, showing earlier contents, if any.

Choose **Paste** from the **Edit** menu.

Now the Clipboard contents appear in the Scrapbook window.

The Scrapbook window shows the number of items it contains (the number is automatically incremented as you add to the Scrapbook); at the bottom right, the data type appears (TEXT, PICTure, and so on). Select Scrapbook items by clicking the ends of the scroll bar.

Close the Scrapbook window by clicking its close box.

Choose **Quit** from the **File** menu (Save the file if necessary).

Open the second program, and find the place where you want to insert the Scrapbook information.

In many cases, you will need to click this location to place an insertion point there.

Choose **Scrapbook** from the Apple menu.

Select the desired Scrapbook item by clicking the arrows.

Choose **Copy** from the **Edit** menu; *or*

Choose **Cut** if you won't need the item again.

After you cut the item, it will disappear from the Scrapbook.

Close the Scrapbook window by clicking its close box.

Select **Paste** from the **Edit** menu.

The Scrapbook item appears in the second program.

Not all information can pass between any two programs; there are restrictions related to both the source program and the receiving program. For more information, see Chapter 19.

Only one Scrapbook can be in use at a time, the one on the disk you used to start your current task—the program disk if you opened the program icon, the document disk if you opened the document icon.

KEEPING TRACK OF THE SYSTEM DISK

On the Desktop, the top disk icon in the upper right corner is the current system disk. You can change system disks by opening an application program on a new disk, or by holding down both the Command and Option keys and double-clicking on the Finder icon on the new disk.

The fonts and desk accessories available depend on the contents of the system disk. One way to distinguish system disks quickly is to set up each disk with a distinctive Desktop background pattern (use the Control Panel desk accessory). Then the Desktop pattern will change with each system disk.

HARD DISKS

Hard disks are covered in detail in Chapter 14. The software supporting most hard disks lets you divide the disk into several volumes; each volume behaves as if it were a distinct disk. You can mount and dismount volumes at will. For the sake of speed, don't mount more volumes than you need; otherwise the Macintosh will frequently have to perform housekeeping steps for the unused volumes. For the same reason, eject all unneeded floppies by putting their disk icons in the trash (this ejects but does not erase the disk).

RAM DISKS

You can set aside some memory so that it behaves as if it were an additional disk drive; doing so creates a RAM "disk," so-called because information is stored in random access memory (RAM). Such a disk is practical only on the 512 KB or larger Macintoshes. Because it is purely electronic, with none of the moving parts of a real disk drive, a RAM disk is very fast—much faster than even a hard disk. RAM-disk programs are available from many sources, including public-domain libraries. The better programs, such as Mac Memory Disk (from Assimilation), let you choose how much memory to set aside, and they automatically copy selected files into the RAM disk.

But RAM disks have two drawbacks: They take away from your memory space, and their contents are ephemeral, disappearing during a power failure or when you turn the computer off.

DISK CACHES

Like a RAM disk, a disk cache is a portion of RAM that stores disk information. But a cache does not store any specific files; instead it acts as a buffer between working memory and the disk drives, holding the information most recently read from or written to the disk. For example, many programs repeatedly read the same portions of a disk.

If that information is already in a disk cache, it is available immediately without the mechanical delays of a disk drive. On the other hand, when a program needs new information, there is no speed gain.

A well-designed disk cache regularly writes to the disk, so power failures usually will not erase unsaved information. The performance improvement from a disk cache depends greatly on the interaction between a specific program and disk files.

To use a disk cache, you simply install it with a utility program and then work as usual. The only effects you will see are faster operation and a smaller working memory. (A disk cache can also be installed in hardware, usually as part of a hard disk drive; for more information see Chapter 14.)

SWITCHER

Apple's Switcher is a powerful, outstanding utility program that lets you load up to four application programs in memory at the same time and switch instantly among them. You can specify the amount of memory you want to allocate to each application and share information among them via the Clipboard or the Scrapbook. Switcher does not actually run all four programs at the same time; only the one on screen is active. Most applications work fine with Switcher, although some need modification. Games often will not run because they use nonstandard programming tricks. The only problem with Switcher is that you become acutely aware of how little memory you have, even with a 512 KB Mac; more memory will help greatly. Because of memory limits, you can't run Switcher and a RAM disk at the same time unless you have more than 512 KB.

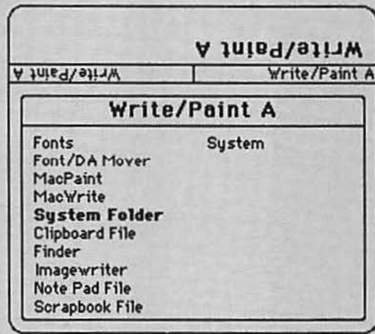
KEEPING TRACK OF DISKS

Write the version number and date on all new program disks when you receive them; otherwise you may later confuse them with updated versions. Table 9-1 lists products by category that help you keep track of your disks.

TABLE 9-1. DISK-ORGANIZING PROGRAMS

Disk labels

The MacLabeler: From Ideaform (Fairfield, IA); creates convenient labels for Macintosh disks, including inverted printing for the back and upside-down printing for the edge; reads the disk directory to create labels like the one shown at right.



Sector readers and editors

MacZap (Micro Analyst), ***MacTools*** (Central Point Software), and ***F Edit*** (John Mitchell): Specialized sets of programs; read and display raw information on the disk for you to inspect and change.

Desk-accessory programs

Memory indicator: Lets you know how much memory is in use at any given time.

Desk-accessory mover: Lets you add and remove desk accessories; most widely distributed, on a donation basis, is from CE Software; works well, although version 1.4 must reside on the system disk you are modifying (delete it when you are done). Duplicates some of the functions of Apple's Font/DA Mover but lets you rename desk accessories.

Delete file: A public-domain desk accessory; lets you delete files without quitting an application program; handy for making space on disk when you are in a jam.

DiskInfo: A desk accessory from Maitreya Design; displays files and file type for any disk without leaving an application; version 1.2 also deletes files.

Programs not yet available

FileInfo: Will display the first line of any text file so you can check contents quickly without loading an application program.

File Manager: Will look at every active disk and compare the date and times of all files with identical names; will ask if you want all copies to be replaced with the latest (or any other) version; combined with a back-up program, can be used to keep the contents of two hard disks up to date—especially useful if you have a Mac at the office and one at home.

Scrapbook Manager: Will let you temporarily inactivate the current Scrapbook and use one on another disk, or merge the two Scrapbooks together; will also let you edit the Scrapbooks by providing a full-screen display of the contents, which can be selected by pointing and clicking.

Chapter 10
Programming Languages

Programming languages are a complex and often controversial subject. Each language has its strengths and weaknesses; each also has strong advocates who insist that no other language is worthy of attention. Recognizing that others may not agree with my opinions, I offer this brief guide to computer languages in general.

DO YOU NEED TO PROGRAM AT ALL?

For most people, the answer is no. Increasingly sophisticated application programs will fill all common requirements, including most tasks previously accomplished by writing programs. So the majority of microcomputer users will never write programs—at least not in the traditional sense. When you work with a spreadsheet or data base, you are doing a form of programming, but you are using a very high-level language designed for a specific task. Traditional programming would use a general-purpose language to build up the steps required to perform the task.

Certainly no microcomputer user should begin with programming; everyone should learn applications such as MacPaint or MacWrite first. But if you need functions that packaged software cannot provide, or if you are simply curious, by all means learn to write your own programs.

WHO SHOULD LEARN TO PROGRAM?

For hobbyists, the choice of computer languages is not usually critical. Any modern structured language will probably suffice. The best choices are the most popular languages, such as BASIC and Pascal, since you can get help from many books, magazines, and friends.

Children may find programming interesting and absorbing. Parents should not, however, assume that skills in present computer languages will ensure a child's employability in coming years. By the time today's children become adults, vastly more powerful computers will have changed programming greatly. There is even the common, although dubious, argument that learning traditional languages will inhibit fluent usage of more powerful programming tools.

For children, the best language to begin with is probably whatever language is taught at school—BASIC and Logo are the most common. For high-school students studying computing, the College Entrance Examination Board chose Pascal as the standard language for its Advanced Placement examinations. The Board may also accept other languages in the future.

In companies, the decision of whether to program and which language to use can be critical. Poor choices, even if they yield quick results, can and do lead to years of frustration later. In many corporations, the traditional computer languages are already a severe barrier to progress. Programs written in those languages are increasingly difficult to maintain and improve. Data-processing departments have begun spending more time trying to maintain existing programs than writing new ones. Getting out of this fix requires new ways of writing programs; the traditional languages will not serve any more. For an excellent discussion of the current situation in corporate data-processing departments, see *An Information Systems Manifesto* by James Martin (Prentice-Hall, 1984).

AN OVERVIEW OF LANGUAGES

Languages can be classified in many ways. The distinctions are rarely absolute, however, as the successful ones evolve over time, gaining attributes of other languages.

However, computer languages do fall—more or less—into two general categories, high-level and low-level. Although the boundary is not well defined, high-level languages, such as BASIC, Pascal, and

Logo, contain elements of English in their instructions (source code) that make them easier for people to use. Low-level, or machine, language on the other hand, deals directly with the microprocessor instructions and consists almost entirely of cryptic abbreviations or numbers.

Languages can also be either imperative or object-oriented. The most common high-level languages are imperative, consisting mostly of commands that perform specific manipulations of defined objects. For example, you might begin with the declaration:

```
A, B, and C are numbers.
```

and then give the instruction

```
C = A + B
```

and the language will manipulate the numbers accordingly.

In contrast, object-oriented languages send messages to objects; the response depends on the object. To add two numbers, your instruction follows this form:

```
Message to object A: add yourself to object B.
```

When object A receives the "add yourself to" message, it is responsible for performing the instruction or complaining if it cannot; the programmer is freed from the detail work that the objects carry out. Mixing two different object types can create new objects that possess the attributes of both types, so object-oriented languages are suited to defining new procedures and recursive (repetitive) functions in which each step depends on the result of the previous step.

Source Code

The computer instructions, or source code, written in the early languages all come out in a flat, linear sequence—a long list of statements executed in order. All lines are or can be numbered, and the order of execution can be changed with instructions to go to another line. Changes in one part of the program often force changes in many other portions. The often complex interrelationships of instructions within these programs has inspired the term "spaghetti code," because figuring out how the programs work is so difficult.

Most modern languages, however, use a structured form. Programs written in these languages are divided into blocks, or logical subunits. Each block is a complete procedure and can be changed without affecting any other procedure. Blocks are usually set up by indentations in the source code, somewhat like an outline. The layout tries to push the programmer into thinking in an orderly way.

Interpreters and Compilers

Interpreters and compilers take high-level language instructions and turn them into low-level machine instructions. A language interpreter takes one line of high-level instructions and converts it into the low-level form the microprocessor can understand. The microprocessor carries out that line of instructions and hands control back to the interpreter, which converts the next line, and so forth.

Because it works a line at a time, an interpreter is less efficient than a compiler, which translates the entire high-level language program into a machine-language program before carrying it out. A compiled program will usually run much faster than an interpreted one. But developing and debugging a program is much faster with an interpreter because you don't have to stop and recompile the whole program in order to find out if it works.

(For an excellent discussion of computer languages, see *The Secret Guide to Computers,* Volume 2, by Russ Walter, 92 Saint Botolph St., Boston, MA 02116. The *McGraw-Hill Encyclopedia of Electronics and Computers* contains a concise, but highly technical essay on computer languages.)

LANGUAGES FOR THE MACINTOSH

A steady stream of languages has appeared for the Macintosh. The selection is varied; the choice is yours. See the end of this chapter for a source code comparison.

BASIC

BASIC (Beginners' All-purpose Symbolic Instruction Code) is the most common microcomputer language and remains a good choice for short, relatively straightforward jobs. BASIC was originally written by John Kemeny and Thomas Kurtz at Dartmouth College in 1964 as a simplified form of FORTRAN. BASIC is easier to learn than either Pascal or Logo.

In its original, unstructured form, BASIC works somewhat like a stream-of-consciousness process—quick and intuitive, but often a little untidy in actual execution. You will probably find it difficult to follow a BASIC program written by someone else, or even a program you wrote some time ago, unless you include unusually clear annotations in the source code.

BASIC has been a moderately well-specified language; there are industry standards for the instructions and syntax, but now there are two virtually distinct forms of BASIC: the old form and the new, structured versions.

Microsoft and True BASIC are similar versions of BASIC for the Macintosh. With either of them you can write the BASIC program in one window and see the program's output in another window. However, the BASICs are not compatible; you cannot write a program in one version, or dialect, and run it in another.

Microsoft BASIC is interpreted; True BASIC compiles to an intermediate code. Both are much easier to use than the BASICs for earlier computers. Unlike earlier BASICs, they use a structured design; they have no line numbering, and the source code uses indentations to mark the beginning and end of procedures, much like Pascal. These changes answer the most severe criticisms leveled against BASIC in the past.

Microsoft BASIC (version 2.0)

With the addition of structured programming, version 2.0 has changed greatly from earlier Microsoft BASICs. It can, however, run programs written for the earlier version, and remains compatible with MBASIC versions for other computers except for instructions that are specific to a particular microcomputer design—mainly variations in graphics and sound generation. You can thus run or adapt many programs from earlier books and magazine articles, and also write new programs in the structured form. The structured version will appear for other microcomputers too. If you look at books on BASIC, check for the version described. A possible exception to choosing version 2.0: If you have children who are learning BASIC in school on an Apple II, they may find it easier to start with the older Microsoft BASIC on the Mac.

If you write a program in Microsoft BASIC, you can choose from three distribution methods. You can distribute the original source code, but your recipients must have BASIC in order to use the program. Or you can use the run-time version of Microsoft BASIC; a run-time interpreter (also called run-time system) will run your program but the person using it cannot edit the program or write a new one.

Microsoft will offer a compiler for its BASIC, so programs developed with the interpreted version can be turned into much faster programs after completion; compiling will be the preferred way to prepare BASIC programs for commercial sale.

For programmers, specific features of Microsoft BASIC compared to the IBM PC version and Applesoft BASICs are listed at the end of this chapter.

True BASIC (pre-release)

True BASIC was written by Kemeny and Kurtz, the original authors of BASIC. It incorporates their current thoughts about the language and represents their first effort for microcomputers. A source code-compatible version of True BASIC is also available for the IBM PC; even graphics programs can run without change. True BASIC does not plan to release a compiler but is planning a run-time module. Its publisher, Addison-Wesley, is traditionally strong in textbooks, and will emphasize educational uses of the language.

Pascal

Pascal (named after 17th century mathematician Blaise Pascal) was designed by Nikolas Wirth, who based it in part on Algol, a language more popular in Europe than in North America. Pascal comes much closer than earlier languages to the academic concept of how a good computer language should work. Pascal popularized structured programming; it has been a better choice than BASIC for long programs, but its advantage is fading as BASIC becomes more structured.

Pascal standards are spelled out by the International Standards Organization (ISO). Although several dialects exist, the presence of clear standards inhibits the dialects from straying too far. With Pascal programming experience, you have a head start learning C, now the most popular development language for commercial microcomputer programs.

Macintosh Pascal

Developed by Think Technologies and published by Apple, Macintosh Pascal is the first commercial version of Pascal to appear in interpretive form. It is particularly suited to teaching the language, but runs too slowly for use in commercial programs. Think Technologies plans to offer a run-time system.

Lisa Pascal

A version of Apple's Lisa Pascal for the Macintosh. It is compiled and can run source code, with minor modifications, written in Macintosh Pascal.

UCSD Pascal

UCSD Pascal is a family of products originally based around a Pascal language and operating system developed at the University of California at San Diego. The publisher, Softech Microsystems, offers two versions. The Designer Series is the standard p-System brought over to the Macintosh. It can run ordinary UCSD Pascal and Apple II Pascal programs, but these programs generally do not make use of special Macintosh features because p-System programs are designed to run on many microcomputers and use resources common to all. The second version, MacAdvantage, is a modified UCSD Pascal designed to use the Macintosh operating system and special features. (In May 1985, Softech withdrew from the microcomputer software business.)

Modula-2

Nikolas Wirth wrote Pascal as a teaching language; he went on to develop Modula-2, which is similar, as his "real" language. Modula-2 cleans up rough edges in Pascal and adds missing elements, such as the ability to compile a program in separate modules. Modula-2 also offers features interesting to advanced programmers. For example, it supports a form of parallel processing, in which the execution of one segment of a program can be halted to allow another to run, so it can serve as a real-time control language. With real-time control, a program can be interrupted to respond immediately to specific events, such as a message from a peripheral device. Like UCSD Pascal, Modula-2 uses a compiler to create an intermediate code that is then interpreted to run the program. The Modula Corporation (Provo, UT) provides a Modula-2 for the Macintosh.

C

C (written after the language B) is a middle-level language, developed by Dennis Ritchie at Bell Telephone Laboratories for use in writing the UNIX operating system. C is structured and uses memory efficiently, and finished programs run quickly, but it is fairly difficult to learn. It is currently the most popular language for serious microcomputer software development.

On the Macintosh, you can choose from many different C compilers. Fortunately, comparisons are beyond the scope of this book.

Logo

Logo was originally written by Don Bobrow and a group at MIT headed by Seymour Papert. It is best known as a language for children because its graphics are very easy to learn. Logo is, however, a full programming language. For complex programs, Logo is harder to use than BASIC or Pascal; progressing beyond the simplest steps is difficult. Because Logo is an extendable language (you can define new procedures within the language), there is no standard form.

Two versions of Logo are available for the Macintosh. One, Logo for the Macintosh, is published by Microsoft and was written by Logo Systems in Montreal, the company founded by Papert. Logo for the Macintosh is interpreted, following the original concept of Logo as a language designed more for ease of learning than for computational speed.

ExperLogo from ExperTelligence is the first compiled Logo for microcomputers. It emphasizes graphics in its design and includes some 3-D graphics and LISP-like features. As a compiled language, ExperLogo is harder to use as an educational tool than interpreted versions. The finished programs run very fast—too fast for most teaching purposes, so it includes a deliberately slowed-down graphics mode (called "turtle" graphics) in addition to the fast mode ("bunny" graphics).

LISP

Experience with Logo can be applied to LISP (LISt Processing), a language developed specifically to deal with lists and recursive definitions (definitions that are nearly circular, building upon themselves), two functions hard to do with traditional languages. You can define and manipulate lists of numbers, text, or other lists. With long text strings to compare and recombine, LISP uses up memory quickly; several megabytes of RAM are not too much. LISP was written for, and remains largely limited to, academic and industrial research into artificial intelligence. LISP was the precursor language for Logo. ExperTelligence has a version of LISP for the Macintosh called ExperLISP and plans cross-compilers for the Symbolics 3600. David Betz has written XLISP and put it into the public domain.

Forth

Forth was originally written by Chuck Moore to control telescope-positioning motors. It was initially called Fourth, for fourth-generation language, but the IBM 1130 used for development would only accept five-character identifiers.

Forth works with a stack of variables organized in reverse-Polish notation, a non-algebraic form of mathematical expression in which the operator (plus, minus and so on) follows the numbers it affects—for example, 23+2 is written as 23 2 +. Reverse-Polish notation is the same notation used on a Hewlett-Packard calculator. Programming often consists of keeping track of where you are on the stack; since the stack can be many thousands of variables deep, this can get complicated. But for the skilled programmer, Forth program development is quick, especially for numerically intensive computations. Forth programs are hard to read and modify but use memory efficiently. Forth is moderately well-defined.

Creative Solutions offers MacForth in three different forms. Level 1 is the basic language; Level 2 adds an in-line assembler, floating-point arithmetic, and more graphics features. Level 3 is aimed at commercial software developers.

Micromotion's MasterForth is based on Forth-83, described in *Mastering Forth* by Anita Anderson and Martin Tracy (Brady, 1984), and includes a macro-assembler; it is available for several other microcomputers.

Neon

Neon, from Kriya Systems, is an object-oriented language that lets you define different routines, including those in read-only memory (ROM), as objects; you can then work with the objects by giving single commands. Neon is an extendable language modeled after Smalltalk and was written in Forth; a Forth interpreter is built in.

COBOL

Since 1960, COBOL (COmmon Business Oriented Language) has been the most widespread language for business applications on large computers. Unlike most other early languages, which were designed to deal with numbers, COBOL was designed to deal with text and records; it looks more like English than other languages do. Critics of COBOL are legion, charging that COBOL programs are hard to maintain and modify and that use of the language continues out of inertia. Consider COBOL only if you work with existing programs that are not available in another language.

MicroFocus sells Mac COBOL for the Macintosh. With it, existing COBOL installations can use the Macintosh as a freestanding development tool.

FORTRAN

A language from the 1950s, FORTRAN (FORmula TRANSlation), like COBOL, has also seen its time come and go. It has been in a slow, steady decline for more than a decade. FORTRAN was originally developed for solving mathematical problems and remains entrenched in universities and research centers as the main language for scientific computation. As with COBOL, use FORTRAN only if you work with existing software.

MacFortran, from Absoft, is a full FORTRAN 77, including complex numbers and 32-bit integer operations. Although supplied on disk, MacFortran's compiler supports the Macintosh Toolbox and can also handle FORTRAN 66 source code.

Another FORTRAN 77, from Softech, runs under the p-System.

APL

Kenneth Iverson developed APL (A Programming Language) about 1960 for manipulating arrays of numbers. Especially good for use with numbers arranged in matrices, it is used for engineering and scientific tasks. APL source code is unusually compact; the only control statement is a GOTO command. There are odd-looking symbols in the source code, which make APL programs distinctive and also inspire the jibe that APL is a write-only language—once you have written a program, no one can figure out how it works. Long an interpretive language for mainframe computers, APL has appeared on microcomputers in recent years. For the Macintosh, Portable Software (Cambridge, MA) supplies PortaAPL.

Graphical Languages

A new class of programming languages may emerge on the Macintosh. They are languages that use icons instead of text. Odesta Helix has some of these features, although it's sold as a database, not a programming language. For years, instructors have taught programming students to begin with a flow diagram before writing a program. Hardly anyone ever does, but with Helix the flow diagram itself is the program—a development that could attract wide attention in educational circles.

Assembly Language

Because high-level languages are so different from the machine code that a central processing unit (CPU) can execute, programs written in high-level languages often can't take full advantage of a CPU's power and they usually run slowly. Programs written directly in machine language would optimize speed and efficiency, but machine code consists purely of numbers and is much too tedious and difficult for most people to write. Assembly language, a low-level language close to machine language, but easier for humans to work with, offers a solution by replacing the numeric codes with mnemonics. Even so, only the most determined programmers write in assembly language.

The Macintosh Assembler/Debugger from Apple gives the assembly-language programmer access to the programming tools built into the Macintosh's ROM. You can choose from alternate debuggers. In one mode, the Assembler/Debugger operates when two Macintoshes are connected; one Mac runs the program under development; the other runs a diagnostic program to track the operation of the first machine. Another mode uses just a single computer. Mainstay and others also offer assemblers. Some programming languages include an assembler for writing assembly-language instructions.

Disassemblers do the opposite of assemblers, taking the existing program code and displaying the assembly-language equivalent. MacNosy from Jasik Designs (Menlo Park, CA) is an interesting disassembler that automatically follows logical paths. It works on ROM programs as well as on conventional disk-based programs; it's for experienced assembly-language programmers only.

For adequate speed and flexibility, most commercial programs for the Macintosh are written in a combination of compiled code (mostly C or Pascal) and assembly language. Apple has not yet specified a standard format for compiled code, so you may not be able to link modules written in different languages. However, companies that offer several languages often make sure that modules written in any of their languages will be compatible.

THE MACINTOSH TOOLBOX

For effective operation on the Mac, any program should be able to use the Mac Toolbox, the programs in Mac's ROM. But programming-language manuals contain only sketchy information about the Toolbox at best. The standard reference comes from Apple as *Inside Macintosh* (Addison-Wesley, 1985). For more accessible discussions, however, see the two volumes of *Macintosh Revealed* by Stephen Chernicoff (Hayen, 1985) and *The Macintosh Developer's Guide* by William G. Nisen and Dennis F. Brothers (Addison-Wesley, late 1985 or early 1986).

DEMONSTRATION PROGRAMS

The simple programs that follow in this section do exactly the same thing: They read a number, A, from a disk file, and then they read a number, B, from the keyboard. Each program adds the two numbers to find C and then draws a horizontal line the length of C.

The authors of these programs were asked to make them comparable, rather than fancy. These samples hardly show the nuances of the languages used, but you can compare them and get a glimpse of them in action.

```
'Drawline--Microsoft Basic 2.0 program by Philip W. Marshall
OPEN "Drawline Data File" FOR INPUT AS #1
INPUT #1, A
INPUT "Enter a number", B
C = A + B
LINE (0,0) - (C,0)
END
```

```
10 REM Microsoft Basic 1.0 program by Philip W. Marshall
20 OPEN "Drawline Data File" FOR INPUT AS #1
30 INPUT #1,A
40 INPUT "Enter a Number: ",B
50 LET C = A + B
60 LINE (0,0) - (C,0)
70 END
```

```
! True BASIC program by Brig Elliot
!
open #1:name "MyFile"
input #1:a
input b
let c = a+b
plot 0,1;c,1
end
```

```
program Drawline;
{ Macintosh Pascal program by Philip W. Marshall }
 var
  A, B, C : integer;
  DiskFile : file of integer;

begin
 reset(DiskFIle, 'Drawline File');
 read(DiskFIle, A);
 write('Enter a number: ');
 readln(B);
 C := A + B;
 moveto(0, 0);
 lineto(C, 0);
end.
```

```
MODULE Drawline;
(* Modula-2 program by Donald L. Cohn *)
FROM InOut IMPORT ReadInt, ClearScreen, WriteString, WriteLn,
     OpenInput, CloseInput;
FROM QuickDraw1 IMPORT PenSize, MoveTo, LineTo;

VAR
 a, b, c : INTEGER;

BEGIN
 ClearScreen;
 WriteString("Respond to the prompt below with the name of a file");
 WriteLn;
 WriteString("containing an integer.");
 WriteLn;
 OpenInput("TEXT");
 ReadInt(a);
 CloseInput;
 WriteString("Give an integer. ");
 ReadInt(b);
 c := a + b;
 ClearScreen;
 MoveTo(0, 0);
 PenSize(1, 4);
 LineTo(c, 0)
END Drawline.
```

```
/* C program by Greg O'Brien */
#define SMALL_MEM
#include     <inits.h>
#include     <quickdraw.h>
#include     <dialog.h>
#include     <pb.h>
#define NIL (char *)0

char  numbuf[16] ;
char  volName[16] ;
long  count ;
int          a,b,c ;
int          item;
short type  = 0 ;
short vRefNum ,
             fd ;
DialogPtr dp;
Rect  box ;
Handle       itemPtr = 0 ;

main()
{

      InitGraf(&thePort);
      InitWindows();
      TEInit() ;
      InitDialogs((ProcPtr)NIL);
      SetCursor(&arrow) ;

      /* Read a number from a file */
      GetVol(volName,&vRefNum) ;
      FSOpen("\Pdatafile",vRefNum,&fd) ;
      count = 3 ;
      FSRead(fd,&count,numbuf) ;
      FSClose(fd) ;
      a = atoi(numbuf) ;

      /* Read a number from the keyboard */
      dp = GetNewDialog(7, (Ptr)0, (WindowPtr)-1);
      /* Read in value from text box */
      do {
            ModalDialog((ProcPtr)NIL, &item);
      } while (item != 1);
      GetDItem(dp,3,&type,&itemPtr,&box) ;
      GetIText(itemPtr,numbuf) ;
      DisposDialog(dp);
      b = atoi(&numbuf[1]) ;
      c = a + b ;

      /* Draw a line of pixels on the screen */
      MoveTo(0,0) ;
      LineTo(c,0) ;

}
```

```
TO DRAWLINE
; Mac Logo program by Kerry E. Lynn
; Draws a horizontal line of specified length in upper
; left hand corner of graphics window.
 Clean
 SetWSize "Graphics [200 200]
 Open "MyFile "#Disk
 SetRead "MyFile
 Make "A ReadWord
 Close "MyFile
 Type [Enter a positive integer :]
 Make "B ReadWord
 Make "C :A + :B
 PenUp SetPos [-100 100] PenDown
 SetHeading 90
 Forward :C
END
```

```
(defun drawline (&aux a)
  ;;ExperLISP program by Dexter Pratt
 (moveto -230 -75) ;top left of default graphics window
 (with_open_read "ExperDemo:disk-data"
       (setq a (read)))
 (print "input a value for B -> ")
 (line (+ (read) a) 0)
```

```
SCR # 15        "MyBlocks"   04/16/85   03:00:16 PM
( MacFORTH program by Kerry E. Lynn )          ( 041685 KEL)

VARIABLE MyFileNum
VARIABLE A        VARIABLE B        VARIABLE C

: ReadFile ( --- | read an int from file "MyFile" into var "A" )
 NEXT.FCB MyFileNum !      ( next available file number )
 " MyFile" MyFileNum @ ASSIGN ?FILE.ERROR
 MyFileNum @ OPEN          ?FILE.ERROR
 4 MyFileNum @ SET.REC.LEN  ( MacFORTH ints are 4 bytes long )
  A 0 MyFileNum @ READ.FIXED  ( read record 0 into var "A" )
  MyFileNum @ DUP CLOSE REMOVE ;

-->


SCR # 16        "MyBlocks"   04/16/85   03:00:28 PM
( MacFORTH program by Kerry E. Lynn )          ( 041685 KEL)

: ReadKB ( -- n | return an int from keybd, no range checking )
 0 >IN ! 0 BLK ! QUERY          ( get string from keybd )
 BL WORD BL OVER COUNT + C! NUMBER ; ( convert it to a number )

: Drawline ( --- | draw a line in corner of interpreter window )
 GINIT  PAGE
 ReadFile
 ." Enter a positive integer : "   ReadKB B !
 A @ B @ + C !
 0 0 MOVE.TO C @ 0 DRAW.TO ;
```

```
\ DRAWFILE, a NEON™ program by Reese Warner, Kriya Systems, Inc.

File aFile

basicStr aStr
new: aStr

: DRAWFILE { fileName \ a b c -- }
   get: fileName name: aFile
   interpret: aFile -> a
   ." Please input keyboard value for line length  "
   #in -> b
   a b + -> c
   cls
   10 10 gotoxy
   c 0 line
;

\ " TEST" put: aStr
\ aStr drawFile
```

```
        Program Drawline
!       Fortran 77 program (MacFORTRAN) by Paul S. Linsay
!
        Include Toolbx.par
        Integer*4 A,B,C
!
        Open(UNIT=1,FILE='Drawline.dat',STATUS='OLD')
        Read (1,'(i2)') A
        Type 'Enter B '
        Accept B
        C=A+B
        Call Toolbx(MOVETO,5,25)
        Call Toolbx(LINE,C,0)
        Pause
        Close(1)
        End
```

```
     ∇DRAWLIN
      ⍝ APL PROGRAM BY BILL WESTLAND
      'DRAWFILE' ⎕FHTIE 1 ⋄ A←2 ⎕FREAD 1 ⋄ ⎕FUNTIE 1
      B←⎕
      C←A+B
      MOVETO 1,1
      LINETO 1,C
     ∇
```

Now, here's a segment of an assembly-language program that performs the same simple task (the first 34 lines have been omitted):

```
; 68000 assembler program by Greg O'Brien
;     Read a number from a file
;
      pea    vRefNum(A5)      ; where to store the volume ref. number
      pea    volName(A5)      ; address of volume name string
      _GetVol                 ; get volume reference number
      add.w #8,a7        ; clear arguments from the stack

      pea    fd(A5)           ; where to store file descriptor
      move.w     vRefNum(A5),-(a7); which disk volume to read
      pea    Filename(A5)     ; name of file to open
      jsr    _FSOpen          ; open the file
      add.w #10,a7            ; clear arguments from the stack

      move.l      #3,count(A5)       ; number of characters to read
      pea    numbuf(A5); where to place characters that are read
      pea    count(A5)   ; how many characters to read
      move.w       fd(A5),-(a7); specify file to read from
      jsr    _FSRead               ; read from file
      add.w #10,a7                 ; clear arguments from the stack

      move.w       fd(A5),-(a7); specify file to close
      jsr    _FSClose   ; close the file
      add.w #2,a7           ; clear arguments from the stack

      pea    numbuf(A5); where to get string
      jsr    atoi            ; convert ascii string to binary integer
      add.w #4,a7            ; clear arguments from the stack
      move.w       d0,A(A5)   ; store value in A
;
```

. . . by now you get the idea. The actual program continues on for another 42 lines.

And finally, here's the machine-language code. Again, length decrees that lines be omitted; this time, 437 of them:

```
0100111011010101
0000011001100110
0011011100110111
0010111001110010
0110110000100000
0110111001101111
0111010000100000
0110011001101111
0111010101101110
0110010000001101
```

MICROSOFT BASIC NOTES FOR PROGRAMMERS

If you want to move a program written in Applesoft BASIC to Microsoft BASIC on the Macintosh, you must write replacement graphics instructions for the following Apple II-specific statements: HTAB, VTAB, HGR, GR, HLIN, VLIN, HPLOT, HCOLOR, COLOR, HGR2, and PDL. Any PEEKs and POKEs in Applesoft BASIC will not work on the Macintosh.

If you want to move a BASIC program from the IBM PC to Mac, here are the statements in IBM BASICA (version 1.0) not supported by the Mac version: BLOAD, BSAVE, COLOR, COM(n) ON/OFF/STOP, DEF SEG, DEF USR, DRAW, KEY, LOCATE, MOTOR, ON (event) GOSUB, OUT, PAINT, PEN, PLAY, SCREEN, SOUND, STRIG, and WAIT. Similarly, the following IBM BASICA intrinsic functions are not present in the Mac version: CRSLIN, INP, SCREEN, STICK, and USR. Finally, these machine-specific statements/intrinsics will need modification: CALL, CLEAR, PEEK, POKE, and VARPTR.

Sectio

Understanding Macintosh

Three

How does the Macintosh hardware and software work? These chapters include descriptions of the inner workings of a Mac and its accessories, along with a comparison of the Macintosh and the IBM PC-AT, and a look into the future.

How Macintosh Works

Chapter 11

Computers process information—numbers, words, graphs, or nearly anything we would call information. To be practical, a computer must: take information in (input); manipulate the information in some way (processing); get the information out (output); and save the information (storage).

If you're curious about how the Macintosh does these things, you should read this chapter. Although the material is not essential to working successfully with your Macintosh, much of the terminology explained here appears in other chapters. We'll discuss the basic functions of the key hardware components and the software that tells them what to do. Once you have a general idea of how these elements work together, you can find more detail about specific hardware and software in the following chapters.

BUILDING UP A MACINTOSH

Let's start with the most basic components: a keyboard to enter information, a microprocessor to manipulate it, and a video screen to display the output.

Keyboard → Processor → Screen

Suppose you want to type the letter A. When you press the A key, the keyboard generates an electrical signal corresponding to the letter A. This signal is sent to the microprocessor, which, with associated components, turns it into a different electrical signal and sends it on to the screen. These steps are physically accomplished by Mac's hardware: the keyboard, microprocessor, video screen, and connecting wires.

Computer programs, or software, control all the hardware. Software is nothing more than a set of instructions for the microprocessor; it enables the microprocessor to understand the keyboard's signal for the letter A and to create the dot pattern that makes an A appear on your screen.

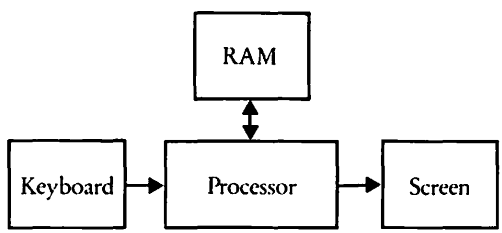
Memory

The hardware and software discussed so far constitute no more than a video typewriter—interesting perhaps, but not very useful. A computer needs to be able to move, copy, and otherwise work with the keyboard entries. To do this, it requires a kind of scratch pad—a place to keep the keyboard characters while manipulating them.

Random access memory (RAM)

The computer's scratch pad is an electronic storage area called random access memory, or RAM. RAM is fast; characters can be stored or retrieved in a microsecond (a millionth of a second). Random access means that the microprocessor can go instantly to any spot in the storage area for information, without having to look at any other part of memory first, and can then jump forward or backward to another spot without having to read any information in between. A special area, called video RAM, is set aside as a map of the screen. Software controls this area to produce the images you see.

RAM has one major limitation; it is transient. When the power goes off, anything stored in RAM disappears.

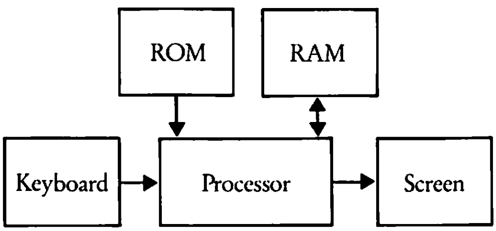


The Macintosh also has a separate, tiny RAM apart from the main RAM. This small, parameter RAM remains alive all the time with battery power. The parameter RAM stores time, date, control-panel preferences, and serial-port configurations (input/output specifications). Removing the battery will reset the parameter RAM; you will need to reset the clock and calendar afterwards.

Read-only memory (ROM)

Programs must be in electronic memory to instruct the microprocessor, but not all software operates from random access memory. Because RAM only offers temporary storage, some instructions—such as the program that tells the microprocessor what to do when the power comes on—can't stay there. These programs are held in another form of electronic memory called read-only memory, or ROM. ROM is permanently stored on a wafer of silicon called a chip.

In all computers, ROM contains the initial instructions for starting the computer. The Macintosh's ROM also contains essential programs for controlling how a disk drive works, for interpreting input from the keyboard and mouse, and for drawing graphics or text on the screen.

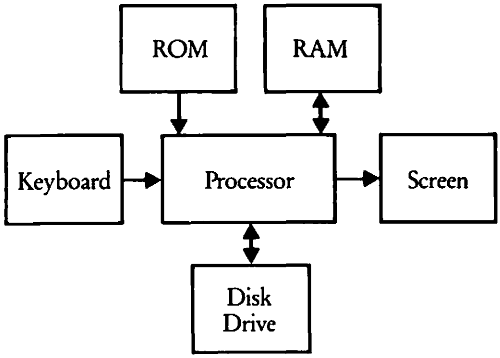


From the microprocessor's standpoint, ROM is simply another information source, just like RAM, and even a little faster. But whereas RAM is transient, ROM is fixed. Once the computer leaves the factory, its ROM is permanent, whether the power is on or off; the only way to change it is to replace the ROM chip itself. Because ROM is thus not as permanent as hardware, ROM programs are sometimes called firmware.

Storage

To store large amounts of information permanently, or at least for long periods, microcomputers use disk drives that can read and write information on magnetic disks, just as a tape recorder can play and record sound on tape. Like a tape recorder, a disk drive stores information as a series of magnetic pulses, except that on a disk the pulses are arranged in concentric circles.

Although a disk drive can read and write much faster than a human, it is far slower than RAM; finding a character on disk can take a disk drive a few seconds, compared to RAM's microseconds.



Because of this tremendous speed difference, all microcomputers normally use RAM for active work and disks for permanent storage, exchanging information between the two as needed. When you finish working with one block of information, or file, you tell Mac to store it on disk so that its RAM is free to work with another block of information. Once a disk location is found, a disk drive can read and write continuous information at the rate of 60,000 characters a second.

Like text or graphics information, computer programs can also be stored in disk files. Before a program may be used, it must be read (temporarily copied) into RAM for fast, effective operation. When you insert a MacWrite disk, for example, and use the mouse to select the program, the computer transfers a copy of the MacWrite instructions from the disk into RAM. If you quit MacWrite and change to MacPaint, the MacWrite instructions are replaced with the MacPaint instructions. But in both cases, the programs also remain stored on the disk.

Now let's look briefly at how information is coded and how it travels through the computer.

BITS AND BYTES

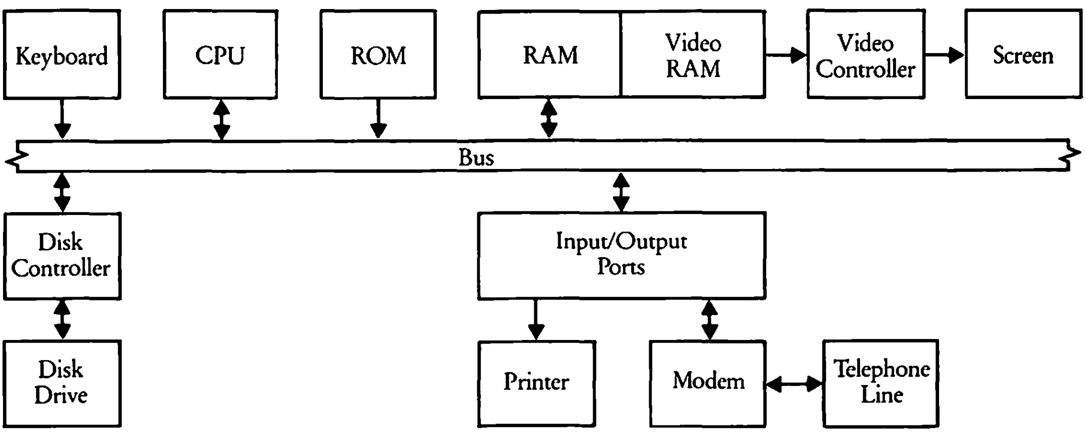
A computer can only process information it understands. Computers understand electronic signals with just two states: on and off. They process information as individual on/off signals, or bits, coding each bit as 1 (on) or 0 (off).

One bit can't convey much information, so a computer strings many bits together to create something useful. A single character (a letter of the alphabet, a number, or a punctuation mark) is coded by eight bits in sequence, or one byte. The letter A, for example, is 01000001, B is 01000010, and so on. Each hardware component—keyboard, memory, disk drive—codes the letter A the same way.

The information content of a single byte is still limited, so much of the time we use larger units, kilobytes. A kilobyte, or 1 KB, is 1024 bytes. Although "kilo" ordinarily means 1000, a kilobyte isn't an even 1000 bytes because the computer's counting system is based on the number two, not the number ten. Two multiplied by itself ten times comes out to 1024.

Disk-file size is customarily measured in kilobytes. A file 6 kilobytes long contains about four pages of text, or 6144 characters (a typical double-spaced typewritten page holds about 1500 characters). A 6-kilobyte file doesn't have to consist of characters, though; it can be a program or a picture of equivalent length.

A few other measures: 1024 kilobytes equal 1 megabyte; 1024 megabytes equal 1 gigabyte. Usage is erratic, however; a megabyte is sometimes defined as 1000 kilobytes, a gigabyte as 1000 megabytes. In most cases, the difference is immaterial.



THE BUS

Coded information travels through the computer on a bus, a set of wires serving as a data highway that links the computer's components together. Each component communicates with the microprocessor via the bus.

The bus carries two main kinds of information: One group of wires conveys the actual data, such as the coded letter A; another group of wires carries the address of the component to which the data are headed. Each component accepts only information addressed to it; for example, information intended for the printer will not inadvertently be accepted by the disk drive.

Most of the action on the bus is orchestrated by the computer's central microprocessor; we'll take a quick look at that next.

THE CENTRAL PROCESSING UNIT (CPU)

The heart of every microcomputer is a single integrated circuit chip—the microprocessor, or central processing unit (CPU). The Macintosh's CPU chip is a Motorola 68000.

The CPU is powerless by itself. To do anything useful, it carries out, one at a time, the step-by-step instructions provided by software. An instruction might read: "Take the information stored in memory location 125, add 1, and put the result in location 240." Or: "Take the character placed on the bus by the keyboard and put it in memory location 300." Each step is simple; computers do useful work because they can perform millions of steps in rapid succession.

A CPU's power depends on three factors:

- How much information it can work on at once, measured two ways:

  How many bits (called input/output, or I/O, bits) the CPU can take from and put onto the bus at a time—16 bits in the 68000.

  How many bits the CPU processes internally at one time—32 in the 68000.

- How many different kinds of instructions it can perform.
- How fast it operates—how much time each instruction takes.

By current standards, the 68000 is a powerful chip, but the differences among chips are less important than advertisements claim.

THE MANY FORMS OF SOFTWARE

You can already see that software comes in many forms, some built-in and some available on disks. Some essential software tells the microprocessor how to read a disk; other software can draw a picture of a disk. The next several sections describe the differences.

ROM Software: Key to the Visual Interface

Mac's 64 KB ROM contains the key to its operation. Apple's Lisa, and now the Macintosh, were the first microcomputers to have such a comprehensive collection of programs in ROM. These programs provide the power behind Mac's visual interface.

The ROM software controls the interface; it draws key parts of what we see on the screen, monitors the mouse, and does much more. It therefore defines the way that we deal with application programs—the word processors or spreadsheets we use for our work. ROM programs make up a kind of programmer's tool kit, to be used by both professional and amateur programmers. This unique interface is the reason different Macintosh applications work much the same way.

The ROM programs include the following:

- QuickDraw draws complex graphics on the screen quickly.
- The Font Manager uses QuickDraw to create typefaces on the screen.
- The Event Manager keeps track of what we do with the mouse and keyboard.
- TextEdit is a basic text-entry and editing program.
- The Window Manager draws and controls windows on the screen.
- The Control Manager creates and monitors the dialog boxes and our choice of buttons within the boxes.
- The Menu Manager creates and monitors the pull-down menus.
- The File System creates and controls files in memory and on disk.

Other programs control other essential operations.

The Operating System: A Traffic Cop

A fundamental program called the operating system acts as traffic cop, keeping track of and directing all Macintosh operations. It manages everything in memory and keeps track of information going to and from each component, whether the disk drive, printer, keyboard, or screen.

In conventional microcomputers, the operating system is read in from a disk and stays mostly in RAM when the computer is working. Some popular operating systems are Apple DOS (Disk Operating System) for the Apple II; CP/M-80 (Control Program for Microprocessors); and MS-DOS (Microsoft Disk Operating System).

In contrast, most of the Macintosh operating system (which has no name) resides in ROM. The rest is stored in a file named System on the disk you use to start Mac. This file is read from the disk into RAM when you first turn the computer on; it adds to or modifies the ROM instructions.

The System file also contains information such as the specific keyboard layout. Storing this information on disk makes changing to a foreign-language keyboard easy; Apple merely changes the System file used with foreign versions of the Macintosh.

The System file also contains many other programs, including:

Utility programs. Some of these are nearly as important as the operating system; others are simply handy to have available. When you select a file icon and then choose Get Info from the File menu, for example, you are actually starting a small utility program that checks that file and displays information about it.

Desk accessories. The menu under the Apple symbol (far left on the menu bar) lets you choose functions such as a clock, a calculator, or a note pad. Because these programs are short, they can usually share RAM space with an application program.

Font data. This information dictates the font—the actual shapes of the letters you see on screen. Several fonts are essential to Mac's operation—the ones you see in the Finder and main menus, for example. Many additional fonts and type sizes are also available from this file when you are using MacWrite, MacPaint, and other programs. To create these fonts, the operating system transfers the information from disk to RAM. If you change fonts or greatly change the type size, you may have a short wait while the Font Manager program (in ROM) goes back to disk to bring the new information into RAM.

Because information about each font takes up considerable disk space, you may want to use the Font Mover program to store rarely used fonts on a separate disk.

Messages. Both warning and advisory messages also reside in the System file. For foreign versions of the Macintosh, Apple changes this file to give messages in another language. (The ROM programs contain no text in any language.)

The Finder: Keeping Track of Disk Files

Another important program stored on disk and read into RAM when you first turn Mac on is the Finder. The Macintosh Finder includes many functions traditionally performed by a computer's operating system. It handles most operations that involve disks: creating the disk window with its file icons, copying files, copying disks, and so on. The Finder doesn't work alone; it uses many programs in ROM for actual disk access, in effect acting as liaison between you and the ROM programs that control the disk drive.

Each disk has a directory that functions as its table of contents. The directory contains the file headers, which hold information about each file on that disk. When you insert a disk, the Finder puts this directory information into RAM where it remains, even if you change disks. You can display the directory from an ejected disk on the screen, but if you want to use a file from it, the Finder asks you to change disks.

The Desktop file

For each disk, the Finder creates a hidden Desktop file to hold information about each file on the disk. The Desktop file "remembers" such on-screen details as the icons associated with the file and the size and format of the disk's window on screen. It also notes whether a file is an application program or a document; if it's a document, the Desktop file checks the file's header for the application program that created it, and keeps track of the icon images required by the program. See Chapter 19 for more details on the file structure.

Clock/Calendar

Mac has a battery-operated clock/calendar whose time and date are read into RAM. Every time you create or modify a disk file, the date is automatically stored with the file header. This time-and-date keeper drives the desk-accessory clock as well.

Application Programs: Your Primary Tools

You use application programs for doing work—MacWrite to produce written documents, MacPaint to create illustrations, and Microsoft Multiplan for financial calculations. Permanently stored on disk, application programs are read into RAM when you need them. Because these programs are large and complex, usually only one at a time is in RAM. If necessary, the operating system moves programs or files from RAM to disk to make space for the application.

Even such maneuvering cannot free enough space for some programs, however. Some sophisticated applications, such as Lotus's Jazz, are simply too large to fit into RAM all at once, even on a 512 KB Mac. These programs operate with a core program that stays in RAM, plus components called overlays that remain on disk until needed. As you select functions—a sorting routine or trigonometric calculation, perhaps—the core program brings needed overlays into RAM; each new portion replaces other overlays not currently in use.

Although the overlay procedure lets you use powerful programs, it also slows operation. But as Macintoshes gain more memory, overlays become less necessary; a large enough amount of RAM can contain an entire application, and the program will run much faster. With enough memory, even several application programs can be in RAM at the same time, and you can switch applications instantly, without waiting for disk drives to read the new program into RAM.

THE MANY USES OF RAM

A Macintosh has far more activity going on in RAM than does a conventional microcomputer. RAM holds a lot of information:

- The video memory.
- Parts of the operating system.
- Utility programs.
- Desk accessories.
- Current font data.
- Icon images.
- The Finder.
- Disk directories.
- An application program.

RAM also contains two forms of data:

- Data used in the application program.
- Data on the Clipboard.

Data: Your Information

The point of all this software, of course, is to do something with your information.

You can enter your data into the computer through the keyboard, mouse, or disk drive, or from another computer, over a telephone line or through a network linking the computers together.

From an application program's standpoint, the data source doesn't matter because, in most cases, the program will put the data into RAM before beginning work. Some programs must have all the data in RAM at one time (RAM-based data); other programs can read some data into RAM and leave the rest on disk, swapping chunks as needed (disk-based data).

The Clipboard

Once some or all of your data are in RAM, whenever you Cut or Copy anything from the screen, that information goes into the Clipboard, an area of RAM set aside to act as a holding area for information exchange between programs. For example, you may want to cut a series of numbers from a Multiplan spreadsheet and paste it into a MacWrite document, so you can include a financial statement in a memo.

You can store text, a drawing, or numbers in the Clipboard, but you can only store one item at a time. If you need to store more items, paste the Clipboard contents into the Scrapbook; this frees the Clipboard for another item.

WHAT HAPPENS WHEN YOU START A PROGRAM

To tie all these software components together, here is a brief outline of what happens when you start MacWrite. The outline is not complete and events don't occur quite so linearly, but you'll get an idea of how the Macintosh system works.

When you turn Macintosh on, a ROM program called Boot tells the microprocessor to check whether a disk has been inserted in the drive. (The term boot comes from the idea that the computer is pulling itself up, or on, by its own bootstraps.)

If there is no disk, the Boot program puts an image of a disk with a question mark on the screen, in effect asking you to insert a disk.

Once a disk is in the disk drive, the program instructs the disk-controller circuitry to send electronic signals to the disk drive to move the disk-drive head to the disk's outer edge and begin transferring information from the disk into RAM. First, the System file containing the RAM portion of the operating system is read from the disk. Mac then reads in the Finder from disk and creates the Finder display.

To find out what program and data files are stored on the disk, you select the disk image by clicking it with the mouse; you then choose Open from the File menu. The Finder creates a window showing the file icons and names. You move the mouse so that the pointer is over the MacWrite icon. The Event Manager (in ROM) detects the mouse position. You double-click on the icon. The Event Manager tells the Finder about the clicks. The Finder checks the pointer location and concludes that you want to open the MacWrite file.

The Finder checks the disk directory for the location of the MacWrite file and passes the file location to the disk controller, which starts turning the disk and moves the disk-drive head over the beginning of the file.

As the disk-drive head reads the MacWrite file, the disk controller puts the information on the bus. From the bus, the information passes into RAM, in space allocated by a ROM program called the Heap Manager.

Once in memory, MacWrite begins changing the screen. It replaces the Finder menu bar with the MacWrite menu bar, showing the MacWrite selections.

Almost simultaneously, the Window Manager (in ROM) puts a window on the screen, complete with scroll bars and title.

The Font Manager (in ROM), which has been busy creating the text on the screen, also checks over the System file to see which fonts it contains. It passes the number and available sizes of each font on to the Menu Manager (in ROM), which sets up the Fonts menu.

As you can see from this incomplete outline, even something as simple as starting a program requires an enormous number of steps. That the procedure works at all is amazing; that it works so well is a tribute to thousands of engineers and programmers who, during the last 50 years, have made computers possible.

The next chapters take a detailed look at Macintosh components.

The Video Screen

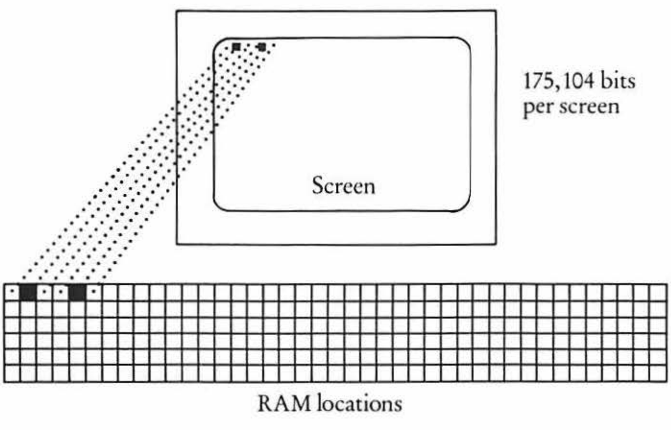
Chapter 12

Mac's screen is a cathode-ray tube (CRT) like an ordinary television screen, but with much higher resolution and a much sharper image. The image is made up of 175,104 dots (512 horizontally and 342 vertically), called picture elements, or pixels.

MAPPING THE SCREEN

All the information needed to generate the screen image is stored in the special part of random access memory called video RAM. This section of RAM uses a little more than 21 kilobytes (175,104 bits divided by 8 divided by 1024).

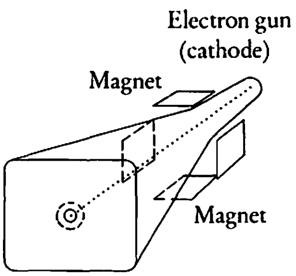
Each pixel corresponds, or is "mapped," to a bit in video RAM; in computer jargon, the screen is bit-mapped. To generate the screen image, Macintosh software takes the display information stored in video RAM and computes which pixels should be lit up.



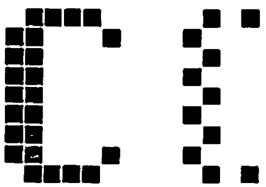
Bit-mapped screen: Each dot on the screen corresponds to a bit in memory.

Drawing bit-mapped images calls for a lot of computing. Every time you change anything on the screen, the computer must recompute each pixel of the changed object (it usually does not recompute stationary items).

CREATING THE SCREEN IMAGE



Electrons strike phosphor, creating light.



The letter C shown as normal and dimmed.

Once all the screen bits have been computed, a video-controller circuit creates a video signal suitable for the screen. The video signal controls a beam of electrons, turning it on or off. The beam, pointed toward the inner side of the screen, passes between electromagnets that direct the electrons' path. When the beam is on, the electrons strike a phosphor coating on the CRT face, and light is produced for that particular pixel. (Historically, electron beams were called "cathode rays," after the device, a cathode, that produced them.)

The images you see are thus made of pixels that are either on or off. The Macintosh has no true grays, although objects can be made to look gray by turning off alternate pixels.

To keep an image constant on the screen, the electron beam sweeps across it repeatedly. Macintosh's circuitry refreshes the screen image more than 60 times each second, frequent enough that your eye and brain see a continuous image. If you turn up the screen brightness, however, or look at the screen out of the corner of your eye, you may see a slight flicker because human peripheral vision is especially sensitive to movement.

The Macintosh has a nonstandard video signal; it displays lines 50 percent faster than standard video systems and unlike ordinary televisions, which generate images by drawing every other line 60 times a second, the Macintosh draws *every* line 60 times a second (in jargon, the Macintosh produces a non-interlaced image). These differences mean that you cannot record or broadcast the Macintosh video signal directly with ordinary television equipment, to make a videotape for training, for example. You can, under some circumstances, use other video equipment with Mac; for more information see Chapter 28.

LARGER SCREENS

The 9-inch Macintosh screen has a 4¾- by 7-inch (12- by 18-cm) image area—small by most standards. Use of larger screens with the Macintosh is possible only by opening the case and redirecting the video signal. Caution: Only experienced technicians should attempt this; again, see Chapter 28 for more information.

COLOR SCREENS

Mac will not generate color video images. A color CRT with Macintosh resolution can cost as much as the computer and still will not produce images as sharp as the built-in black-and-white screen.

A few companies have produced high-quality color CRT display systems that can work with any microcomputer. From the computer's standpoint, the display acts as an output device, just like a printer or a plotter. To drive the display, you must find suitable software or write your own program.

Some color displays have large memories and sophisticated internal processing; some models display nearly 700 by 500 pixels and boast a palette of 16 million colors. Prices range from $2500 to $6000. Unfortunately, these special displays do not work interactively the way the Macintosh does with its own graphics; you can't use a mouse with these screens. As the market grows, however, display companies will develop special software designed to work with the Macintosh.

Some software already built into the Macintosh ROM can handle color and will be pressed into service on future Macintosh models.

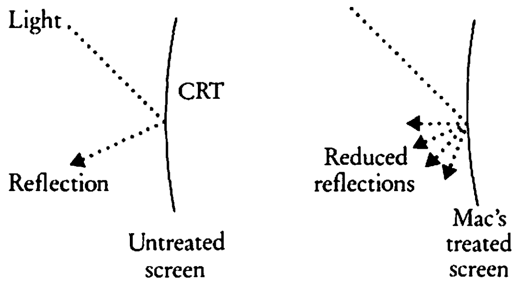
LOOKING AT THE SCREEN

Some people find that looking at a computer screen for a long period strains their eyes. Usually these problems can be eased or eliminated by careful attention to installation.

If the Macintosh rests on a typing-height table, you may find the screen a little low. You can raise the Macintosh or, better, tilt it back a little to get a more comfortable viewing angle. (Don't tilt it back too far without adding a fan, because the ventilation holes are designed to let the warm air rise vertically.) Several companies make adjustable tilting platforms that fit under the Mac. But since you only need a fixed angle, a block of wood works just as well and is a lot cheaper.

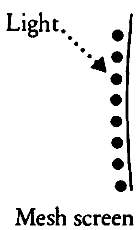
Glare

Because Mac's screen surface is slightly roughened to reduce reflection, you don't need to set it up in a darkened room. Just look for a place out of direct sunlight; the brightness of the screen and the surrounding area should be about the same.



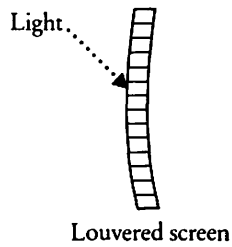
Room lighting is a frequent problem. The most common office lighting, with large fluorescent fixtures directly above the work area, can cause discomfort for users of video displays. Any bright objects—walls, clothing—facing the screen can produce a reflection. If this happens, you may want to put up a dark wall hanging opposite the screen to suppress such reflection. If you are planning an installation, the room should have indirect lighting or, at least, lighting a short distance away from the computer. Desk lamps or lamps on flexible arms near the computer can illuminate work areas without flooding the CRT with light. Large windows can be tamed with screen accessories (see the following sections) or by controlling the light with room dividers, horizontal venetian blinds, or better yet, vertical louvers that can be adjusted to block the sun and leave a view.

Still, you may not find a place for your Macintosh that avoids glare completely. If your room has a big picture window, it would be a shame to give up the view so you could stare at a computer screen. An accessory anti-glare screen may help, though some may darken too much. These screens come in four forms:



Mesh screens ($10 to $20) use closely woven, matte-black filaments to scatter room light from any direction. Because of Mac's small pixel size, the mesh must be very fine. You will probably need to turn up the brightness, and the image will be a little less sharp. Clean dust from the mesh by blowing on it.

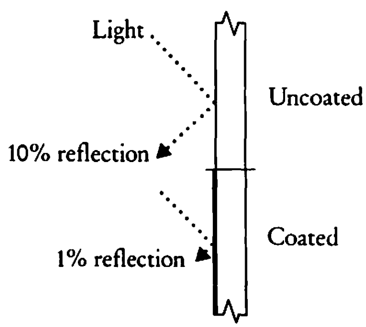
Louvered screens (from 3M, about $25) use many tiny, parallel, horizontal strips embedded in a plastic sheet to control light reflection. If you have strong lights directly over your Macintosh, one of these screens may solve your glare problem. The strips force you to view the screen from a narrow vertical angle, awkward for long sessions with the computer; sharpness suffers a little, but brightness is not reduced much. The narrow viewing angle also provides some privacy from prying eyes. Clean with any mild window cleaner.

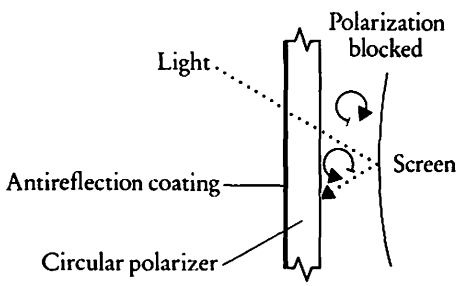


Optically coated screens use anti-reflection coatings of the type found on camera lenses and other optical surfaces. The flat, polished glass does not degrade the image, but strong light sources can occasionally create disturbing reflections despite the coating. A small adjustment in the screen angle usually fixes the problem.

The less-expensive interference coatings ($40) work best with a darkening (neutral density) filter, which cuts screen brightness.

The best cure for screen glare, however—and the most expensive—is a circular polarizing filter, which dramatically cuts reflection. The plastic versions (Polaroid CP-50, Kensington Microware, and others; about $50) work but aren't durable. The best type, made of optical glass, costs over $100. Suppliers include American Hoechst (Vu-Tek filters) and Polaroid (CP-70). Clean these filters with lens cleaners only.





Green and Amber Screens

Some companies claim special "ergonomic" advantages because of a green or amber screen, supposedly more restful for the eyes. There is actually little evidence one way or the other; the Macintosh uses black characters on a white background to correspond with black ink on white paper.

Some self-appointed experts have called for regulations to require that screens be green or amber—an unjustified step. They might just as well advocate that all books be printed on green or yellow paper.

Green and amber filters designed to fit over Mac's screen are available. If you don't mind the fuzzier image through the filter and are convinced that the results are more restful, you could try a color filter.

TAKING CARE OF THE SCREEN

Aside from occasional cleaning with a household glass cleaner (unless you have an accessory filter), the Macintosh screen should require little maintenance.

One problem may develop if you leave the screen illuminated for weeks or months without actively working with it. When an electron beam strikes a screen area continuously, the phosphor there eventually wears. Stationary images left for long periods may thus "burn in," leaving a faint shadow. (You can see this wear on airport television screens that show flight schedules 24 hours a day.) To reduce phosphor wear, you should turn down the brightness control whenever you leave your Mac turned on, but will not be looking at the screen for a long time. Or use a desk accessory program from Silicon Beach Software to black out the screen after several minutes' inactivity; you click the mouse button to restore the screen image.

The Keyboard and the Mouse Chapter 13

To use most computers, the keyboard is usually sufficient for entering data and giving commands. To use Mac's visual interface, the keyboard and a pointing device—the mouse—are essential. These and other input devices are discussed here.

THE KEYBOARD

Like any English-language typewriter, the Macintosh keyboard has the standard QWERTY layout, named for the first row of letters. A typewriter usually has 84 or 88 characters, including the capital letters; like most computers, your Macintosh adds the symbols < > | \ ~ and ^, for a total of 94 characters.

Special Keys

In addition to the character and the familiar Shift (capital-letter) and Caps Lock (all-capitals) keys, the Macintosh keyboard also has two special shift keys, Option and Command (⌘), which work like the Alternate and Control keys on other computers: You hold down one (or both) of the keys while pressing another key to get special functions. Another key, Enter, works somewhat like the Return key, but with a difference described shortly.

The Option key

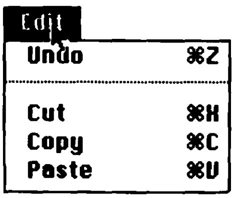
You generally use the Option key to get symbols or special characters for a foreign language. You can see them with the Key Caps Desk Accessory.

¡ ™ £ ¢ ∞ § ¶ • ª º – ≠
œ ∑ ´ ® † ¥ ¨ ˆ ø π " ' «
å ß ∂ ƒ © □ ∆ □ ¬ … æ
Ω ≈ ç √ ∫ ˜ µ ≤ ≥ ÷

Option-[	“	
Option-Shift-[	”	
Option-]	‘	
Option-Shift-]	’	
Hyphen	–	*(en-dash)*
Option-hyphen	-	*(ordinary hyphen)*
Option-Shift-hyphen	—	*(em-dash)*

The Command key

You use the Command key as a shortcut to some menu commands. Some pull-down menu commands have a (⌘) symbol and a letter; holding down the Command key while pressing the letter produces that command immediately—just as if you had used the mouse to pull down the menu and choose an item. For common commands, you may find the Command key faster than using the mouse.



Unfortunately, the Command-key combination for a particular command can vary from program to program, but most software does use Apple's choice of keys (Command-x for Cut, Command-v for Paste, and so on).

Because the Macintosh lacks special function keys, some programs with many commands require use of three keys at a time, such as Command, Shift, and a letter key.

Control characters. When communicating with other computers, you may need to generate control characters. Control characters perform such functions as ringing the other computer's bell (Control-g) to wake up the operator or telling the other computer when to start a new page (Control-l). To generate these characters on Mac, the communications program linking your Macintosh with other computers makes the Command key the equivalent of a Control key; to produce a Control-c, for example, you will type Command-c.

The Escape character. Macintosh software doesn't normally need the Escape key common on other computers. If you're hooked up to another computer and need to generate the Escape character, your communications program will tell you how to send it. Some programs define the tilde (~) key as Escape (to send a tilde character, you may use Command-tilde); others simply use Command-[, because on conventional computers Escape is equivalent to Control-[.

Enter

Pressing the Enter key usually marks the end of an entry made from the keyboard—as when you fill in a blank in a dialog box. To give software developers flexibility, Apple has not rigidly defined the Enter key; its function varies from one program to another. In some, the Enter key acts like the Return key except that the pointer moves in a predefined way after Return but not after Enter. In Multiplan, for example, the cell selection moves down one step if you press Return, but remains in place if you press Enter. In most dialog boxes, pressing Enter accepts the default response (generally the response with a double outline).

How the Keyboard Works

The keyboard contains its own 8-bit processor, an Intel 8021, programmed for Apple. The 8021 contains its own ROM and RAM and scans the keyboard for activity every 3 milliseconds. All but the shift keys are wired in a 9-by-6 matrix; the shift keys (Caps Lock, Shift, Option, and Command) are each wired separately. Once a key is pressed, the 8021 notes the matrix location and whether one or more shift keys are also depressed. Then the keyboard code is sent on to the CPU, where a keyboard device-driver program looks up the code and defines the appropriate character. The keyboard has two-key rollover: If you hold down one key and depress a second, the keyboard can still decode both keys correctly in sequence.

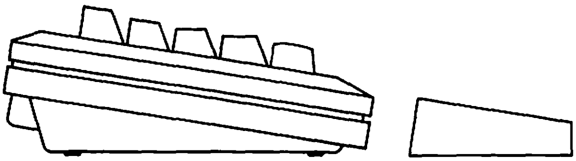
See Apple's manual for keyboard adjustments available through the Control Panel desk accessory.

Keyboard enhancer

Mac Tracks (first released version from Assimilation) lets you assign keystrokes and menu selections to command keys. You can assign boilerplate text, such as a return address, to a command key; typing the command key alone gives you the entire address. Or you can assign menu selections to command keys. You can even string a series of commands and text together; with a single command key you can add "Power to the Revolution!" at the bottom of a letter, save the file to disk, open a new one, and begin with "Victory to the Counterrevolution!" Mac Tracks runs as a desk accessory within a single application; you cannot string commands together to quit one program and start another. You cannot edit a Mac Tracks command-key sequence; you must redo all the steps. Check Mac Tracks for compatibility with your application program, however; some programs may not work with it.

Palm Rests

The Macintosh keyboard is a little taller than other computer keyboards, particularly European ones. If you need support for your hands, almost any thin support will do, or you can make one out of wood. Commercial palm rests are also available.



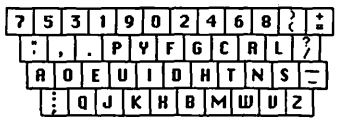
Keyboard Palm rest

The Dvorak Keyboard

The standard QWERTY layout used on the Macintosh keyboard was developed by Christopher Scholes in 1873 to slow typists down. The layout, which separated often-used letters from one another, prevented typists from getting ahead of the typebar mechanism in the first Remington typewriter. The QWERTY layout is thus the oldest component of modern computers.

With this century's emphasis on production-line efficiency, many alternative layouts have been proposed to improve speed and accuracy. August Dvorak devised the best-known alternative. He analyzed letter frequencies and letter-pair frequencies in English and placed the most often-used letters so they would be pressed by the most powerful fingers using the least motion. He also analyzed number frequencies, laying down the number row as 7531902468. This rearrangement proved too much even for Dvorak enthusiasts, who have mostly restored the 1234567890 order.

Dvorak keyboard with the original number-key assignments

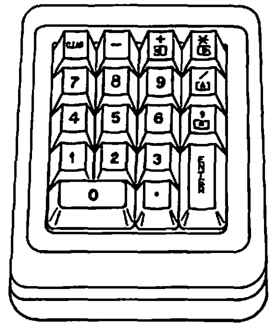


Although perhaps not optimal, the Dvorak layout is clearly superior to QWERTY. Speed improves some 5 to 25 percent with about half the errors made with QWERTY. The world's typing speed record (170 words per minute) was set on a Dvorak keyboard.

Experiment with Dvorak if you're curious; Dvorak layout programs are available for the Mac, but you must change all the caps on the keys, obliterating the standard arrangement. Since even Dvorak advocates claim only a few thousand Dvorak keyboards in existence, however, choosing Dvorak means choosing isolation.

Dvorak is probably a lost cause. The QWERTY keyboard is like irregular verbs: Everyone hates them but no one can change them.

The Numeric Keypad



The optional numeric keypad adds a calculator-like keypad, handy if you work with numbers frequently; it plugs in between the normal keyboard and the system unit, and has its own 8021 processor.

Macintosh programs don't normally require the 16 additional keys, but the Macintosh can distinguish between a number pressed on the main keyboard and the same number pressed on the keypad. Some programs take advantage of this ability by letting you reprogram keypad keys to generate specific characters or commands; the keypad then works like the function keys of other computers.

Four cursor keys on the keypad move the cursor, or pointer, on the screen. Not all software recognizes these keys, however.

Taking Care of the Keyboard

The keyboard and keypad will need little, if any, special care. The most common threat to both is spilled coffee. Many companies have a sensible rule against placing coffee or other drinks on any table occupied by a computer.

You might buy or make a cover to keep dust off. Although many commercial covers are available, you can make a perfectly good one from coated ripstop nylon sold by outdoor/sporting goods stores.

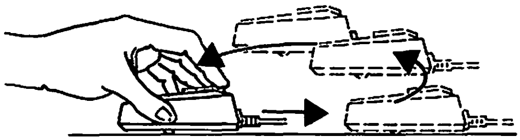
For situations where liquid or debris may fall into the keyboard, such as on a shop floor, flexible, transparent keyboard covers (Viziflex Seels) are available from Multiflex Seals. Such covers inhibit touch typing, but simple keyboard entries are still possible even with the cover in place.

THE MOUSE

The keyboard is necessary for entering text but, when using any computer, you are pointing at the screen much of the time—choosing a menu item, marking text when word processing, aiming rockets in a game, or drawing.

The most satisfactory general-purpose pointing device remains the mouse. Many studies have shown its superiority, but people are often skeptical until they try one.

Learning to use a mouse takes virtually no time, and once you have worked with it, you will appreciate its rapid, natural operation. You don't have to look at a mouse while operating it, and it stays in place when you let go; your arm generally rests comfortably on the desk. If you run out of desk space in the middle of a movement, you simply pick the mouse up (the pointer stays where you left it on the screen) and use the same space again to continue your movement.



Moving a mouse over the same desk space

Disadvantages of the Mouse

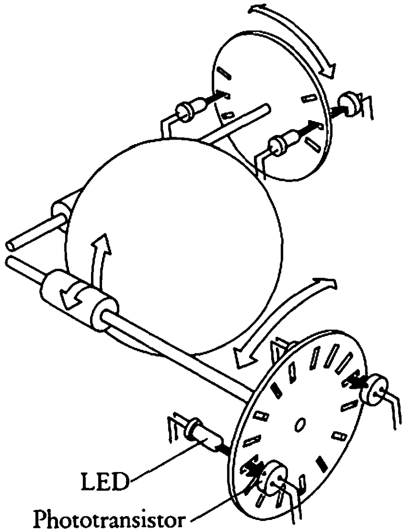
A mouse is not ideal for every application. With a word processor, for example, using the mouse to move the pointer on screen means taking your hand away from the keyboard, making a movement with the mouse, and moving your hand back to the keyboard again. The effort is worthwhile for moving a long distance, but small moves are simpler with cursor keys, so many users will want the optional numeric keypad, provided the software is compatible.

The mouse is not ideal for drawing, either, because you can't coordinate your hand and the mouse as well as you can your fingers and a pen. Skill does come with practice, though; see Chapter 17 for more information.

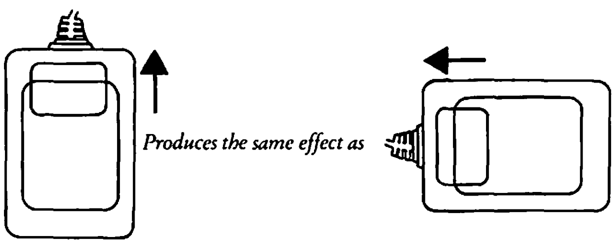
A mouse requires extra desk space, which may be hard to come by in a crowded office—but then, five years ago you probably didn't have space for a computer, either.

How the Mouse Works

Mac uses a mechanical/optical mouse; the roller on the bottom is mechanically coupled to two rotating vanes with slits. The slits in the vanes interrupt beams from light-emitting diodes (LEDs) that light up phototransistors. As the beams are interrupted, the two vanes track vertical and horizontal motions with respect to the screen. Detecting the direction of motion is a little tricky. The LEDs and phototransistors are in pairs; because of a small offset, the lead or lag of one phototransistor's output with respect to the other gives the direction (in jargon, quadrature modulation).



The vertical and horizontal orientations are defined by the mouse, not your point of view. If you hold the mouse at an angle and move it straight back (vertically), the pointer moves at a corresponding angle.

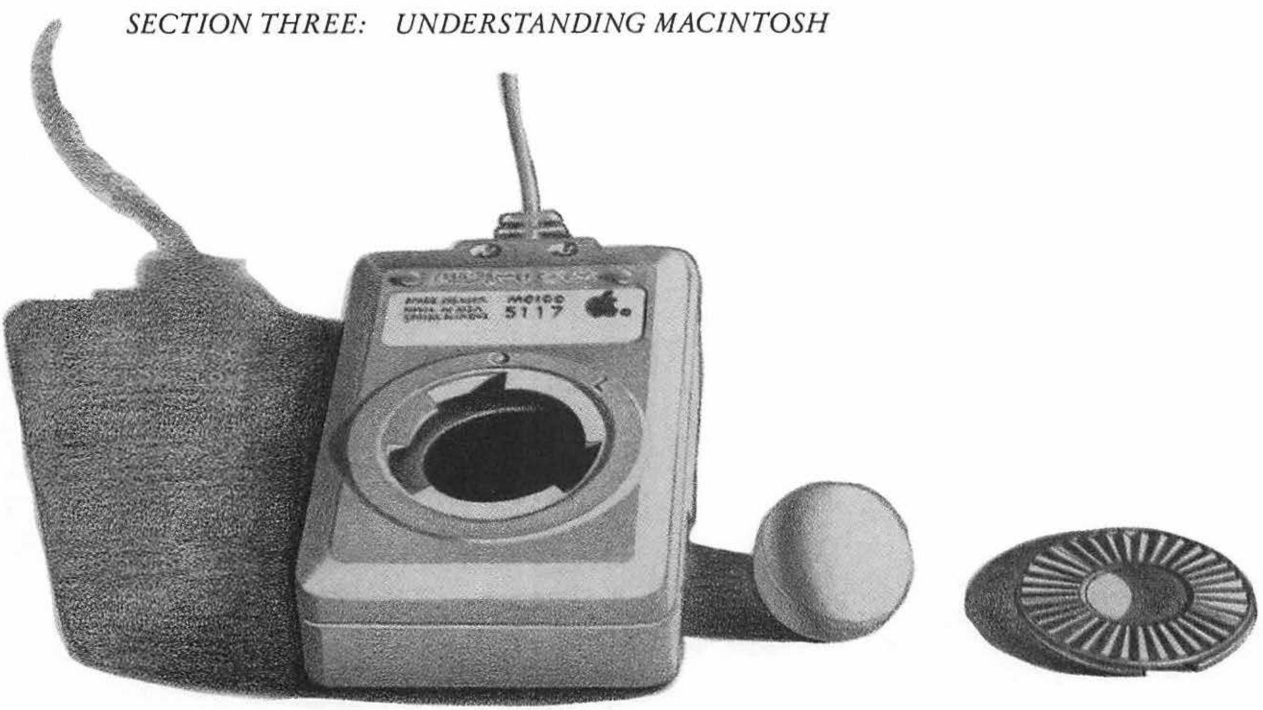


Mouse movements depend on the absolute orientation of the mouse.

A mouse registers relative movement; the controller electronics can tell how far the mouse has moved and in which direction, but not the mouse's absolute location.

Taking Care of the Mouse

The Macintosh mouse works best on a clean, reasonably smooth surface. If necessary, use a pad on the work surface (a low-cost, matte-finish plastic place mat works fine). A rough surface can wear down the bottom of the mouse. For the same reason, when you hold the mouse, don't bear down hard.



The roller can get clogged with debris picked up while rolling over a dirty surface. Even a little dust and oil will produce erratic mouse action. If your mouse falls ill, turn it over and undo the ball-retainer ring. The most likely problem is dirty rollers inside the ball chamber. Clean them carefully with a cotton swab or cloth moistened with alcohol. Then clean the ball, brush and blow out the ball chamber, and reassemble the mouse.

OTHER POINTING DEVICES

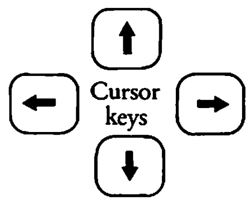
Because pointing is so fundamental to computer operation, designers have developed many different ways other than the mouse to do it. Most types of pointing devices are now available for the Macintosh as accessories.

Alternate Mice

In addition to Apple's mechanical/optical mouse, you might find two alternatives useful. Mouse Systems has an optical mouse (the A+ mouse). Instead of rollers, infrared LEDs and photosensors track the mouse position over a special pad inscribed with vertical and horizontal grid lines. Optical mice are less prone to getting clogged, so they often work better in locations, such as factories, where debris is a problem. On the other hand, the pad is sometimes a nuisance. The feel is different; an optical mouse offers much less frictional resistance to motion. Many users may prefer the feel of the A+ mouse to the original Macintosh mouse.

MacNifties has announced an optical mouse that can use graph paper as the pad, and a wireless mouse that uses infrared light. The need to change batteries will offset the wireless convenience in many cases.

Cursor Keys



Older computers have generally provided cursor keys for moving the pointer around the screen.

By far the most common pointing device, cursor keys are better than a mouse for making small movements, but they are awkward for longer ones; you have to bang the keys repeatedly or wait for an automatically repeating key to move the cursor where you want it. To shorten the wait, however, some programs have set up additional keys to move the cursor a word or a page at a time. Cursor keys are hopeless for drawing, giving no physical sensation of how far you've moved.

As mentioned earlier, the numeric keypad for the Macintosh has cursor keys, although many programs do not recognize them. A few programs such as Microsoft Word have also defined four command keys on the main keyboard as cursor keys.

Touch Screens

Touch screens boast the most natural operation of all; you point at something by actually touching it on the screen. Different touch screens use a variety of electronic techniques to locate your finger.

Touch screens are rare because they don't really work well. They have a place in computer systems that must be simple enough for a passerby to operate, as in a store display, but for most uses they present intractable problems. A finger is too big to select a character on the screen, much less a pixel; few people will put their fingers in a pencil sharpener first. And while actually pointing at something, you can't see it because your finger is in the way. Moreover, fingerprints soil the screen. These problems are bad enough when word processing, but they are untenable for drawing, when you must have precise pixel-by-pixel control.

OK, say some companies, we'll simply offset the cursor so it will always be half an inch above your finger. Now you may be able to see the cursor, but you must still hold your hand up to the screen while jockeying the cursor around—or while you are thinking—a tiring proposition.

A touch screen for the Macintosh is possible, most likely with a larger, external CRT.

Light Pens

A light pen is a pen-shaped device containing a light receptor that is activated when you press the pen against the screen. The receptor detects the electron beam from the CRT as it moves across the screen (see Chapter 12), and a timing circuit uses the beam to locate the pen. Most light pens won't work in the black areas of the screen. (The system may have to reverse the video briefly to locate the pen.)

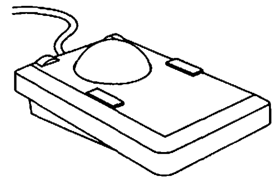
Light pens share many of the same problems as touch screens, including arm fatigue; imagine holding a pen up to a vertical screen while composing your next sentence. Moreover, you must pick up and put down the light pen each time you need it.

Light pens come in several forms and work well for some applications, particularly computer-aided design. A light-pen-like device might also be used with future computers with flat screens that fold down near the desk top. A light pen for the Macintosh would be difficult to build. As with touch screens, such a pen might work better on a larger CRT.

JOYSTICKS AND TRACKBALLS

The joysticks and trackballs used with arcade games work well for moving laser guns to shoot at attacking monsters but they lack the rapid, precise control essential to normal computer operation. Unlike a mouse, joysticks and trackballs give you little kinesthetic sense of how far you have moved.

Joysticks have few applications aside from games. Of the three joystick types, only the simplest has appeared thus far for the Macintosh; it merely has four switches (up/down/left/right). The other joystick designs are: a continuous-motion stick with two potentiometers so the stick stays where you left it; and the rare strain-gauge variety, which senses how hard you are pushing in any direction.



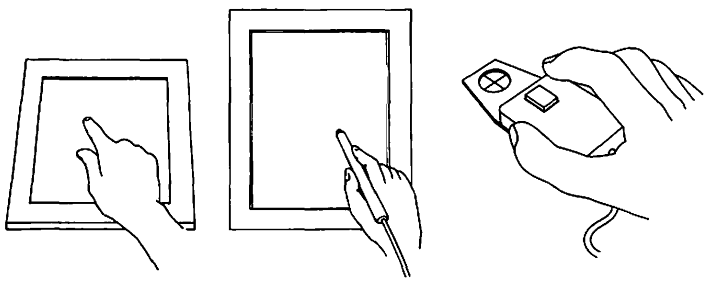
Internally, a trackball is essentially an upside-down mouse. Trackballs do save table space and aid in rare situations; some people find them easier for drawing smooth arcs in MacPaint. Several trackballs for the Macintosh have appeared. Interfirm Systems Corporation (Mountain View, CA) has announced McBall, which has two mouse buttons, one of them latching. The Mac Turbo Touch trackball (from Assimilation) is harder to use because it lacks a latching switch; you will need two hands to do the equivalent of dragging with a mouse.

Joysticks and trackballs can help for some physical handicaps; see Chapter 24.

Touch Pads and Digitizer Tablets

Touch pads, operated by a finger, share the resolution problems of touch screens. They also suffer from "skid patches" left by finger oil; movement is often erratic and fine control is difficult.

Digitizer tablets, which use a stylus, offer excellent resolution and a natural drawing action similar to holding a pen or pencil. Standard tablets work as absolute-position devices, much like a piece of paper, unlike mice, which only register relative positions. Good digitizer tablets are moderately expensive, and because you must continually pick up and put down the stylus, they are not generally a real substitute for the mouse.

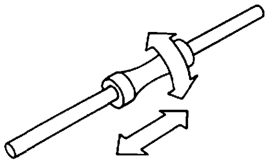


Touch pad, digitizer tablet, and puck

The Summagraphics MacTablet works well; it is an absolute-position device, so you can draw and trace on it easily. The stylus has interchangeable tips, including a ballpoint pen. An accessory puck aids in precision work with existing drawings. The Macintizer from GTCO (Rockville, MD) works in a similar way and connects to the mouse port. The Kurta Penmouse uses a wireless stylus and a tiny radio transmitter.

Koala has announced that its KAT tablet can be switched for relative or absolute motion. Although it is cheaper than the MacTablet, the working area is much smaller and less suited to intensive work.

Touchbars



Cheetah Control (Woodside, CA) has designed an unusual pointing system with a rolling bar that indicates vertical position and a slider that tracks horizontal position. Pressing on the bar or a separate switch activates the equivalent of a mouse button. The touchbar is compact and fairly easy to use with a word processor; it is not suitable for drawing.

Additional Pointing Devices

Several other pointing devices can help in special situations—if one hand is occupied with leafing through papers, if you are working on an assembly line, or if you have a physical handicap. A foot-operated, four-way switch can substitute for cursor or other keys. A voice-command system can accept spoken instructions.

Two companies have developed pointing devices that you direct by moving a small reflector mounted on your head or your glasses. An infrared light source and sensor on top of the Macintosh senses the reflector position and angle. Your natural head movements will need some retraining, particularly for side-to-side motions; most people move their head to read, not just their eyes. Although the systems can resolve to the pixel, they are more suitable for text than for drawing.

Stride Computer (Reno, NV) calls its unit the Nod and is seeking general uses for its product. Position Orientation Systems, Inc. (149 Cherry Street, Burlington, VT 05401) is specifically interested in aids for the handicapped; when you hold your head still long enough (the delay is adjustable), this system generates a mouse click. Personics has a head-mounted pointing system using ultrasound.

What about the most natural pointing technique of all—tracking where your eye is looking? Eye tracking is extremely difficult. The common systems follow the edge of the iris with photosensors and aren't accurate enough for most computer applications. Precision eye-tracking systems rely on specially designed—and uncomfortable—contact lenses. But even with such equipment, normal eye movements consist of slow moves and sudden shifts, which are too undisciplined for pointing.

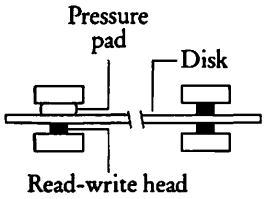
Disk Drives and Disks — Chapter 14

Computers use disk drives and disks for long-term information storage. A disk drive records and plays back digital information much as a cassette recorder records and plays back speech or music, but instead of recording on tape, the disk drive records on a thin plastic or metal disk. Disks are reusable; old information can be replaced by new information many times.

Disks come in two major forms (although there are intermediate types as well): A floppy disk is a circle of plastic, coated with a magnetic surface, that rotates inside a protective envelope. A hard, or Winchester, disk has a similar magnetic surface, but the coating is on a rigid aluminum platter. Hard disks are much faster and can store much more information than floppy disks, but they also cost more. I'll discuss floppy drives and disks first and then take up hard disks.

MACINTOSH MICROFLOPPY DRIVES AND DISKS

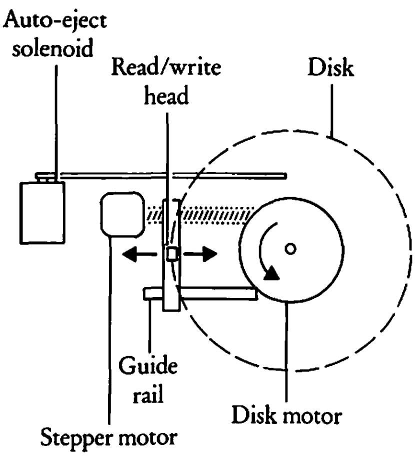
The Macintosh uses small 3½-inch microfloppy disks, developed originally by Japan's Sony. These little disks are far more convenient and easier to handle than earlier, larger floppy disks.



Single-sided disk drive uses a pressure pad opposite the read/write head (left); double-sided drives will use two heads opposite each other (right).

The first Macintoshes have single-sided disk drives that can store 400 KB (over 400,000 characters) of information on one side of each disk. Single-sided drives have one recording, or read/write head. Later double-sided drives (although not formally announced as this is written) will have two read/write heads and be able to store 800 KB on each disk. Single-sided disk drives are 50.8 mm high; double-sided disk drives will be about half as high, so the Macintosh case has room for two of them (but Apple may not put two in. . .).

When you insert a microfloppy into the Macintosh disk drive, a lever opens the disk's spring-loaded metal dustcover, exposing the magnetic surface. On single-sided drives, the head is on the bottom surface and it presses up while a pressure pad on the other side holds the disk against the head. On double-sided drives, the two heads will press on opposite sides of the disk. A motor then moves the head (or heads) radially across the disk, while another motor turns the disk at 390 to 605 revolutions per minute.

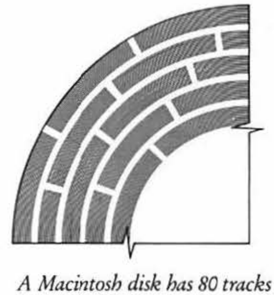


The disk-drive mechanism has two motors; one turns the disk, while a stepper motor moves the read/write head radially across the disk.

The read/write head records information as magnetic changes in 80 closed, concentric circles, or tracks. Disks thus resemble phonograph records in some ways, although a phonograph record has only one continuous groove.

The larger-diameter outside tracks are longer than the inside ones and can therefore hold more information. To take advantage of this, Mac's disk-controller circuitry turns the disk slower when the drive head is at the outer edge and faster when the head is at the inner edge, so the drive head writes more information on the outer tracks than on the inner ones. The changes in motor speed produce the different tones you hear as the drive operates. (Because they operate at a constant rotational speed, most disk drives for other computers store less information on each disk.)

Each radial track of a Macintosh disk consists of several sectors; the outermost 16 tracks have 12 sectors each, the next 16 tracks have 11 sectors each, and so on to the innermost 16 tracks, which have 8 sectors each. Each sector contains 512 bytes of information.



A Macintosh disk has 80 tracks on a side. The outermost 16 tracks have 12 sectors each; the next 16 tracks have 11 sectors, and so on, until the innermost 16 tracks have 8 sectors each.

Buying Disks

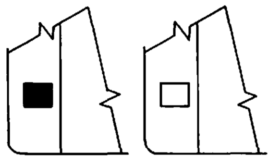
Despite their labels, single- and double-sided disks have magnetic coatings on both sides. The disk manufacturers test and certify each side. On some production lines, double-sided disks that fail on one surface are sold instead as single-sided disks. Since the test and certification process is more stringent than the demands of normal usage, such "single-sided" disks will generally work in a double-sided drive. But the small cost saving isn't worth the occasional failure; better, when they are available, to get certified double-sided disks.

Most microfloppies on the market are specified for 135 tracks per inch (tpi), the current standard, and the density at which Mac records information. In Japan and some other countries the earlier 67.5 tpi disks are also available; avoid them, since their failure rate at Macintosh's 135 tpi will be a little higher.

All the reputable disk brands work; differences in quality, if any, are hard to discern in normal use. If you are using the disks in extreme temperatures or climates, do some testing to make sure they are reliable. Under any circumstances, be cautious if you are offered unbranded disks at an unusually low price; defective disks do exist.

Some earlier microfloppies had slightly different dustcovers from those now being sold. Other than this small distinction, they work fine. If the cover of one of these disk latches opens, press with your thumb and forefinger on the arrow in the upper left corner labeled "pinch," and the shutter will close.

Write-Protection



If the small square is blocked, you can write on the disk. If the square is open, the disk is write-protected; you cannot write on it.

A common danger when working with any magnetic storage medium is writing over (erasing) important information on a disk. A small plastic insert (write-protect tab) in each microfloppy disk lets you guard against this. If you set the insert so you cannot see it from above, the disk drive cannot write on the disk. Some write-protect tabs are mildly inconvenient; you have to break off a plastic piece and reinsert it. The disk drive senses the write-protect tab mechanically, not optically.

When practical, you should set your original disks so they can't be written on and make working copies for daily use; you can't do this with some copy-protected programs, since you may have to use the original disk as the working disk. (Some programs require you to write on the program disk as you use them.)

Initializing Disks

When you buy disks to store your data, they are blank. Before using them the disk drive must initialize, or format, each disk by laying down magnetic markers for each track and each sector. If you are reusing a disk, initializing erases any information previously stored on that disk.

If you put in a brand new disk, or a disk that was used on another type of computer, a utility program in the System file puts a dialog box on the screen to ask if you want to initialize the disk. If you click the Initialize button, the drive goes ahead and initializes the disk; if you don't wish to initialize the disk—for example, if you realize you have put in the wrong disk by mistake—you must click the Eject button.

The Disk Directory

How does the Macintosh operating system know where on your disk to find a specific file? Several tracks store the disk's directory, an index to its files. The directory, together with a hidden Desktop file, holds the information that you see in the Macintosh Finder—each file name and its icon; the operating system uses the directory to find the file on the disk.

When you select a file to work with, the operating system first goes to the disk directory, locates the file name, and notes which tracks the file is stored on. It then moves the disk-drive head to those tracks and begins reading the file into RAM.

When you finish your work and wish to save it, or write it on the disk, the Macintosh first adds a directory entry, finds tracks available for storage (these tracks won't necessarily be in sequence), and then moves to each track in turn to record the information.

If you've run out of space on the disk, an alert box appears on the screen. In most cases (depending on the software you're using), you can switch the disk for another one with more storage space.

Speeding Up Disk Operation

Simply because it's mechanical, a disk drive operates more slowly than electronic RAM. In addition, a disk drive may perform many steps to read or write a file, moving back and forth between the directory tracks and the data tracks, which may be scattered about the disk in several places.

To speed up operation, the Macintosh keeps some disk-directory information in RAM, in an area called the disk buffer. Most programs write changes from the disk buffer to the disk, either immediately or at regular intervals (this is called "flushing" the disk buffer), but some programs may not take time to do this. That's why you can't simply remove a disk from the Macintosh, but must use the Finder to eject it; the latest directory information is automatically written on the disk before ejection. (On other computers, you can easily remove your disk at any time—including the wrong time.)

The speed improvement comes with some risk: A power failure will erase the directory information in the disk buffer. No computer is immune to power outages. Your best defense is to make frequent back-up copies of files and disks, or you can use uninterruptible power supplies, discussed in Chapter 20.

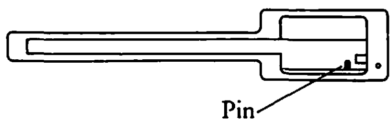
Erasing Files and Disks

When you want to erase a file, drag the File icon over the Trash icon so that the Trash icon reverses color, and then release the mouse button. The operating system responds by changing a single character in that file's directory entry to serve as a flag. The flag says, "Ignore this entry"; the file itself remains on the disk. The operating system only erases files when there isn't enough disk space for a new file or when you choose Empty Trash from the Special menu.

If you make a mistake in sending a file to the trash, you can recover it easily, but best do it soon: Open the trash (double-click the Trash icon) and drag the icon of the file you want to recover back to the disk window. The operating system changes the flag character back.

Under some circumstances, you can even retrieve a disk file after an Empty Trash command. To do so, however, you will need some understanding of the disk file structure, and you will need a program known as a sector reader. If you have a sector reader capable of searching for specific strings of text, you can even find whatever fragments of a file that remain after other disk activity.

If you cannot erase a file, it is probably locked; unlock it by clicking the "locked" box in the file's Get Info window. Occasionally, a disk file has further software protection against erasure, so you cannot send it to the trash bin. If you are sure you want to erase the entire disk, you can force the disk drive to erase and initialize the disk by pressing down on the small pin just inside the disk-drive slot before putting the disk in the drive.



You can also erase disks completely by using an audio or videotape demagnetizer on the microfloppy disk; when you put the disk in the Mac, you will be asked if you want to initialize it. *Caution:* Do not demagnetize hard-disk cartridges or future high-density (5-megabyte) microfloppies because essential formatting information recorded by the factory will be destroyed.

Macintosh Disk Compatibility

You can't use other 3½-inch disk drives with a Macintosh. Although Mac's disk-drive mechanism may look like the 3½-inch microfloppy drives in other microcomputers, it differs in both electrical and mechanical design. Because of these differences, the Macintosh stores information on disks in its own unique way, different from Hewlett-Packard, Data General, and others. The Macintosh cannot read microfloppy disks written by other computers, nor can they read Macintosh disks. The Mac can, however, read Lisa 2 disks with the help of a utility program.

The Macintosh disk-controller circuit is essentially the same as that in the Apple II and III, but adapting a 5¼-inch disk drive to Mac's controller will be difficult or impossible; not all the same electrical connections are supplied.

OTHER FLOPPY DISKS

Floppy disks come in several sizes. The original, and increasingly rare, 8-inch floppies are used mainly on older office computers. The 5¼-inch "minifloppy" is the most common on microcomputers. Both the 5¼- and 8-inch sizes have flexible plastic jackets with openings that expose the disk's fragile magnetic surface. Both require careful handling; a single fingerprint on the exposed surface or a warped jacket can destroy the disk.

The Sony 3½-inch microfloppy design has now defeated its small-disk competitors. (However, there is a new 1.85-inch disk standard designed for analog recording of still-video images.)

Taking Care of Disks

Although you should treat all disks with reasonable care, the microfloppy's semi-rigid plastic case and the spring-loaded metal dust-cover mean that ordinary handling will cause no damage; you can't put a fingerprint on the magnetic surface unless you go to the trouble of holding the spring-loaded cover open. Unlike older 5¼- and 8-inch floppies, you can write on a microfloppy label with a ballpoint pen without damaging the disk. Since a determined assault can ruin even microfloppies, however, keep them away from small children and pets. And even with the cover, disks will not survive coffee spills or direct hits with cigarette ashes.

If you put a rubber band around your microfloppies, wrap it horizontally. Otherwise the band can get caught in the cover, opening it and exposing the magnetic surface.

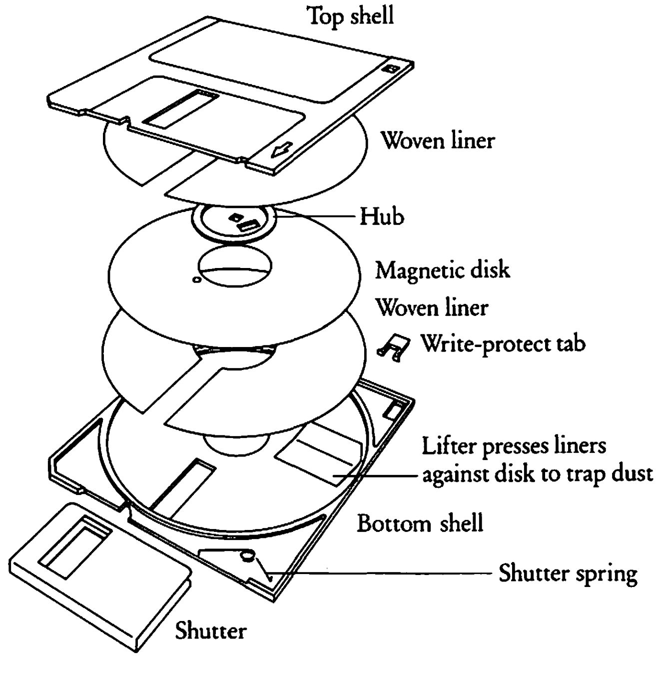
Any disk can be erased by a sufficiently strong magnetic field—over 50 oersteds (Oe). Fortunately such fields are rare; in a typical office, only the small magnets commonly used on memo boards or copy holders are likely to damage data. Other potential problems, such as fluorescent light transformers, are almost always too far away.

A small physical separation—three inches or more—is sufficient to eliminate nearly all magnetic threats, so the best insurance against magnetic damage is tidy housekeeping. Set aside specific areas on your desk for disks; don't throw them all over. Store them in a covered container for protection from dust. The many commercial disk-storage boxes are overpriced; file-card holders (4- by 6-inch size) and plastic shoe boxes work just as well. Magnetically damaged floppies can be reinitialized and reused.

The magnetic metal detectors at airport security checkpoints generate less than 5 Oe in most countries, so they pose no problem. X-ray inspection machines will not damage any computer component or magnetic media.

When shipping a disk, wrap it in a plastic bag; the cover and the edges of the case are not sealed, so dust can sneak through. If you use a sturdy shipping envelope, further reinforcement is usually unnecessary. Always make a back-up disk before shipping.

If a disk gets cracked, bent, or folded during shipping, all may not be lost. A soft disk liner provides some protection from the broken shell. Open up the shell carefully by breaking the corners away from the shutter with a thin knife. If the disk itself is creased or scratched, forget it. If not, transfer the disk carefully (use lint-free photographic gloves) to an opened but intact shell with liners; tape the shell closed and insert it into a disk drive. Transfer all the files immediately to another disk, and then throw the damaged one away.



Inside a floppy disk

Eventually, after a year or more of extended use, the magnetic surface on a disk will wear out. You are only likely to wear out a disk that you use constantly every day. If you have such a disk, make up a fresh working disk after six months and convert the old one to file-storage.

SHOULD YOU BUY A SECOND DISK DRIVE?

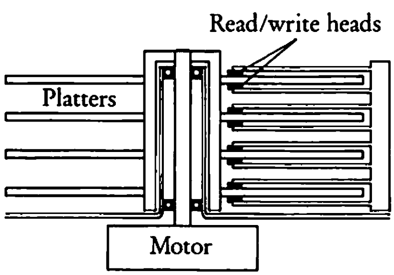
In a word, yes. With single-sided drives, a second drive is absolutely essential. Of course, you can get some work done with just one single-sided drive, but the 400-KB limitation is a severe problem. After storing essential system files and application programs, a single-sided disk doesn't have much space left for your data. A second disk drive solves this problem neatly.

With one double-sided drive, you will be able to get considerably more work done, but even so, you should get a second drive. Many routine tasks, such as copying a disk, are much easier with two drives.

You can add only one additional microfloppy disk drive to the Macintosh. The only practical alternative to a second floppy disk drive is a hard disk drive.

HARD DISK DRIVES AND DISKS

Hard disk, or Winchester, drives also record information magnetically and store it in concentric circles on a disk. But the disk is not made of thin, flexible plastic; instead, a hard disk is a rigid metal platter coated with magnetic material and polished to a mirror-smooth finish. Whenever the power is on, the platter spins continuously in a chamber of filtered air. Most hard disks have two or more platters permanently fixed in the drive.



Inside a hard disk drive

These features mean that hard disks are made much more precisely than floppies and can store much more information in the same amount of space. A microfloppy has 135 tracks per inch while a hard disk can have from 400 to over 900 tpi. The linear recording density of a microfloppy is 5000 bits per inch (bpi); on a hard disk it is 7000 to over 22,000 bpi. A micro-Winchester drive, which fits into the same space as Mac's microfloppy drive, can store 10 or 20 megabytes. Other hard disk drives, only a little larger, can store over 300 megabytes.

If you regularly use several programs and work with many documents, you will find a hard disk drive nearly essential. Hard drives cost more than floppy drives—about two to ten times as much—but when you consider the cost per character of information stored, they are much cheaper than floppies, and prices have been falling rapidly. Hard drives are more fragile than floppy drives, however, so if your computer gets rough treatment, you should probably stick to floppies. And hard disk drives all use fans for cooling, thus adding noise.

Connecting the Hard Disk

Hard disks come in many forms for the Macintosh. Each design has some disadvantages; many have problems working with some software. The most fundamental choice is whether the hard disk is bus-connected or port-connected.

Bus-connected

The best solution from a technical standpoint connects a hard disk through the 68000 CPU bus, because the drive can communicate with the computer at much higher speeds than it can through the ports. The peak data-transfer rate can be over a megabyte per second, but the real speed is limited by disk-access times (discussed shortly). Overall, bus-connected drives run 10 to 40 percent faster than port-connected drives. A bus connection also leaves the ports free for other uses—an important issue, since Mac doesn't have enough ports to begin with. The Macintosh must be modified to use bus-connected drives (potentially voiding Apple's warranty), but hard-disk manufacturers offer their own warranties.

There are two possible bus connections:

Direct to the 68000 CPU bus, internal. The General Computer HyperDrive is the first hard disk to mount internally. It overcomes the extra heat problem with a thermostatically controlled fan. Pro: Fast operation, compact packaging; you can boot from the hard disk with the additional ROM supplied. Con: Adds complex, fragile assembly to a computer not designed for such modifications. In case of problems, you cannot easily restore the Mac's original configuration.

Direct to the 68000 bus, external. Some companies may build adapter circuitry that fits inside the Macintosh and brings the 68000 bus out to an external connector. Pro: Fast operation; flexible design permits additional devices on the bus, including more memory; leaves ports intact. Internal modification is less drastic than with internal hard disk drives; you can restore the Mac almost to its original state by disconnecting the drive (not, however, if a supplementary boot-up ROM has been installed). Con: Voids the warranty; connector reliability is critical; potential problem with electromagnetic interference unless installation is done with care.

Port-connected

Port connections can be made in the following ways:

Via a serial port. Most hard disks for the Macintosh connect through the modem or printer port; many can be connected to either port. The more flexible designs can operate at a higher peak transfer rate (up to 112 KB per second) when connected to a single Mac, and at slow AppleTalk speeds (less than 29 KB per second) when connected to a network. Pro: No modification to the Macintosh necessary; all hard-disk designs can be used with AppleTalk, given suitable software, but those connected via a serial port are the easiest to adapt to AppleTalk operation. Con: Since there are only two serial ports, the drive displaces one unless it is designed to restore the port with an additional built-in connector. A drive without such a pass-through will not let you connect a modem and a printer at the same time, a significant problem. Drives with a pass-through port generally work on the printer port because handling modem information is trickier and a printer buffer can be built in easily—but the AppleTalk connection may not work.

Via the external floppy disk-drive port. The Apple 20 MB hard disk drive connects to the floppy-disk port; the drives will gain in performance with Apple's new ROM. The Quark hard disk also connects through the disk port; initial Quark models displaced the external floppy drive, but Quark plans a revised version that not only lets you connect the floppy drive but also a second hard disk drive. The Quark drive does not look like a floppy drive to the Mac, so you must boot the system with a floppy disk. Pro: Does not modify the Mac. Leaves serial ports free; you can boot up the computer with a drive of suitable design. Con: Speed is limited by port design.

Via the serial port through the AppleTalk network. This method yields the slowest performance. The effective speed depends on the amount of traffic on the network; with nothing but one Mac and one hard disk drive, the peak transfer rate is about 24 KB per second. With more computers and users, the speed drops. See Chapter 26 for more information.

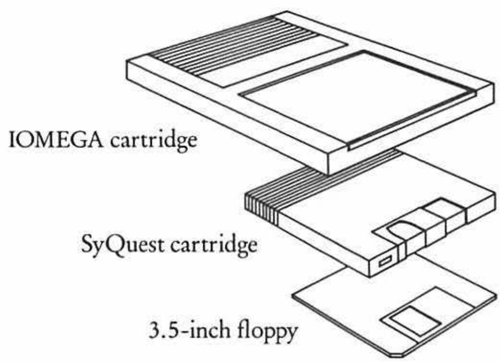
Other Issues

Additional considerations to bear in mind with hard disk drives fall into three categories: hardware, software, and speed.

Hardware issues

Hard disk capacity. Hard disk-drive capacities have been increasing quickly; Tecmar already offers a 140 megabyte drive for the Mac and other, even larger, disk drives will appear.

Fixed and removable drives. Most hard disks are permanently fixed inside the drive, but several designs put the disk in a removable cartridge housing. Cartridge drives are excellent when you have many files; to get more storage you simply add another $50 to $75 cartridge. If you are working with unfinished software, a cartridge lets you separate potentially troublesome programs from your important data. The main disadvantages to cartridges are lower storage density and somewhat higher price.



SyQuest makes the cartridges and drives (5- and 10-MB) used by both Tecmar and Micro-Design. The Bernoulli Box from Iomega uses an unusual 5-MB cartridge disk that behaves mechanically somewhat like a floppy drive but gives the general performance of a hard disk drive. Do not use bulk erasers on these cartridges!

Back-ups. With so much information stored on it, hard disk failure can be devastating. Failures are rare, but for safety you should copy or back up hard disk files regularly. Generally you only need to copy files that have changed since the last back-up so a floppy disk is fine if you create less than 400 KB (800 with double-sided disks) of new information a day. Floppies are inefficient for backing up accounting data bases and other large files, but cartridges are excellent for these back-up tasks; you will need a combination system—either two cartridge drives or one fixed drive and one cartridge.

However you back up your data, you will get added performance by having two fast disk drives—a much more potent combination than the traditional back-up medium, streamer tapes. Streamer tapes are long lengths of magnetic tape in cassette shells; they are expensive, slow, and useless for anything except back-up. However, if you have a very large hard disk drive and need to archive all of it regularly, only streamer tapes have sufficient capacity.

Mixing hard disks from different vendors. In some situations, you can use two different brands of hard disk at a time if the software drivers are compatible (check with the vendors; you may need to alter their supporting software slightly). On AppleTalk, you can install multiple hard disks.

Space and noise. All external hard disk drives take up space, but they don't have to sit on your desk. Macbottom fits underneath the Mac; other drives come with long enough cables that they can be put on a shelf (cartridge hard disks are best kept within arm's reach) or even in another room if you have a remote power switch. This way you can get rid of the fan noise.

Software issues

Number of files. The Finder supplied with the original Macintosh had a limit of 128 files per disk drive, and often slowed down with more than about 80 files on a disk; it was designed for floppies rather than hard disks with hundreds of files. New Apple Finders (version 4.1 and later) can handle many more files.

Volumes and partitions. To get around file-number limits, first Corvus and then other manufacturers supplied software that let you partition their hard disk drives into logical "volumes," so that a single hard disk appeared to the computer as several different disk drives. You can mount or dismount volumes at will. Volumes work well and let you separate files so you don't get an overwhelming number in disk windows or mini-finders. For fastest operation, you should mount as few volumes at a time as possible and keep unnecessary floppy disks out of the computer; otherwise the operating system is constantly reconstructing disk information it doesn't actually need.

Most software for mounting and dismounting volumes operates as a separate program; a few operate as a desk accessory—a convenience, but you may have to be careful about dismounting a volume if a file is open. The software should prevent you from doing this or at least warn you. The best partitioning software allocates storage space dynamically; as you add more files, the selected volume expands up to the storage limit of the disk. On a network, however, dynamic partitioning is not a good idea; a careless user can take over a hard disk drive by using up all the space.

System software. Apple plans to release new system software that will support hard disks properly with a hierarchical file structure. Subdirectories for files will make partitioning hard disks unnecessary. Folders will become logical entities rather than merely cosmetic touches: An application in one folder will be able to open a document in another folder without the operating system needing to look at any other folders. Different files with the same name can reside in different folders, so you can maintain two different versions of an application or a document. Because the operating system will not constantly need to check the entire contents of every disk, many routine operations will gain speed.

Disk caches. A disk cache is a block of RAM set up as a buffer between the computer and the disk drive; see Chapter 9. All disk drives —floppy and hard—have at least a tiny disk cache, but some hard disks have their own large caches—64 KB or more (256 KB in the Micro-Design hard disk). For some types of disk access, these caches can improve operational speed considerably.

Speed

Transfer rate. The raw transfer rate is the maximum number of bits per second read off the disk surface and passed on to the computer. On the Macintosh, the floppies transfer data at a peak rate of 62.5 KB per second.

Hard disk transfer rates depend on how they are connected. The drives themselves transfer raw data at 625 KB per second (high-performance units at 1.25 MB per second), too fast for most microcomputers. With bus-connected drives, the disk-controller circuitry slows the data rate down to about 200 KB per second; port-connected hard disks are much slower.

Track-to-track access time. This is a second major speed factor: How long does the read/write head take to move from one track to another? In many situations, this factor is more important than raw transfer speed. The average access time runs from 30 to 100 milliseconds in hard disk drives; typical access times on a floppy drive are well over 100 milliseconds.

For a complete discussion of disk drive speed, see Chapter 29.

FINAL POINTS

The Macintosh handles floppy disks slowly. Its inefficient design requires the 68000 CPU to give the disk controller full attention when reading and writing on a disk; many other computers have a better design in which the controller by itself does most of the work, freeing the processor for other tasks. The new Macintosh ROM, which should appear early in 1986, will give a little extra speed with improved track-seeking logic on double-sided floppy drives.

Many advertisements for hard disk drives give unrealistic speeds, most often by quoting only the raw transfer rate. Even on the same drive, the time to read an apparently identical file can vary: After many files have been created and erased on a disk, the files tend to fragment—they wind up in pieces scattered around the disk, and require more head movements to read. Many published speed comparisons do not take this factor into account; fair speed comparisons are surprisingly hard to do.

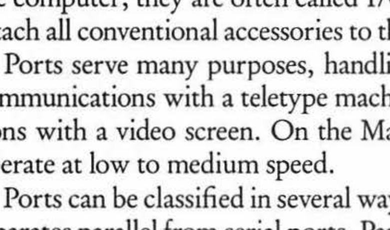
Input/Output Ports and Accessories

Chapter 15

Ports are electrical pathways that carry information in and out of the computer; they are often called I/O, or input/output, ports. You attach all conventional accessories to the Macintosh through its ports.

Ports serve many purposes, handling everything from low-speed communications with a teletype machine to high-speed communications with a video screen. On the Macintosh, all the available ports operate at low to medium speed.

Ports can be classified in several ways; one fundamental distinction separates parallel from serial ports. Parallel ports transmit and receive all eight bits in a byte simultaneously on eight separate wires, while serial ports send and receive the eight bits in sequence on a single wire. Parallel ports generally work faster than serial ports, but the speed difference doesn't matter for many uses. Ordinary printers, for example, work so much slower than a computer that a serial connection works fine. For electrical reasons, serial signals can be carried over much longer wires than parallel signals. Mac has serial ports only. Parallel ports most often drive printers; if you need a parallel port, several companies make separate serial-to-parallel converters.



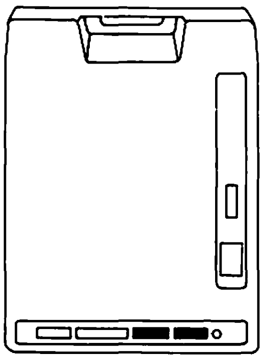
Serial links on left use a single wire; parallel links on right use eight wires. (Both links use an additional ground wire; for immunity to electric noise, the Macintosh serial port can use two signal wires plus a ground.)

The Macintosh has the following I/O ports:

- Two serial ports. In normal use, one is for a printer, the other for a modem.
- A disk-drive port.
- A mouse port.
- A keyboard port.
- An audio-output port.

For technical information, including wiring, see Chapter 29.

THE PRINTER AND MODEM PORTS



Serial ports

The Macintosh serial ports follow the RS-422 protocol, an enhanced form of the older, more common RS-232C protocol. A port's protocol is a set of rules specifying the timing, wiring, and voltage that a computer and its accessories must follow for communication.

The RS-422's improvements allow higher-speed communications and longer connecting wires between devices. RS-232C links are generally limited to 19,200 bits per second (2.4 KB per second) while Mac's RS-422 can run routinely at 230,400 bits per second (or 29 KB per second), and as fast as 920,000 bits per second with special techniques. RS-422 also has much better immunity to electrical noise.

The Macintosh serial ports will handle most RS-232C devices as well as RS-422 devices. Although RS-232C uses higher voltages than RS-422, the Macintosh serial ports will not be damaged. (Such compatibility with RS-232C voltages is not always true of RS-422 devices, however; use caution when intermixing devices.) The nine-pin serial-port connector does not follow the RS-449 connector standard; for wiring information, see Chapter 29.

In many cases, an RS-232C device can be connected with nothing more than a suitable cable; the Imagewriter, for example, is actually an RS-232C device. Other devices, such as elaborate modems, use more complex connection and control schemes than the Macintosh port can support; you will need to either reconfigure the device or buy a (possibly expensive) electronic interface.

Most ordinary RS-232C devices sold for microcomputers should work with the Mac, if supporting software is available and you can figure out how to set the hardware and software. But unless you have considerable skill and patience, don't try to attach an accessory device to your Macintosh that you have not already seen working properly on another Macintosh. In particular, try not to be the first on the block with some new accessory; it's a lot easier to wait for someone else to solve the set-up problems.

The RS-422 ports are bidirectional—information can travel both to and from the computer. You can connect two Macintoshes together directly with a cable through their serial ports. The Mac's modem and printer ports are similar, but not identical. When the computer is reading and writing on the disk, it checks the modem port for characters more frequently than it checks the printer port. As a result, a modem that communicates at 1200 bits per second (bps) or faster will not work reliably on the printer port but will work on the modem port; a slower 300-bps modem will work fine on either port. However, devices that can tell the computer when they are sending information have no problem running at high speed on the printer port, so you can connect 1200-bps and faster modems to the printer port if you add a memory buffer such as the Hayes Transet 1000.

The printer port also serves for communications via the AppleTalk network (see Chapter 26).

The serial-port configurations (settings) are stored in the non-volatile parameter RAM on the clock/calendar chip; this occasionally creates difficulty. See Chapter 18 for a note about modems, and Chapter 29 for general information.

Some software is sensitive to what's connected to the ports. A communications program might expect a modem to be connected; some programs won't start if certain hard disk drives are plugged in.

Accessories

Two serial ports aren't enough. Many accessories, ranging from television cameras to hard disks, need the ports. The only solution is an external switch, but you must be careful about when you use it. Obviously you shouldn't flip a switch while data is passing through it; switching at the wrong time with a hard disk drive connected could be disastrous.

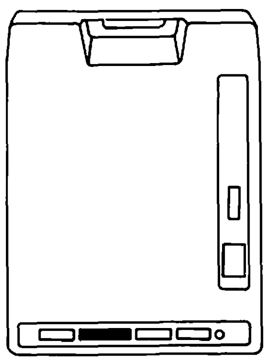
Many devices require an initalizing code to begin operation; if you switch after the code is sent, the device won't work properly (most software does not require the device to acknowledge the code). Another problem is switching while the power is on. Many companies warn you against plugging in their device while the power is turned on, in part because they don't know what the device may be connected to. In fact, many computer devices can be connected without damage via serial ports while powered on, but there will undoubtedly be exceptions; you have been warned. When in doubt, follow the manufacturer's recommendations.

Several companies make passive switching boxes, ranging from $35 to $100. Passive boxes contain only switches, no circuitry. Connectors and switches are among the least reliable computer components, so construction quality matters here. The boxes vary in some details; many switch all nine lines, but some will not switch one of the two ground wires (pins one and three in the serial and mouse ports).

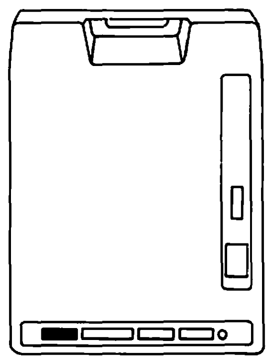
Microsoft's MacEnhancer is a software-selected switch box that you plug into one of Mac's serial ports; instead of a knob, you select the switch setting from a Desk Accessory program. MacEnhancer gives you four ways to connect. The first is an RS-422 port that replaces the port it plugs into. Two are IBM PC-compatible RS-232 ports (wired DTE) suitable for many devices, including most lower-speed devices that would otherwise connect directly to the Mac. The fourth connector is an IBM PC-compatible parallel port; internal circuitry performs the serial to parallel conversion.

THE DISK-DRIVE PORT

The disk-drive port is also bidirectional and operates at 62.5 KB per second. It's designed to support one additional microfloppy disk drive, but can also support a hard disk drive.



Disk-drive port



Mouse port

THE MOUSE PORT

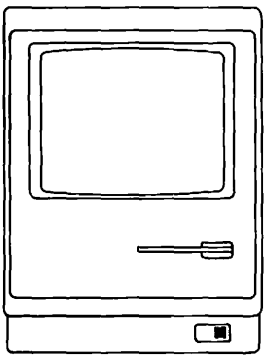
The mouse port is suitable only for attaching a mouse or other pointing device such as a trackball or joystick. Most devices using this port can be "daisy-chained," or wired in sequence so all devices are active at the same time.

THE KEYBOARD PORT

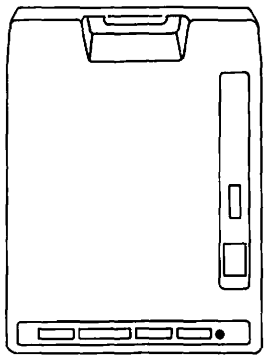
The keyboard port is a slow, general-purpose bidirectional serial port, able to support at least eight devices. The optional numeric keypad also plugs into the same port. Musical keyboards, data-entry pads that accept handwritten entries, and other devices that transmit data relatively slowly can be connected through this port.

TPS Electronics (Palo Alto, CA) and Computer Identics (Canton, MA) have bar-code readers that connect through the keyboard port; TPS also has readers for the magnetic stripe on credit cards. Special keyboards for the handicapped will also use the keyboard port.

Bar code printed by Imagewriter (TPS Electronics)



Keyboard port

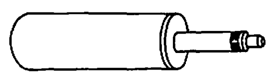


Audio-output port

THE AUDIO-OUTPUT PORT

The audio-output port sends audio information outward only. The port is normally connected to a small, 3-inch, built-in speaker.

The volume setting is stored in the parameter RAM. You can usually set the speaker volume by choosing Control Panel from the Apple menu. Set at minimum, the speaker should be inaudible. If you want to silence the speaker at all times, including the power-on tone, insert a dummy plug (Radio Shack #274-286) without any connections into the audio-output port.



The audio-output port plug (Radio Shack #274-286)

The audio-output port lets you connect the Macintosh sound output to a hi-fi system or tape recorder. The plug is a standard ⅛-inch phone plug. The signal is at speaker levels (maximum 1 volt, peak-to-peak) and should work satisfactorily through a high-level input on a hi-fi. Don't plug it into a magnetic phono cartridge input; you will overload most hi-fi amplifiers and get loud but distorted sound. If you want to record the sounds, connect the port to a tape recorder's high-level or auxiliary inputs rather than the microphone input.

Mac's sound generator can produce frequencies from below audibility to 11 kHz with four simultaneous voices. Although the generator uses digital techniques, it is in no way comparable in quality to digital audio recordings of music. Compact disc laser players, for example, use 16-bit digital-to-analog conversion, while the Macintosh generator uses 8-bit conversion. Nevertheless, digitized recordings of real sounds are possible (the first example was the Airborne game from Silicon Beach Software).

Speech Output

The Mac's sound generator is also flexible enough to synthesize speech. Speech output requires only speech software; no additional hardware is needed. The speech output uses the audio port and operates as an input/output device, just like a printer or modem.

On the Macintosh, the available speech programs operate in two parts. The first part translates ASCII text into phonemes, a phonetic representation of the words. The second part, the synthesizer, takes the phonemes and generates the codes that are turned into sound.

The sound generator can synthesize as fast as it speaks. Speed, pitch, and volume can be regulated independently. To aid pitch and inflection, the synthesizers work on a sentence at a time. Punctuation is handled with simple rules: at a comma, the voice will pause slightly with a rising inflection; at a question mark, a longer rise; at a period, a relatively long pause. Decimals are read as "point."

Since English has ill-defined pronunciation rules with many exceptions, more elaborate pre-processing may be needed to improve the speech quality. Such processing can define the inflection and timing of the speech. Users will be able to write programs that pre-process but will not be able to modify the synthesizer programs directly.

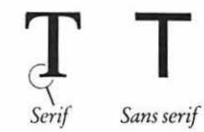
SmoothTalker (version 2.0, from First Byte) uses a proprietary synthesis technique and features a male and a female voice. You can see and edit the intermediate phoneme output, and call the synthesizer from within a programming language. Twelve hundred rules govern pronunciation; an exception dictionary stores words that do not follow the rules. SmoothTalker is much larger (30 KB + 26 KB for the two parts) than Apple's synthesizer but produces better quality speech.

Apple's speech software, written by Mark Barton and Joseph Katz, uses formant synthesis. Its compact code (7 KB + 15 KB in the pre-release version) also stores an exception dictionary. Apple has supplied the program to software developers; no plans have been announced for direct distribution to end users.

Printers Chapter 16

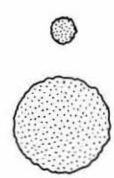
Printers come in many forms. After a brief look at the elements of print quality I will take up conventional designs and some considerations about their use, and then describe more sophisticated units such as the LaserWriter. For test information and a comparison of many printer models, see Table 16-3 at the end of this chapter.

PRINTER QUALITY



The most critical distinction among printers is print quality. Print quality depends on several factors:

- Resolution. This is given here in dots per inch (dpi) or the equivalent if the printer does not use dots. Table 16-1 gives a scale for rating printer resolution.
- Dot size and shape. For the best results, the dot size must be approximately the inverse of the resolution. Square dots are better than round for filling areas, round dots are better for curved edges. Oval dots are unsatisfactory. Most printers, particularly ink-jet and laser printers, produce ragged dots.
- Contrast. This is the difference between the printed image and the paper.
- Uniformity. Does the print quality vary over the page? Does the printer leave the clear areas completely untouched? Are the black areas evenly covered?
- Specular quality. The printed material should have a matte finish, free of distracting reflections.



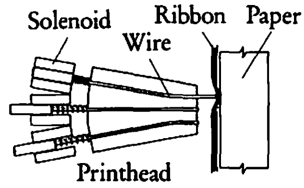
Relative dot sizes—LaserWriter (top) and Imagewriter (bottom).

TABLE 16-1. PRINTER-RESOLUTION RATING SCALE

Rating	*Printer types*	*Description*
Very low quality: under 125 dpi	Ordinary impact dot-matrix printer; Apple Imagewriter in standard mode.	Unacceptable for prolonged reading.
Low quality: 125–250 dpi	Most multiple-pass dot-matrix printers; Imagewriter in "high-resolution" mode; "near letter-quality" or "correspondence-quality" printers.	Unacceptable for prolonged reading. Cannot do an acceptable tapered serif; sans serif or square serif are more satisfactory.
Medium quality: 250–500 dpi	A few premium impact dot-matrix and thermal-transfer printers, but dots are too large to take advantage of this resolution; laser printers and related designs.	With carefully designed fonts, often acceptable for prolonged reading. Cannot do finely tapered serifs well.
High quality: 500–1000 dpi	Formed-character daisy-wheel printers achieve the equivalent of about 700 dpi, but lack precision positioning; lower-end typesetting machines.	Suitable for prolonged reading, with some reservations; daisy-wheel printers are marginal. Typesetting machines can reproduce some tapered serifs adequately.
Very high quality: over 1000 dpi	High-end typesetting equipment only; no microcomputer printer, now or in the near future.	The only choice for long-term reading. Can reproduce all typefaces gracefully, especially when over 2000 dpi.

DOT-MATRIX PRINTERS

The most common microcomputer printers form images by impact, driving an ink ribbon into the paper. Dot-matrix printers like the Imagewriter form characters out of little dots; the printhead contains a set of wires that dart in and out in a vertical array, pressing the inked ribbon against the paper as the head travels across it. The Imagewriter has nine printing wires; other designs have as many as 24. On all low-cost printers, the printed images look less crisp than the screen display; the printed page lacks the high contrast and uniformity of the screen, and the round dots leave gaps or create overlaps, unlike the square pixels on the screen. In the higher-resolution modes, the dot size is too large, smudging fine detail. Also, the printed image fades as the ribbon wears out.



In an impact dot-matrix printer, solenoids drive tiny print wires against ribbon and paper.

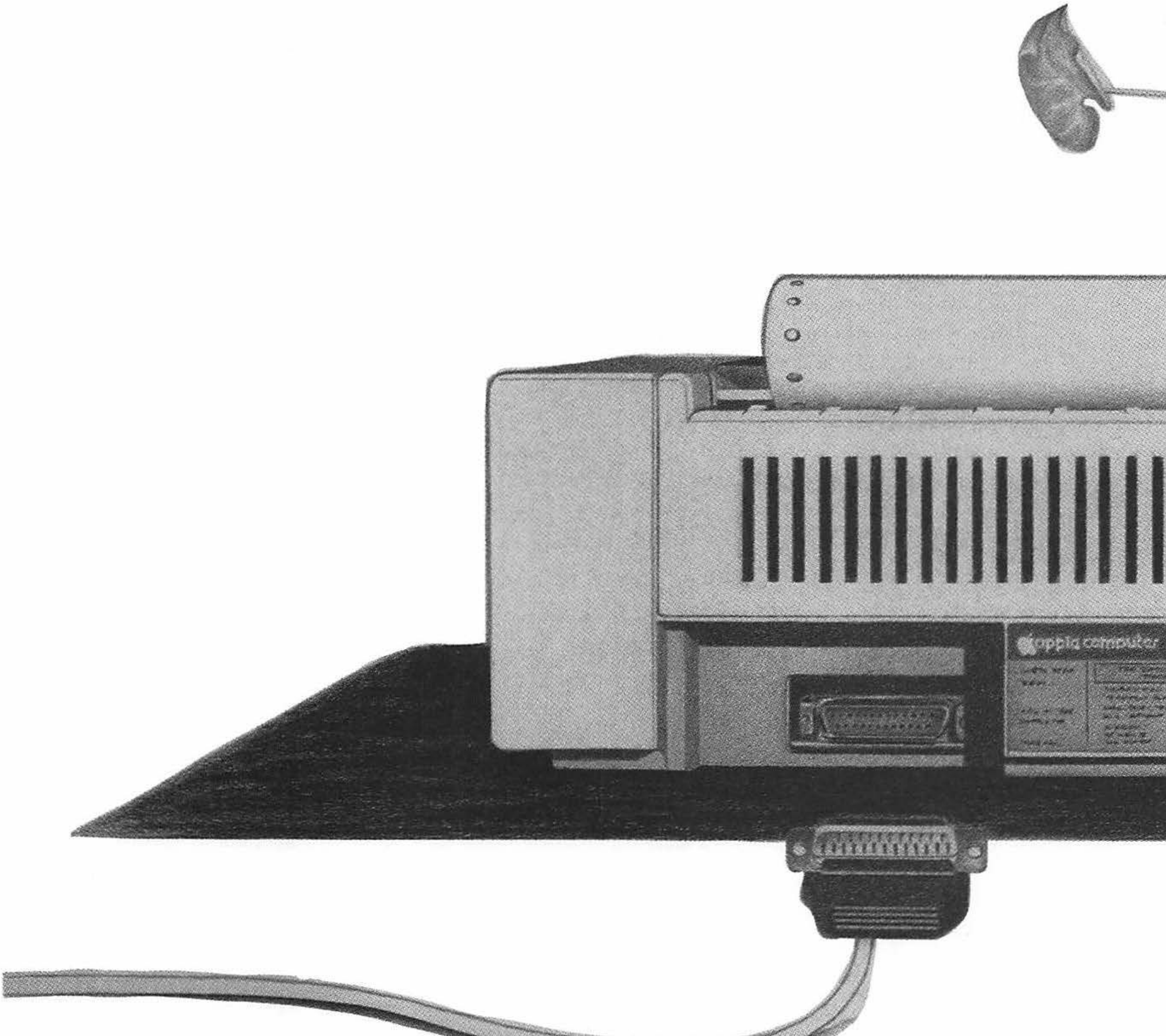
The Apple Imagewriter

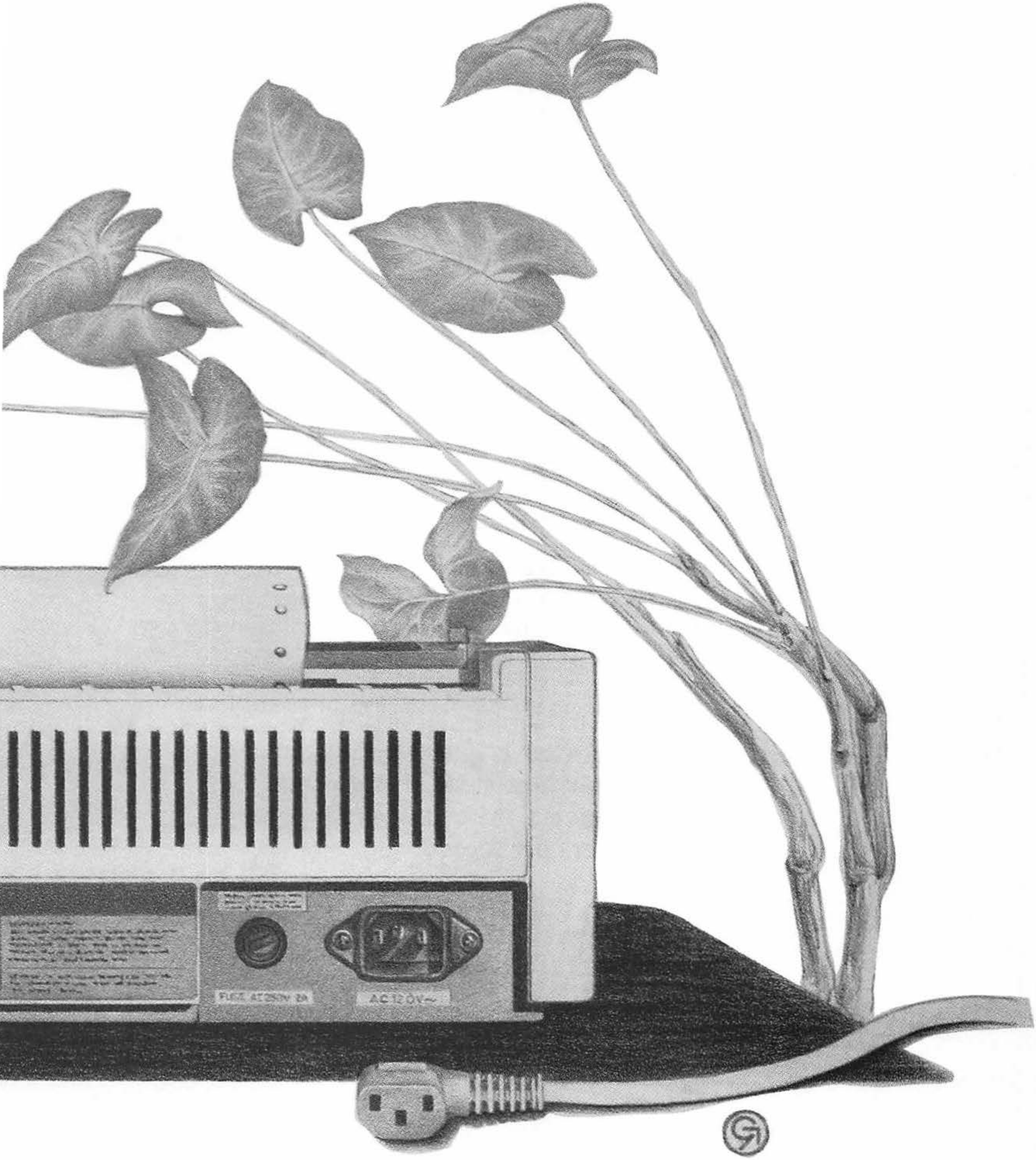
The standard printer for the Macintosh is Apple's Imagewriter, built by Tokyo Electric. It can print all of the Macintosh screen fonts and graphics, as well as screen dumps—replicas of the screen.

The Imagewriter offers the most straightforward printing you get with the Mac; all ordinary programs will drive it, and many accessories are designed to work specifically with it. The Imagewriter dot size is 16 mils (0.4 mm). The printer has three modes, called draft, standard, and high-resolution. The modes available depend on the application software, although many offer all three.

Draft mode is the fastest of the three, but also the lowest quality. Use this mode when you want quick results and don't care how the page looks. To get even word spacing, use a monospaced font, such as Monaco. Draft mode uses a font built into the printer, and only works for printing text and numbers, not graphics. Although Apple quotes a speed of 120 characters per second (cps), the true speed, including paper handling for a two-page memo, is about 50 cps.

The Imagewriter printer operating in draft mode. For a typical two-page memo, the true speed is about 50 characters per second.


apple computer


AC120V~

Standard mode prints at 72 dpi vertically and 80 dpi horizontally, similar to the Mac screen (75 dpi), and prints with pixel-for-pixel correspondence with the screen except for the vertical distortion. (You can often get rid of the distortion, however, as explained shortly.) Standard and high-resolution modes use proportional spacing, varying the letter width so that a capital M takes up more space than a capital I, a w more space than an l; numerals are all the same width so that columns of numbers will line up properly. The dots do not overlap in this mode, and so leave gaps. Nearly all software will print without complications in standard mode. When printing, the software uses the identical font information to generate both the screen display and the printed page. I consider standard mode to be very low quality. For a two-page memo, the true speed is 28 cps.

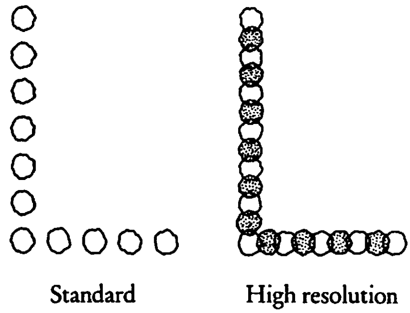
> The Imagewriter printer operating in **Standard** mode. For a typical two-page memo, the true speed is about 28 characters per second.

High-resolution mode, at 144 by 160 dpi, offers twice the resolution of standard mode. To achieve this, the printer prints twice as many dots horizontally and makes two passes over the paper for every line. On the second pass, the paper is rolled up the height of half a dot. In this mode, the dots overlap on the page by 50 percent. I consider this low quality. For a two-page memo, the true speed is 11 cps.

> **The Imagewriter printer operating in High-resolution mode. For a typical two-page memo, the true speed is about 11 characters per second.**

Using the Imagewriter

With conventional printers such as the Imagewriter, the Macintosh software creates a bit map internally and sends it to the printer. The bit map is created in bands, and the printer pauses while the software creates the next band. In high-resolution mode, a page contains 47 bands and 200 passes of the printhead.



High-resolution printing uses twice as many dots per inch as standard.

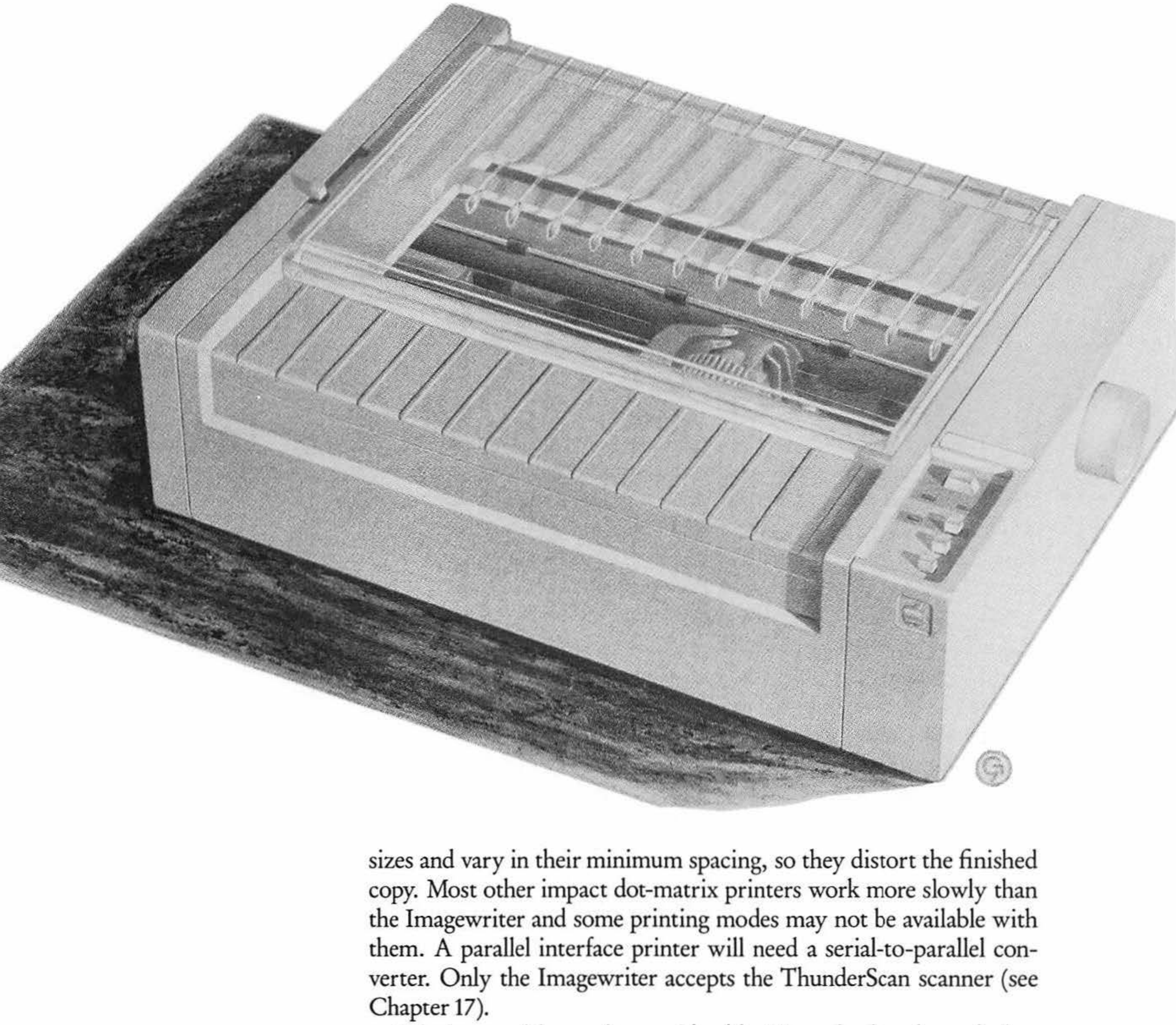
Because of limitations in the screen resolution, you cannot see on the screen exactly what high-resolution printing will look like because the software adds the additional dots when printing on paper. One way that software creates high-resolution printed text is to shrink a double-size character to the size required, using 24-point font data to create a 12-point printed character, for example.

The Imagewriter normally prints pictures elongated vertically about 11 percent compared to the screen; however, you can correct this by choosing the Tall Adjusted printing mode with many programs. But the Tall Adjusted mode forces a compromise in printhead positioning and the results can be a little untidy.

The Imagewriter suffers from clumsy paper handling. Form-feed paper (discussed later) is pushed rather than pulled, under the platen. In addition, the clear plastic paper guide gets in the way of single sheets and the first sheet of form-fed paper, especially heavier-weight paper. The result is compressed print or banding at the top of the page. Lift the paper guide to prevent this or to lift paper that has been curled around the platen cleanly out of the printer without jamming it in the single-sheet feed slot. After two or three inches of paper emerge, you can lower the cover. The Imagewriter driver software "burps" the paper at the top of each page in an effort to tighten things up, but this usually doesn't work.

Other Dot-Matrix Printers

There are many alternatives to the Imagewriter. You might want to use another printer because you already have one, or because of compatibility with other computers. There are some disadvantages: Macintosh software was specifically designed for the Imagewriter, so other printers generally do not work as well; they print different dot



sizes and vary in their minimum spacing, so they distort the finished copy. Most other impact dot-matrix printers work more slowly than the Imagewriter and some printing modes may not be available with them. A parallel interface printer will need a serial-to-parallel converter. Only the Imagewriter accepts the ThunderScan scanner (see Chapter 17).

Printing precision varies considerably. Not only cheaply made, but some relatively expensive high-speed printing mechanisms work less well than the middle-of-the-road printers. The most common problem is in printing vertical lines; any printer can put down a straight horizontal line, but some have trouble matching up segments of a vertical line on repeated passes of the printhead.

Some printers accept (or can be adapted to accept) Imagewriter codes directly; others require a special software driver. To use such a driver, you must use it as a replacement for the Imagewriter icon in the system folder of each disk you use for printing. A few programs bypass the Imagewriter driver altogether and thus might have trouble driving other printers. If you have a problem, neither the software company nor the printer company may be able to help out, since in most cases neither will be familiar with the other's product.

Drivers are available from a variety of sources. SoftStyle (Honolulu, HI) wrote the drivers supplied by several printer manufacturers and also sells them directly. The Microsoft MacEnhancer and Assimilation also have drivers.

Do not buy a printer for the Mac unless you have seen it working with a Mac. Compatibility claims can mislead; for example, many printers claim to emulate the Epson models but do not do so well enough to serve as a Macintosh printer with Epson driver software.

LETTER-QUALITY PRINTERS

The Imagewriter's "high-resolution" mode falls far short of the letter-quality standard of daisy-wheel printers and good electric typewriters. A true letter-quality printer uses a daisy wheel (or its close cousin, a thimble) embossed with molded, or formed-character letters. A print hammer strikes a petal, pressing it against the ribbon and paper and printing an entire character at once. In the most sophisticated daisy-wheel printers, the hammer strikes the letter M harder than it strikes a period, making the overall impressions even. As with typewriters, print sharpness and resolution depend on the precision of each molded character. Since daisy-wheel printers generally use carbon ribbons that are discarded after one use, ribbon condition matters less than it does for dot-matrix printers.

Daisy wheel print DAISY WHEEL PRINT

This is a sample of 10 pitch type.

ABCDEFGHIJKLMNOPQRSTUVWXYZ

But although daisy-wheel printers produce good-looking type, they have many disadvantages.

- They are slow, typically printing 12 to 40 characters per second (cps), although many are faster than the Imagewriter in high-resolution mode.
- They are usually noisier, though lacking the high-pitched whine, of a dot-matrix unit.

- They are limited to one typeface at a time. You can change typefaces by changing the daisy wheel but that slows printing considerably.
- Type size is limited by the size of the daisy wheel itself, so you can only get print sizes available on normal typewriters.
- Macintosh software may not display correct line lengths, making hyphenation impossible with a word processor. Microsoft Word, however, shows true line lengths for daisy-wheel printers.

Most important, daisy-wheel printers are only good for characters and they cannot effectively print graphics. Some daisy-wheel printers use the period to build up graphics images, in effect imitating a dot-matrix printer, but the process is slow, and the period petal on the daisy wheel wears out quickly.

Business users will probably need two printers, a dot-matrix model for graphics output and a daisy-wheel model for letter-quality printing. Many companies, including Assimilation and Creighton Development, have products such as the Microsoft MacEnhancer for connecting the Macintosh to common daisy-wheel printers. If you want to buy a daisy-wheel printer, look for a model compatible with the most popular instruction set, the Diablo 630.

In a pinch you can use an electronic typewriter with a suitable interface and driver software. But electronic typewriters are slow and nearly all of them require manual paper feed, a sheet at a time.

CONSIDERATIONS WITH CONVENTIONAL PRINTERS

The following list of considerations applies primarily to conventional rather than laser printers.

- Paper handling and switching. Many printers make it difficult to change between form-feed and single-sheet paper or envelopes.
- Speed. Manufacturers usually quote the most optimistic measurement, the highest burst speed when printing a single line. When the times to advance the paper by a line or to the next page are included, overall printing speed is much lower. Many low-cost printers lack a true form feed (they can only advance a line at a time; better printers can move an entire page quickly).

- If you often need to print on single sheets, you might check into sheet feeders, an intermittently reliable way to feed paper automatically. Even the best sheet feeders suffer from variations in paper surfaces and dirty rubber rollers. Feeders are available for many daisy-wheel printers but only for a few dot-matrix units. The best sheet feeders are electrically released, driven by the form-feed command sent to the printer. The cheaper, mechanically released feeders are driven by gears connected to the printer platen. They generally require a page-length setting of 15 inches for standard 11-inch-long paper, using the extra four inches to roll the new sheet in. Thus far, few Macintosh programs let you set the page length.
- Wide-carriage printers. Many printers, including the Imagewriter, are available with both standard (8 inches, or 20 centimeters) and wide (14 to 15 inches, or 35 to 38 centimeters) carriages. Printers with wider carriages will let you create larger images, although some software may not take advantage of such width. MacWrite, for example, does not let you create documents wider than the screen although other programs do.

Ongoing Costs

Total printer costs include not only the purchase price but the cost of keeping the printer fed with ribbons and paper. Although paper costs the same for any printer, ribbon costs vary widely.

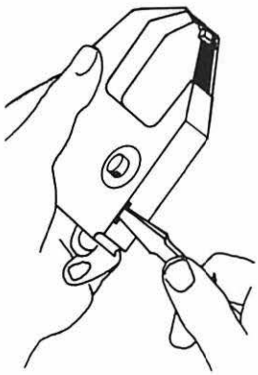
Printer ribbons

The carbon ribbons for daisy-wheel printers and the printing cartridge for the LaserWriter are the most expensive, about 3 cents per page. When considering a printer, find out if it uses standard ribbons available from many sources, or if the printer manufacturer treats you as a captive audience for overpriced ribbons.

Dot-matrix printers generally use cloth (nylon) ribbons. Unlike the fixed-length carbon ribbons, cloth ribbons form an endless loop inside the case. Ribbon longevity thus depends on your tolerance for slowly fading ink. Many users keep an older ribbon for routine printing and a nearly new ribbon for important jobs. The Imagewriter ribbon is the same one used on the earlier Apple dot-matrix printer and some others, including the NEC 8023, C. Itoh 8510, and DEC LA50. If you need the cleanest possible output from the Imagewriter, Aspen Ribbons and Boston Software Publishers supply carbon ribbons.

Depending on printer/ribbon design, you have several choices for replacing cloth ribbons:

- Buy a new one.
- Replace only the ribbon if the case can be opened and you don't mind messy fingers (in some printers, you can simply turn the ribbon over and print on another portion of it).
- Re-ink the ribbon using a kit advertised in computer hobby magazines. You should probably only re-ink a few times; a frayed ribbon can damage print heads.

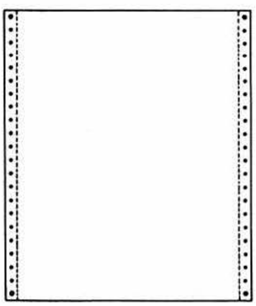


Open the ribbon case with a small screwdriver.

If you are truly desperate, one simple way to rejuvenate a fading cloth ribbon is to open the case and spray a little WD-40 or similar lubricant on the ribbon. Let the ribbon sit for a few minutes after spraying. The lubricant acts as a solvent, spreading more of the ink over the ribbon's surface.

One last hint: Don't stockpile ribbons; because the ink dries out, just keep enough for a few months rather than a few years.

Paper



Form-feed paper

Form-feed computer paper has extra margins with holes that fit into a tractor feed; it comes in many styles and weights. For normal use, buy 20-pound, 9½- by 11-inch paper; tearing off the perforations will leave 8½- by 11-inch paper. In Europe and Japan you can set Mac's software for the slightly narrower and longer A4 size. Many programs also offer a wide printing format that prints sideways, with the top line running down the right side of the paper (the paper is loaded normally). Paper types and sizes are given in Table 16-2.

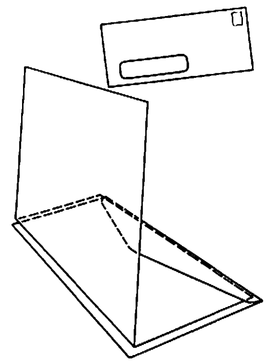
TABLE 16-2. PAPER-SIZE TABLE

B4 (international)	10.1 by 14.3 in	(257 by 364 mm)
Legal	8.5 by 14 in	(216 by 256 mm)
Foolscap	8.5 by 13 in	(216 by 330 mm)
Folio	8.3 by 13 in	(210 by 330 mm)
A4	8.3 by 11.7 in	(210 by 297 mm)
Letter	8.5 by 11 in	(216 by 279 mm)
B5	7.2 by 10.1 in	(182 by 257 mm)
Ledger	11.0 by 17 in	(279 by 432 mm)
International fanfold	8.3 by 12 in	(210 by 304 mm)

Removing the perforated margins of ordinary form-feed paper leaves the edges slightly ragged, but you can buy more expensive paper with much smaller perforations and much cleaner edges, suitable even for letterhead.

Mailing labels also come on form-feed stock. The labels are normally sized for six lines per inch; both MacWrite and Microsoft Word can be set to print at this spacing. Unfortunately many database programs can't handle six lines per inch printing. You could move the text to a word processor for printing, but the process is inconvenient.

Envelopes are a nuisance since you must alternate them with ordinary sheets, and so must attend to the printer frequently. You can copy addresses to a separate file for printing in a batch, but this is also clumsy. Window envelopes are the best solution, since you only have to print the address once, in the letter itself.



To fold a letter for a window envelope, use the envelope flap as a guide.

For copies of your printed pages, you have several choices:

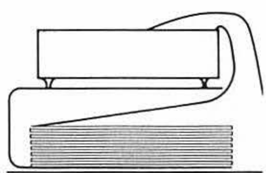
- Make a photocopy. This saves the trouble of changing papers or inserting carbons.
- Use carbon paper with single sheets. Impact printers can usually handle at least one carbon copy. If ordinary carbon paper does not make dark enough copies, try Carter's Super Midnight copy film.
- Use multipart forms paper. This automatically includes a carbon copy or equivalent.
- Print a second copy. This is fairly quick with laser printers, tedious with conventional printers. You might select the draft printing mode on the Imagewriter to save time.

Printer Buffers

Computers can feed information to conventional printers much faster than the printers can print. As a result, you will spend considerable time waiting for the printer to finish a job. A printer buffer takes the print instructions from the computer at high speed and stores it in RAM, feeding the information slowly to the printer as required. The computer "thinks" that the printer is done once everything goes to the buffer, and it returns to other work. Many buffer products are available; some are freestanding hardware accessories, others are built into hard disk drives, and still others use the Mac's main RAM.

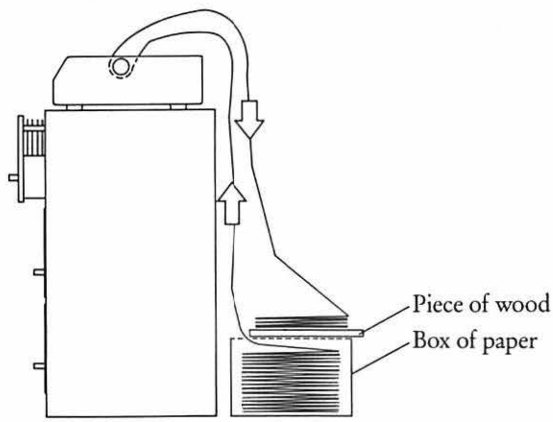
However they work, printer buffers are handy, but they must have certain features. For example, you must be able to erase their contents. Often you will start printing and realize that you have made a formatting error. If you cannot erase the buffer, the only way to avoid printing the entire file is by turning off the computer and printer. (Typing Command-period does not affect the buffer.) The fanciest and most expensive buffers are the freestanding variety; some include an erase button as well as a reprint button, so if you want another copy, the buffer resends the information to the printer.

Printer Stands



A printer stand makes it easy to use a printer on a table.

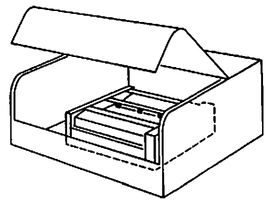
A printer stand can simplify operations by keeping form-feed paper from becoming tangled on the floor. If you have the space, build or buy a stand with space below the printer for a large box of paper. If space is limited, simple stands will fit on a tabletop. If you feed paper from a box and cannot put the box underneath the printer (because the printer is on a file cabinet, for example), put a small board across the paper box to receive the output. In all cases, check that paper leaving the printer doesn't interfere with paper feeding in.



Printer Sound Hoods

All impact printers—dot-matrix and daisy-wheel—make too much noise for anyone to work nearby; you can't carry on a telephone conversation in the same room. A sound hood—a large, bulky box with a sound-absorbing lining—can make life with a printer much more peaceful. Prices run from $100 to $500.

If you use a hood, make sure that the printer has adequate ventilation to dissipate heat, and check paper handling with the hood on; many hoods make getting at the paper difficult. The printer can also be banished to another room or a closet. Putting sound-absorbent material such as a cloth wall hanging or curtain next to the printer can also help; the best but most expensive material is the acoustic foam used in sound recording studios.



Sound hood

Laser printers and some other designs are relatively quiet except for a fan and minor paper-handling noises.

THE LASERWRITER

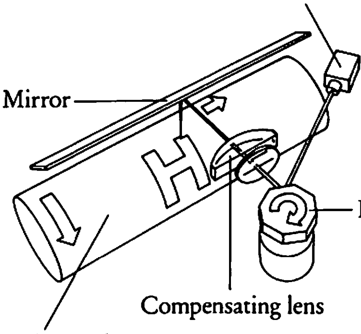
Laser printers use xerographic technology, and so work in a fundamentally different way from impact printers. Rather than stamping or inking images onto paper, they use electrostatic charges, revolving drums and cylinders, powdery black toner, heat, and pressure to create a photocopy-like image. The amazing range of processes built into laser printers makes them the most complex technology in offices today. Laser printing involves many complicated steps, which are described in the following section. If you are not interested, or have no need to understand the details, you can skip the section or come back to it at another time.

How It Works

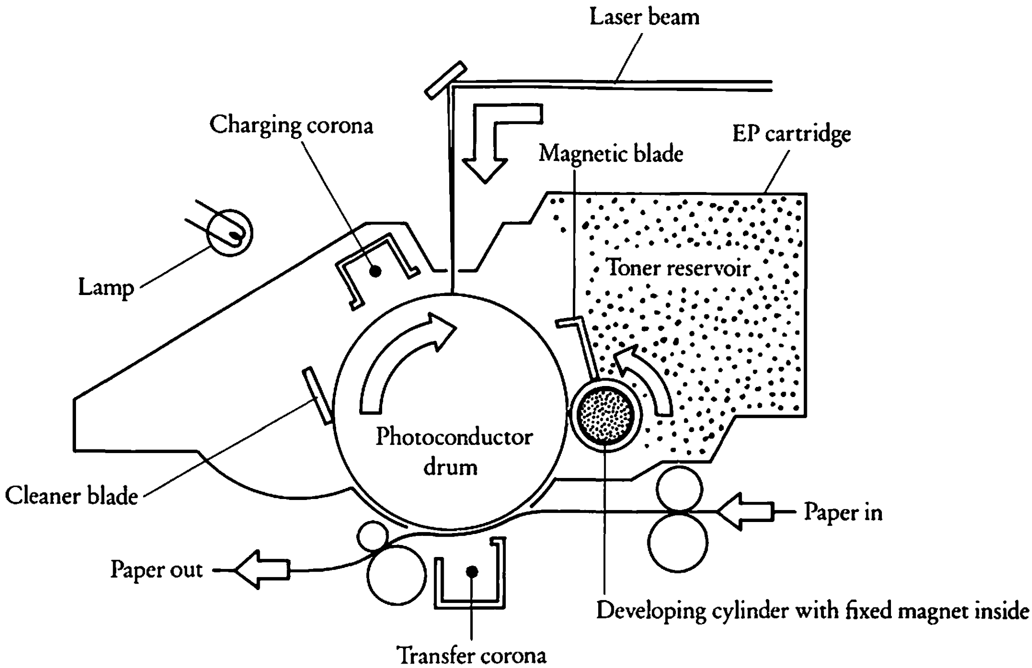
- To print an image, the Macintosh application software first sends information to a utility program, the laser printer driver, in the Mac. The printer driver converts the information to a special computer language, PostScript. Some software creates PostScript commands directly.
- The printer driver sends the PostScript commands via an AppleTalk network to the printer.
- The PostScript processor, an independent computer inside the LaserWriter, interprets the PostScript commands and converts the printing information into a dot-matrix image with 300 dots per inch. The image is laid out in a sequence of raster lines much like a television picture.
- The raster information modulates a laser beam.

Meanwhile the main printing process plays out on a rotating photoconductor drum inside the printer:

- A preconditioning lamp illuminates the drum surface, neutralizing any leftover charges from printing the previous page.
- The charging corona, a fine wire held at a high voltage, ionizes and lays down air molecules as an even layer of negative electric charges on the photoconductor surface.
- A rapidly spinning mirror deflects the modulated laser beam across the photoconductor. A compensator lens corrects for the scanning geometry.



- Wherever the beam strikes, the photoconductor, as its name implies, conducts electricity so that the negative charges drain off what will be the image's black areas, leaving these areas at a positive potential relative to the rest of the image. This creates an electrostatic latent image, much like the latent image on photographic film after exposure.
- The toner consists of magnetic particles covered with meltable plastic resin and carbon black (in jargon, a monocomponent toner). An organic polymer coating on the toner particle creates negative charges by friction (triboelectricity) against other toner particles. The toner is fed into a small gap between a magnetic blade and the developing cylinder.



- The rotating developing cylinder contains a fixed magnet inside; the concentrated magnetic field creates a toner "curtain" between the blade and magnet. The developing cylinder picks up an even coating of toner from the curtain as it turns. The toner sticks to the cylinder because of the fixed magnet.
- The developing cylinder and the photoconductor drum rotate together very closely, but not quite touching. A rapidly alternating electrical field between the cylinder and drum creates a tiny cloud of toner along the gap. Areas on the drum that are negatively charged repel the toner, pushing it back to the cylinder, where it is held by the magnet. Discharged areas are positive with respect to the toner and attract it, so the toner jumps the gap and sticks to the photoconductor, developing the latent image.
- A transfer corona charges the paper positively, so when the paper rolls against the photoconductor drum, the negatively charged toner sticks to the paper.
- The toned image is fused to the paper by a combination of heat and pressure.
- A cleaner blade wipes the excess toner off the drum so the process can begin again.

The Page-Image Buffer

The LaserWriter can print an 87-square-inch area, distributed as follows:

Page type	*Paper size*	*Image size*	*Pixels*	*Buffer size*
Letter	8.5 by 11 in	8 by 10.92 in	2400 by 3276	982,800 bytes
Note	8.5 by 11	7.68 by 10.16	2304 by 3048	877,824
Legal	8.5 by 14	6.72 by 13	2016 by 3900	982,800

The printable area is always centered on the page.

For image stability and geometric accuracy, a laser printer must print an entire page once it starts. Unlike ordinary printers, it cannot pause in mid-page to receive more information; thus all the information must be available when it starts. At 300 dots per inch, there are nearly 8,000,000 dots on the page, so a printer capable of full graphics must have 8,000,000 bits or about a megabyte of memory as a page-image buffer. No standard microcomputer communications link can move the information fast enough for the computer to drive the laser beam directly; the printer must have its own processor and memory. The Apple LaserWriter printer has a 12 MHz 68000 CPU with 1.5 MB of RAM and 512 KB of ROM, which is considerably more power than the Macintosh itself.

PostScript

PostScript, a graphics language developed by Adobe Systems, provides the LaserWriter with several forms of information:

- Font information. This includes style, size, and printing angle. Size and angle are continuously variable.
- Character information. Once the font is specified, the printer driver needs only send simple character codes until the font changes.
- Graphics primitives. To draw a line, the PostScript code sends information about the line's length, width, and orientation. In the LaserWriter, the PostScript processor figures out the actual pixels; this saves considerable time for structured graphics that can be described in elemental shapes. Since MacDraw creates images from such shapes, it generates cleaner results on a laser printer than MacPaint.
- Bit-map information, or the pixel-by-pixel image. Imagewriter-style printing can be done this way; each pixel is treated as a block of 16 LaserWriter dots at an effective

75 dpi resolution. Bit maps at the full 300 dpi are also possible, but this should be done as a last resort because it takes so long to send information with so much detail.

- Gray-scale information. Macintosh software can send gray-scale information along with the bit map, and the PostScript processor will create a half-tone pattern to simulate the grays. On the LaserWriter, this works effectively at the equivalent of a 60-lines-per-inch screen.

Four basic fonts and several variants are built into the LaserWriter ROM:

- Times Roman, Times Italic, Times Bold, Times Bold and Italic.
- Courier, Courier Oblique, Courier Bold, Courier Bold and Oblique.
- Helvetica, Helvetica Oblique, Helvetica Bold, Helvetica Bold and Oblique.
- Symbol.

Times is similar to the New York screen font, Helvetica to the Geneva screen font. Bolds and italics are designed for the purpose rather than generated by simple algorithms as in the screen fonts.

Font storage and usage

PostScript's storage and usage of fonts is complex. As with the description of laser printing, you can skip this section if you have no need for or interest in the information.

- The basic font information consists of outlines rather than a bit map. One set of outlines is used to generate all sizes from 4 point to as large as the paper (on the Macintosh, the practical limit is the largest screen size, 127 point). Beside the font outlines stored in ROM, new font outlines (typically requiring 15 to 20 KB per font) can be stored on disk and transmitted to the printer as needed.
- The PostScript processor divides up its 1.5-MB memory as follows: 878 KB for the Note-size page buffer, 150 KB for the character cache, 345 KB for "virtual memory" or a work area, and the remainder for PostScript's internal processing. The page-buffer size is under software control; by using Note size as the standard format, Apple gives up some printable area around the margins to conserve memory. The total memory cannot be expanded.

- Three fonts have bit maps stored in 36 KB of ROM so they are always available: A full ASCII character set in Courier 10 point, and alphanumerics and common punctuation in Times Roman and Helvetica 12 point. When you turn the power on, the LaserWriter builds up several more fonts by reading the outlines in ROM, creating a bit map for each character, and moving the bit map to the character cache. These fonts are Times Roman and Helvetica in 10 and 12 point, plus lowercase letters for Times Bold and Helvetica Bold in 10, 12, and 14 point. The bit map is trimmed within a rectangular block to save memory. A complete bit map for a typical 10-point font takes about 10 KB of memory; the storage required goes up with the square of the point size.
- When you print a character, the PostScript processor checks the character cache; if the desired character is there, the processor expands it to a full bit map and places the result in the page buffer.
- If you ask for fonts and sizes not in the character cache, the processor creates the bit maps and puts them in the cache. This is done character by character, as needed, rather than an entire alphabet at one time. If the character cache fills up, the processor discards the least recently used character(s) to create space. The cache can store about 700 characters (in the 12-point size); a typical page uses about 200 distinct characters. If the printing is rotated from the usual horizontal orientation, each character at each specific angle has its own bit map.
- Large characters with a bit map greater than 1200 bytes are not put into the character cache. Instead they are created and immediately moved to the page buffer. This conserves the cache's memory without requiring too much repeated work, since such characters are usually not used often. A 24-point size marks the approximate dividing line; at this size, some characters exceed 1200 bytes.
- The process of building up a font takes about ⅓ second per 12-point character. Thus the first page using a new font will be slow to print, but a second page will come out more quickly. Moving a character's bit map from the cache to the page buffer is quick: about 500 microseconds.
- Each laser font comes with a matching 75-dpi font for the screen. The great challenge in screen-font design is getting the laser font and the screen font to come out with the same

line length. A screen font's width can only be adjusted in increments of 1/75 inch while the laser character can vary by much smaller increments. The matching screen fonts are designed so that some characters are narrower than the printed width and others are wider, using a statistical distribution so the overall line length comes out close to the laser-printer line length. For left-justified text, PostScript simply takes as much space as it needs to print the characters in a line. For full-justified text, the Print Manager program distributes the characters in the line to fit the line width on the screen. In rare cases the mismatch in screen and printed widths will be off enough to create a problem. (A new version of QuickDraw permitting objects with fractional pixel sizes will cure this problem.)

- In small sizes the matching screen fonts may not look as good as the normal screen fonts because the highest priority is not appearance but to get the widths set correctly. This visual mismatch is particularly a problem for 12-point and smaller fonts. You can use the standard New York and Geneva fonts on-screen, and swap the screen fonts for LaserWriter fonts just before printing, but the character spacing will be unsatisfactory. You can also print the screen fonts designed for the Imagewriter. With straight printing, the LaserWriter rendition can look a little better because the dots are square; with smoothing, some fonts improve, while some intricate fonts do not fare so well; the smoothing only does 45-degree and 90-degree straight-line interpolation. The smoothing works differently when printing black characters on white than when printing white on black.

For more information about fonts, see Chapter 17.

Print Quality and Speed

The LaserWriter prints 300 dots per inch, or medium resolution. The laser writes dots of a fixed size (4 mils or 0.1 mm) on the photoconductor drum, but the printed pixel size varies. The paper surface quality and the toning and fusing steps all distort the dots into irregular shapes on the finished page.

A laser printer's tremendous flexibility with graphics and typefaces offsets its quality problems. The printed image isn't really letter quality; the slightly softer image lacks the crispness of formed-character impact printers. Because electrostatic images are unstable (the charged toner particles repel each other on the photoconductor drum) black areas are printed unevenly. Like all xerographic printers, the LaserWriter makes originals that look like photocopies; edges are a little rough and the printing process leaves bits of stray toner on the paper.

The LaserWriter's speed varies considerably with the nature of the printing. For ordinary text, the printer runs at about two to three pages a minute. Pages with complex graphics take more time than simple text, where all the necessary bit maps are already in the character cache. Once the page image is in memory, printing takes 12 seconds a page. If you need several copies of the same page, the LaserWriter runs at its full speed, eight pages per minute.

LaserWriter Notes

The LaserWriter is based on the Canon LBP-CX printing engine, the engine used by all the first truly low-cost laser printers. The LBP-CX was designed for light-duty printing, rated for 3000 pages a month and a total lifetime of 100,000 pages, at a final cost of nearly ten cents per page. However, the engine is fairly robust and some units have already far exceeded these figures.

The paper cassette tray can handle 16- to 21-lb (64 to 80 g/m^2) paper; the manual sheet feeder 16- to 34-lb (60 to 128 g/m^2). Paper designed for photocopying works well, better than paper with a high cotton rag content. For offset masters, use 70-lb coated stock, such as Mead Offset Enamel or Shorewood Gloss, and start with the printing density control turned all the way up (graphic arts stock uses a different paper weight scale).

The paper-feed tray on the LaserWriter and other LBP-CX-based printers holds just one hundred sheets, and when you print multiple-page documents, the stack comes out with the first page face-up on the bottom. The output tray holds only 20 pages. Especially when the printer is running in a local area network, someone must check the paper trays regularly. However, you can remove the output tray and let the paper drop 10 inches. If the paper placed in the feed trays is oriented so the slight curl bulges downward, the additional curl caused during fusing will cause the paper to flip over on output, restoring normal orientation. (This isn't necessary for MacWrite, which prints the last page first.)

Ziyad makes accessory paper-handling equipment for some Canon LBP-CX units—a dual 175-sheet paper-feed tray plus an envelope feeder. The Ziyad paper feeder needs special codes embedded in the text to select the paper-feed trays and thus requires some changes in Macintosh software.

A small plastic belt on the LBP-CX prevents printing within 0.35 inch along the right edge; the belt carries the paper through the mechanism. The LBP-CX can print to within 0.2 inch of the left edge, 0.04 inch of the top, and to the bottom edge. Apple's Print Manager software uses a 0.42-inch border to square up the margins and reduce the page-buffer memory. Special software can print to the LBP-CX limits, however.

Some Canon-based printers are much cheaper than the LaserWriter because they utilize far less sophisticated electronics. The Hewlett-Packard LaserJet printer, for example, has just 59 kilobytes of RAM and can print only a tiny five square inches of full-resolution graphics on a page. Text is printed from fixed-size bit maps in ROM without resorting to a RAM memory buffer; the LaserJet is essentially a daisy-wheel printer using laser technology.

The LBP-CX printing engine is similar to the Canon PC series of small photocopiers. They both use a disposable cartridge that contains critical components, such as the photoconductor drum and toner. When the toner runs out, or if the photoconductor is seriously scratched, you simply replace the cartridge, eliminating the cause for most service calls. The EP cartridges for the LBP-CX, however, are not mechanically or functionally interchangeable with the PC cartridges for the copiers; the copiers use positive imaging but the printer uses negative imaging (the laser writes the black areas of the finished page). Also, the LBP-CX incorporates a safety interlock, activated by the EP cartridge, for the laser beam. The EP cartridges can print on acetate for overhead transparencies. They come in black, brown, and blue, and contain enough toner for 3000 pages of average coverage; pages with large printed areas will reduce the page count.

The nearly identical paper cassette trays on the LBP-CX and PC copiers are not compatible.

The LaserWriter will work with other computers, either via the AppleTalk network or directly connected to them via an RS-232C port; through that port, the LaserWriter can operate either in PostScript mode or as a text printer, emulating the most common Diablo 630 commands.

OTHER LASER PRINTERS

Other laser printers will work with a Macintosh; some of them are designed for a much higher printing volume. The Quality Micro Systems Lasergraphix 1200 ($20,000) is based on the Xerox 2700 printing engine; it runs at a maximum speed of 12 pages per minute, and is rated for 15,000 pages a month. Dataproducts plans a unit based on a Toshiba printing engine rated at 30,000 pages a month. The key requirements for such printers are a PostScript processor and an AppleTalk interface. The output of fast printers will be limited by the time required to build up a page image and the speed of AppleTalk.

Some laser printers (including the Xerox) are positive printers; the drum starts out black and the laser writes on the areas that will be white on the printed page. (Some printer manufacturers define positive and negative with respect to the photoconductor, reversing the usage here.) Most users won't care whether a printer is positive or negative, but the print quality is subtly changed. With negative imaging, a character is built up from black dots on paper; with positive imaging, they are "chiseled away" with white dots. Because of this difference, a one-pixel-wide line is practical on a negative printer; positive printers, however, usually have a two-pixel minimum width because single-pixel lines tend to be obliterated. Adobe changes processing details within PostScript for each printer type.

SHARING A PRINTER

If you need to connect several Macintosh computers to a single conventional printer, you can install a switch box between the printer port of each Macintosh and the printer. The LaserWriter only works with a Macintosh through the AppleTalk local area network, which is designed to permit sharing (see Chapter 25).

PLOTTERS

Instead of printing dots, plotters draw lines by moving a pen directly on paper. Ballpoint, felt-tip, and liquid-ink pens produce a more even, continuous line than most dot-matrix printers, and many plotters offer a choice of pen colors. With suitable inks, a pen can write on transparent sheets for overhead projectors.

But plotters generate characters slowly and can't shade or fill in regions as well as dot-matrix printers can. As dot-matrix resolution has improved, plotters have lost ground, more and more becoming tools for specialists. Plotters work better on structured images such as graphs than on pixel-based images such as MacPaint drawings. Software for driving plotters is available from Microspot (Maidstone, Kent, England) and Mesa Graphics (Los Alamos, NM).

TABLE 16-3. PRINTER INFORMATION

Printer	*Relative image size*	*Dot size (mils)*	*Draft speed (cps)*	*Bit-mapped printing speed (cps)*	*Print quality*
Apple Imagewriter	1 × 1	16	50	28	average
Apple LaserWriter[1]	1.03 × 1	4/14	100	6	above average
Brother TwinWriter 5[2]	1 × 1	13	27	16	slightly below average
C. Itoh ProWriter	1 × 1	16			average
Citizen MSP-10	1.12 × 0.93	18 × 14			below average
Epson FX-80	1 × 0.97*	15	31	13	average
Epson JX	1 × 0.97*	14	41	16†	below average
Epson LX-80	1 × 0.97*	14	31	18†	average
Epson LQ-1500[3]	1 × 1*	12 × 15	31	16†	far below average
Hewlett-Packard ThinkJet[4]	0.90 × 0.75	9	69	17	below average
Hewlett-Packard LaserJet[5]	1.06 × 0.96	4/14	33	22	above average
Mannesmann Tally 85	1 × 1	13		25	average
Okidata 192	1 × 1	18		20	slightly below average
Riteman Blue Mac	1.11 × 1*	24 × 15			far below average
Texas Instruments 850XL	1 × 1.08*	15	25	10†	average
Texas Instruments 855XL	1 × 1.08*		56	24†	average
Toshiba P1340	0.9 × 0.94	16	27	6	below average
Allied Linotype 101,300		<2			Typesetters; far above average

Notes:

Full test information was not available for all models. Measurements are based on driver software supplied or recommended by the printer manufacturer (speed figures can depend on the driver software). Relative image size is measured with respect to the Imagewriter for a MacWrite standard-mode page (values marked with * are measured for a screen dump), horizontal axis first.

Dot sizes are measured on paper; oval dots are given horizontal axis first. Speed measurements: Draft printing speed uses fonts built into the printer, so the output does not correspond to the Macintosh screen except for the LaserWriter. Bit-mapped speeds are for a MacWrite document printed in standard mode, including paper handling (figures marked † are calculated from graphics printing). Print-quality comments are based on standard-mode printing only, using Macintosh screen fonts.

Footnotes:

1 Apple LaserWriter uses 4 mil dots on a 3.33 mil pitch. When duplicating screen graphics, each screen pixel is rendered as a 14-mil square on paper; printing is very slow. Using PostScript fonts, print quality is much higher than average and still maintains screen correspondence; speed is the fastest of any printer described here, 100 cps.

2 Draft speed is for daisy-wheel printhead. Brother TwinWriter 5 combines dot-matrix and daisy-wheel printheads in a single chassis; with suitable software, both printheads could operate on a single page.

3 Driver for Epson LQ-1500 interpolates pixels when printing in graphics mode, producing pixel-doubling artifacts.

4 Hewlett-Packard ThinkJet requires short-fiber paper to prevent ink smear, but even with such paper the dots are exceptionally ragged.

5 Driver for Hewlett-Packard LaserJet slows down printing by computing positioning information in draft mode. The LaserJet has the same dot structure as the Apple LaserWriter; see text on laser printers.

Chapter 17
Graphic Arts

On early microcomputers, graphics was a one-way process; a program might create the graphics on the screen and printer, but you could not move them to another program or mix the output of different programs. The Macintosh is the first microcomputer to have a standard graphics format that allows pictures created with one program to be moved to another. You can process graphics nearly as easily as words, but do it with care; don't lend credence to the joke that one million Macintoshes sold will mean ten million bad pictures.

This chapter covers several graphics topics in turn:

- Scanning existing images and printing pictures.
- Using clip-art libraries.
- Choosing fonts.
- Designing with page-makeup programs.

This chapter includes some fairly technical information. The glossary defines essential terms, but you may want to skim the chapter first and go back to specific sections again later. If you need to reproduce the screen image, see Chapter 28.

SCANNING IMAGES

The power of MacPaint and other graphics programs has almost obscured the essential fact that creating attractive images takes both skill and an aesthetic sense. Artists can create satisfying pictures on the Mac quickly, but the rest of us need help.

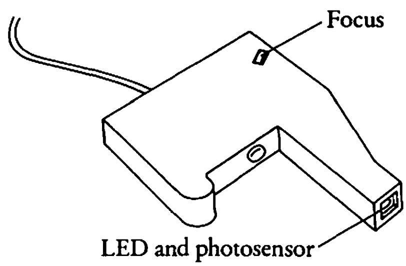
The majority of images we want to use already exist, either on paper or as an object we need to "photograph" with a Mac. To convert one of these images into a Macintosh file, you can scan it or draw it. Most of the time, scanners are the best choice; they are much easier and quicker, and generally yield better results. The best scanners are modestly priced and soon pay back their cost in time saved.

Scanners and Digitizers

Scanners and digitizers convert artwork or a video signal into a set of pixels for a computer. (The terms are often used interchangeably; by one convention, a scanner includes the image-sensing hardware, whereas a digitizer works on the output of a separate image sensor, such as a video camera. I will use the term scanner generically to refer to this entire class of equipment.)

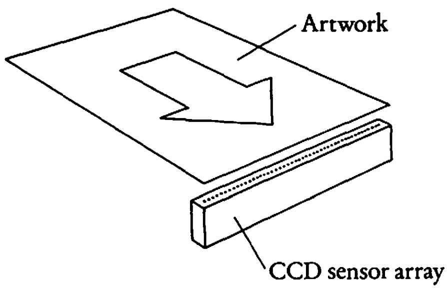
On the Macintosh, all scanners connect via the serial ports. How scanners operate and the types of information they store vary widely, but they come in three basic varieties (all descriptions here are for the initial released hardware and software):

- "Printer" scanners take advantage of a scanning platform most Mac owners already have—the Imagewriter printer. The $230 ThunderScan system replaces the standard printer ribbon cartridge with a light source and photoelectric sensor built into a cartridge housing. You roll the artwork into the platen and Thunderware's software runs the printing mechanism, rolling the artwork up while the sensor scans horizontally as if it were printing. ThunderScan can digitize an area up to 8 by 10 inches.



ThunderScan cartridge

- Array scanners employ a linear array of photosensors. Again, the artwork must be on paper. Low-cost array scanners are usually set up only for digitizing a sheet of paper. They work somewhat like a photocopier. You feed in the artwork or lay it down in the scanner; the array spans the width of the paper, digitizing a line at a time while either the paper or the array moves lengthwise. The moving array form is safer, since paper feeds can mangle some artwork. Prices are moderately high, $2000 and up, depending on accessories. An optical character-recognition feature adds another $1000 to $8000. More expensive models are built like a camera. Laserfax, Datacopy, Dest, Microtek, and others make array scanners; several companies plan to support the Macintosh specifically.



Array scanner

- Video digitizers convert the output of a television camera or other video source, including broadcast programs, video tape, or even a computer. The signal must conform to a standard form known as RS-170, however; Mac and many other computers do not generate such signals for their display. Prices, without a camera, range from about $200 to $600 for the common units. Some with special features can cost as much as $2500. The low-cost Micron Eye does not use a video sensor but rather a rectangular, solid-state array (actually a RAM chip without a cover; the memory cells are light sensitive).

Resolution

Resolution is a complex topic; I'll cover some issues here briefly.

Scanner resolution is most commonly quoted as the total number of pixels that can be created for a single image area. The following list gives the resolution of some scanners in pixels.

Video digitizers	
Koala MacVision	512 by 342
MAGIC	512 by 324
Micron Eye	256 by 128
Servidyne	511 by 322
ThunderScan	
1.0X magnification (72 dots per inch)	576 by 720
1.12X magnification with gray scale (80 dots per inch)	645 by 806
2.76X magnification without gray scale (200 dots per inch)	1589 by 1987
Array scanner	
200 dots per inch	1728 by 2200
300 dots per inch	2400 by 3300

The ThunderScan and most array scanners are designed as page scanners, to scan a sheet of paper, most often at life size. The ThunderScan magnifications apply to printing the scanned image on an Imagewriter at 72 dots per inch. Higher resolution printers can handle the higher dot densities shown without resorting to magnification. Similar considerations apply to the array scanners.

Video digitizers are limited by the quality of the incoming video signal. Few, if any, television cameras have sufficient resolution to faithfully reproduce such intricate detail as alternating black and white pixels on a horizontal scan line. Nevertheless, a high pixel count aids in positioning and helps render curves more smoothly. The color information in a video signal will obscure some detail, so a few digitizers, such as Servidyne's, include a switchable filter to remove the color signal. The best cameras to use are the high-quality black-and-white models designed for precision closed-circuit TV applications.

The maximum vertical resolution is set by the number of lines in the video signal. The television format (NTSC) used in North America and Japan imposes a 484-line limit (the usual count of 525 lines includes 41 black lines in the blanking interval, the black bar between pictures); digitizers that can handle the European CCIR format (the basis of PAL and SECAM color standards) can get up to about 570 lines of the nominal 625.

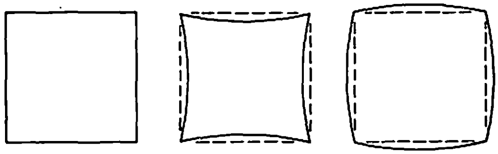
All television systems use interlacing, which breaks up a complete frame into two vertically offset fields—one of 242 even-numbered lines (285 in CCIR) and one of 242 odd-numbered lines. The odd- and even-numbered fields are supposed to intermesh exactly, but many low-cost television cameras cannot interlace precisely; often the two fields simply superimpose on each other. If so, you cannot get more than 242 lines of usable resolution, regardless of the digitizer's capabilities.

The page scanners don't suffer from the video-signal limitations, so they can cope with much more intricate detail than video digitizers; an Imagewriter cannot take advantage of their full resolution, but the LaserWriter can.

Geometric Accuracy

After scanning, will straight lines remain absolutely straight? Page scanners have high accuracy; the ThunderScan is limited by the precision of the Imagewriter mechanism, which is fairly good.

Video systems, on the other hand, suffer from many problems. Camera lenses, particularly zoom lenses, often exhibit visible barrel or pincushion distortion. Further, the camera pick-up tube, based on vacuum-tube technology, is nonlinear and gets worse with age. Professional video equipment incorporates circuitry to compensate, but it costs too much for most applications. The newer solid-state cameras and the Micron Eye do have perfect sensor geometry, so they are limited only by the accuracy of the lens.



Pincushion and barrel distortion

Scanning Speed

Scanning speed is analogous to a photographic camera's shutter speed. The fastest scanners are the video units with enough image-buffer memory to grab a frame in 1/30 second and hold it for digitizing. (For fast action, the buffer memory could store a single field in 1/60 second; specialized video cameras with external shutters can freeze even shorter action.) Most video digitizers don't incorporate a frame grabber, taking anywhere from 4 to 30 seconds. Without a frame grabber, the subject matter either must be stationary or you must create a stationary image. The still-frame function of home videocassette recorders isn't usually stable enough (although videodisc players are fine), but some industrial recorders have excellent still-frame abilities. Some new digital-signal processing televisions incorporate a frame buffer.

Slow video digitizing can be a nuisance while you're trying to focus a camera. If you have a slow digitizer, plug in an ordinary video monitor to get a live image for focusing and framing.

Array scanners take from 5 to 20 seconds to scan a full page. The ThunderScan is the slowest, taking 10 to 20 minutes for a full page, or about five minutes for a pixel count equivalent to a video scanner.

Gray Scale

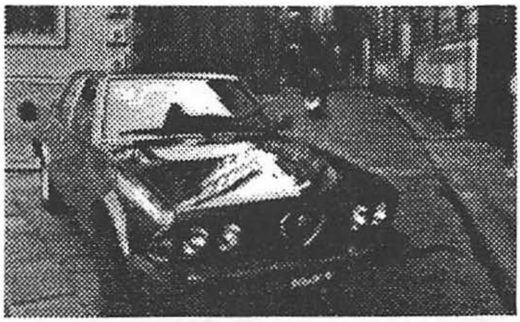
The Macintosh display devices—the CRT and printers—can only show black or white pixels; they cannot show gray in a region except by simulation with a pattern of on and off pixels.

The sensors in many scanners detect grays; what happens to the gray information varies. Some lower-cost page scanners detect black and white only, but if they can sense grays, most scanners give you some sort of contrast adjustment so you can arrange the distribution of gray steps between black and white. Set for high contrast, the scanner should drop grays altogether; you then set the threshold between black and white.

Gray-scale information can be handled by a scanner in one of two ways: rendered grays and stored grays. The simpler method, rendering, takes the original gray tones and reproduces them with a dot pattern that simulates shades on the screen and the Imagewriter. Some scanners, such as MacVision, do not send gray-scale information to the Mac; they simply render grays and then discard the original gray information. Changes in shading must be made by rescanning. This speeds up processing if you only want a MacPaint file, but you cannot adjust the appearance of grays after rendering.

For output on the Imagewriter, throwing away the gray-scale information is not usually a serious disadvantage. If you print on the LaserWriter, losing the gray information could be a serious disadvantage. With 300 dpi, the PostScript processor in the LaserWriter can create the equivalent of variable-sized dots to simulate grays. And the LaserWriter can print the higher resolution images of the page scanners, a resolution that video systems cannot match.

LaserTools (from Knowledge Engineering, New York) can convert ThunderScan's native files into PostScript code for printing with half-tone simulation.



The PostScript processor in the LaserWriter can reproduce grays by simulating a half-tone screen at 60 dots per inch. Courtesy Adobe.

The sophisticated scanners store the gray information with each pixel. For a MacPaint file, most of these systems assign a repetitive pattern of black and white pixels to each level of gray. With storage, you can change the pattern used to depict each gray level at leisure after scanning. This is a valuable feature with ThunderScan, since its scanning is so slow. In some cases you don't want a realistic-looking rendition; sharply different patterns can emphasize subtle differences in the image. And for many scenes, including portrait photographs, such repetitive patterns produce a less satisfactory appearance than the pseudo-random patterns of MacVision.

Two examples of rendered grays in a MacPaint image: repetitive patterns (left, from the MAGIC digitizer) and pseudo-random (right, from the MacVision digitizer). Courtesy Macworld *magazine.*

File Formats

All Mac scanners can save images in a MacPaint format, which may include a gray rendition. Some can also use their own "native" format that stores gray-scale information for each pixel. Unfortunately there is no standard gray-scale format so the various scanners create incompatible native files. MacPaint stores a single bit per pixel and uses a compressed format, so a long string of black or white pixels is coded into a single number rather than as separate pixels. Gray-scale storage adds several bits per pixel and does not compress as easily. ThunderScan will generate files of 100 kilobytes or larger in many cases. It does not use data compression in its native format, so to save disk space, it drops gray information when digitizing large areas at more than 72 dots per inch. Array scanners create even more data—half a megabyte to a megabyte before grays, and easily two to five megabytes with grays. High-capacity hard disk drives are essential with these scanners.

Facsimiles

Facsimile machines are in wide use at corporations, sending text and graphics over the telephone lines. The widespread international Group III standards use 200 dpi resolution and a compressed transmission format, with 9600-bps modems; the latest units use 400 dpi. In principle, a page scanner could serve as a facsimile input device and a laser printer as the output device, but software and hardware support for this function has been slow to appear.

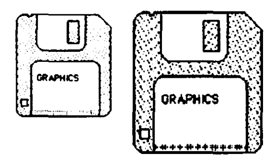
Other Scanner Issues

Scanner procedures vary greatly in actual use. To scan artwork on paper, page scanners are much easier to use than video cameras, which require considerable space for a copy stand and a set of lights. For three-dimensional objects, page scanners are not as quick or efficient as video scanners. You can take a photograph first and then scan the photograph, but what if you want to change the angle of view slightly? An instant camera comes in handy. Photographs are a problem for ThunderScan because instant photographs and double-weight photographic paper do not go through the Imagewriter smoothly.

As for video scanners, you can paste two video scans together to create a larger image, but getting two images to line up exactly is difficult to impossible. For three-dimensional objects, you cannot simply pan the camera on a tripod to make the second scan; because of perspective differences, the images will not match. A telephoto setting on the lens will make matches slightly easier. Page scanners are free from such perspective problems if you use a wide-angle lens for a photograph first.

After scanning, many images will need touching up for the best results. Often the image won't be exactly the size you need. If the output will go to an Imagewriter, scale the image before scanning. With a page scanner, an enlarging/reducing photocopier helps. With a video scanner, you can quickly resize the object with a zoom lens or by moving the camera. You can scale images in MacPaint, but the scaling process introduces distortion artifacts.

If you have a LaserWriter, MacDraw does much nicer scaling and lets you take advantage of the printer's full resolution. For the best results, you can scan an image at four times the finished size, do any touch-ups on the large image with MacPaint, and then scale to quarter size with MacDraw for printing. This gives you a picture with 300 dots per inch, with each LaserWriter pixel corresponding to a single screen pixel. Using this method, rendered grays, particularly MacVision's random patterns, are effective.



Stretching an image in MacPaint distorts some objects.

Since a scanner can function as a copier, copyright laws may apply to the images you scan.

CLIP ART

In 1984, many companies rushed out clip art, or bits of MacPaint art on a disk. Most clip art is not very useful—how many times will you ever need a picture of Albert Einstein? A scanner, particularly ThunderScan, makes most clip-art collections superfluous; you can buy a scanner for the equivalent of just a few clip-art purchases.

But because of the detailed touch-ups necessary with some scanner images, some specialized disk-based clip art can be useful. Decorative borders, for example, are hard to get exactly right with a scanner.

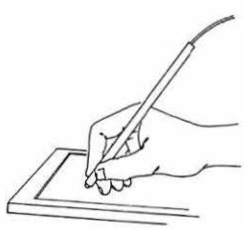
Using Existing Artwork Without a Scanner

If you want to work from existing artwork without a scanner, you have four basic choices:

- Trace the artwork with a digitizer pad.
- Transfer the artwork to a transparent overlay that fits over the screen.
- Trace the artwork with a mouse.
- Use a slide projector.

For the techniques involving tracing, you may not want to trace entire lines. Rather, you may want to trace the end points of each line and other critical features and, after tracing, add straight lines and curves using MacPaint's tools.

Tracing the artwork with a digitizer pad



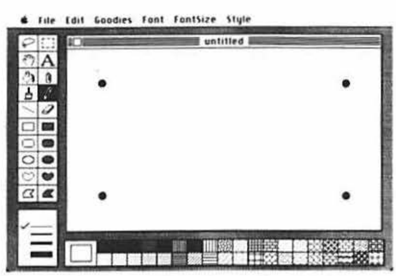
A digitizer pad lets you draw or trace with a pen-shaped stylus, a more natural action than drawing with a mouse.

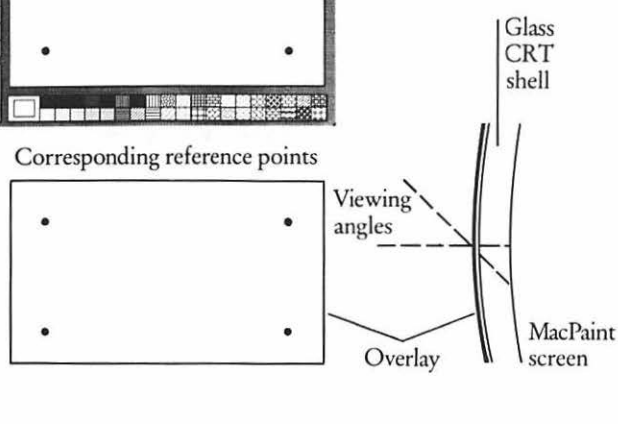
The Summagraphics MacTablet lets you draw on a Macintosh screen with a natural drawing action. You simply pick up and move the tablet's stylus as if it were a pen. You can place the artwork on the tablet for tracing and track your work with an optional stylus tip, which incorporates a ballpoint pen. The MacTablet offers two ways to press the equivalent of the mouse button: by pushing down on the stylus tip, and by pressing a small switch built into the side. Another option, a "puck," helps for precision work (see the illustration in Chapter 13). The MacTablet is an absolute-position device, like pen and paper—far superior to relative-position tablets.

Making an overlay

An overlay is pretty difficult to work with, but serves for a quick-and-dirty shortcut. To make an overlay, transfer the original image to a transparent sheet sized for the Macintosh screen.

Xerographic overlays are the quickest, provided that you have a suitable copier. Copiers with continuous enlargement and reduction work best. All the overlay-tracing techniques lead to parallax problems because of the glass CRT faceplate separating the overlay and the





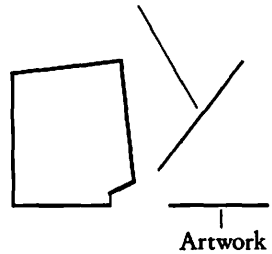
Use reference points when tracing an overlay; otherwise your angle of view will introduce errors in position.

phosphor. While tracing, you must hold your head in the same position; to do this, mark two or three reference points around the overlay before you start tracing, and check your head alignment against these points before tracing each object.

For a photographic overlay, the Polaroid process is the quickest. Use black-and-white transparency film; for full size, you will need the 4- by 5-inch film, although the smaller 3¼- by 4¼-inch film size may cover enough of the artwork for your needs. You can also make the overlay in several sections.

To make an overlay by hand, use transparent acetate and a drawing pen with suitable ink. If you are lucky, the original artwork will have the right scale and you can simply trace it. If you need to change the size, use a pantograph. Then simply tape the acetate over the screen and use the mouse to trace the acetate image. It's not easy, but eye-hand coordination will come with practice.

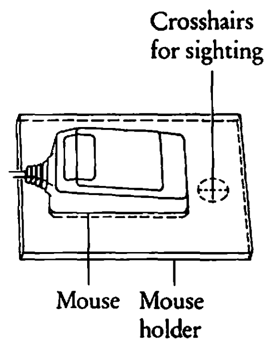
In a variation of this process you can install a half-silvered mirror at a 45-degree angle in front of the screen.



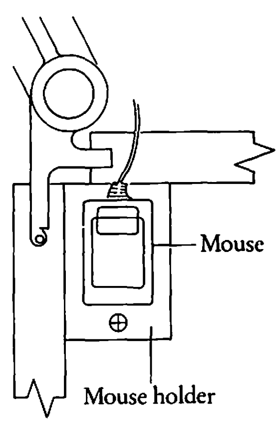
Tracing the artwork with a mouse

A mouse isn't really designed for tracing objects. It registers relative motions—the distance and direction you have moved, not the place you started and stopped. If the rolling ball inside the mouse slips, you merely move the mouse a little farther while watching the pointer on the screen.

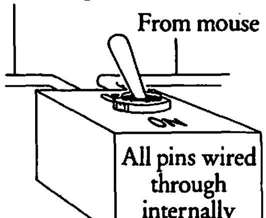
If you are careful and willing to touch up your work, however, the mouse will serve for tracing art. When you trace with the mouse, you must maintain a strict up/down, left/right orientation, so it's best to fix the mouse to a drafting machine, which will maintain orientation over a drawing table. Make a mouse holder out of Plexiglas and fix it to the drafting machine. Several commercial units work in this fashion.



A Plexiglas holder for a mouse



The mouse holder lets you use the mouse with a drafting machine, maintaining up/down, left/right orientation when tracing with the mouse.



Accessory switch for mouse button; mouse button remains operational.

While tracing, you must drag the mouse, but pressing down on the mouse button is a nuisance when the mouse is attached to a drafting machine. You can add an external switch to duplicate the mouse button. (The mouse button connects pin 7 on the cable to ground—pin 1 or 3.) With a toggle or latching switch, you simply turn the mouse on and off for each line, so you can concentrate on the tracing rather than on holding down the mouse button.

Again, you can use a pantograph to change the scale of your original artwork. The alternative is changing the mouse-driver software—altering the "gearing" between the mouse ball and screen movement through software. These changes are relatively easy only for factor-of-two changes; other changes are more difficult.

Using a slide projector

For quick-and-dirty results, you can project the artwork directly onto Mac's screen with a slide projector and use the mouse to trace the artwork with the pointer. This technique is only for the desperate.

FONTS

Fonts were originally developed for a sculptured medium with letters chiseled in stone and then cut in metal type, or for careful calligraphy by a skilled scribe. With any new medium, the font design must be modified. Throughout history, designers have taken years or even decades to understand a new medium. In recent years technology has changed much faster than aesthetic sensitivity, so most printing now suffers from clumsy adaptations of older designs.

With computers, the problems have gotten worse. The Macintosh has given many people the power to choose fonts for the first time. But that power is easily abused; most people aren't sensitive to the fine points of typography. And the coarseness of the Imagewriter has not helped; its resolution of 72 or 144 dots per inch with a 16-mil (0.4 mm) dot size cannot create graceful text. The LaserWriter, with 300 dots per inch and a 3.3-mil dot size, improves matters considerably but still falls far short of true typeset quality. However, the LaserWriter does open the way to adequate typography at a (relatively) low price.

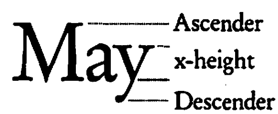
Fonts for the Macintosh come in three forms:

- Screen fonts, designed for the screen and the Imagewriter.
- Fonts for the screen that correspond to laser fonts.
- Laser fonts, designed for the LaserWriter.

Screen Fonts

Mac typeface quality varies with the font and type size and depends on the way the font data are stored. Because storing every dot (a bit map) of every font in every size takes up so much memory, Mac stores bit maps of some sizes and creates the other sizes by scaling them up or down. In the process, some scaling factors work less well than others. Scaling a 24-point original down to a 12-point copy, for example, generally works well because it is a simple matter of dropping every other dot, but scaling the size down to 18 points may leave rougher edges. However, independent companies now supply fonts in the missing sizes, so scaling is becoming increasingly unnecessary.

The proportions of some screen fonts have been fudged for legibility. Most commonly the x-height is higher than it should be in some small sizes because the designers have few pixels to work with; for example, some 12-point sizes use the x-height of a 14-point font. And in some fonts such as Geneva, line weights in the larger size have been kept thin so they look correct when scaled down for the Imagewriter high-resolution printing.



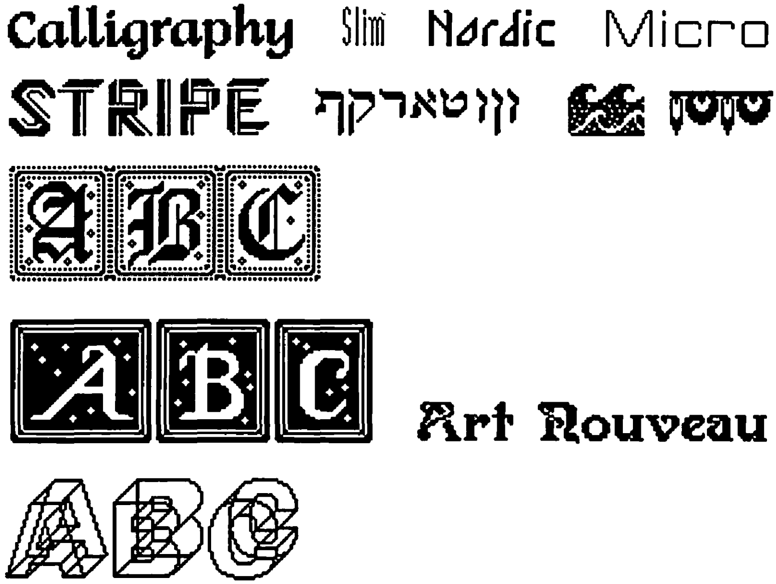
The Imagewriter's high-resolution mode needs a font twice the finished size for the best results, since it uses the larger size to interpolate the dots. If the double-size font does not exist, the printer driver looks for a four-times-size font. If that doesn't exist, it scales the font from another size.

The point sizes given in the style menus of many Macintosh programs are close to, but not quite the same as, the standard printer's point measurements. An American/British printer's point equals 0.013837 inch, or about 72 points to the inch. A European or Didot point equals 0.3759 millimeter or 68 points to the inch. The Macintosh screen fonts and the Imagewriter use 72 points to the inch.

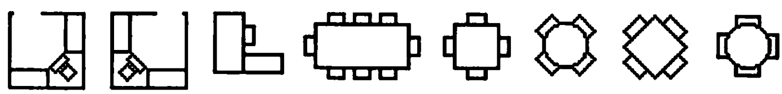
Using screen fonts

Many Macintosh fonts include symbols, such as ● ¶ ® © and true open and close quotes (“ ” and ‘ ’) not found on typewriters or most other computers. Most fonts also have three dash lengths: hyphen -, en-dash –, and em-dash —. Use these symbols as appropriate, but you shouldn't use them when the file will be transferred to another computer that won't recognize them.

You can choose from hundreds of screen fonts. Many fonts are only available in the display sizes (18 point and larger) because of limited screen resolution. Here are some samples:



From Fluent Fonts by Casady Company



From Commercial Interiors by Hayden

Although the new laser printer fonts will change the way typesetting is done with the Macintosh, here are some screen fonts that approximately match typesetting fonts:

New York: Garamond Condensed (ITC–International Typeface Corp.)

abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
1234567890.,;:''&!?$

abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
1234567890.,;:`'&!?$

Geneva:
Helvetica Light Condensed (Haas),
Antique Olive (Letraset)

abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
1234567890.,;:`'&!?$

Toronto:
Helserif (Mergenthaler),
Lubalin Graph Book (ITC)

abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
1234567890.,;:`'&!?$

Monaco:
Avant Garde Condensed (ITC),
Bauer Topic Bold (VGC–Visual Graphic Corp.)

abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
1234567890.,;:□'&!?$

London:
London Text (Compugraphic),
Cloister Black (Mergenthaler)

abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
1234567890.,;:`'&!?$

Athens:
City Medium (Berthold),
Neo Edelweiss (The Headliners International, Inc.)

abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
1234567890.,;:`'&!?$

Venice:
Gavotte (Mergenthaler),
Zapf Chancery Medium Italic (ITC),
Book Jacket Italic (VGC)

abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
1234567890.,;:`'&!?$

Chicago:
Cruz Tempor Medium (John Schaedler, Inc.),
Bessellen (VGC)

abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
1234567890.,;:`'&!?$

Seattle:
Helvetica Regular Condensed (VGC),
Machine (ITC)

abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
1234567890.,;:'&!?$

Los Angeles:
Balloon (VGC),
Hand-lettered Script A-71 (VGC)

All these fonts, except for Monaco, are proportionally spaced. Monaco is a monospaced font, as on a standard typewriter.

Fonts can be used not only for text but also for pictures and symbols; the only limitation is on size—a maximum of 127 points high or 32 KB total file size. There is not enough space to store the entire character set for fonts larger than about 60 points, so if you create or modify such fonts, you may have to split the upper- and lowercase letters into separate files.

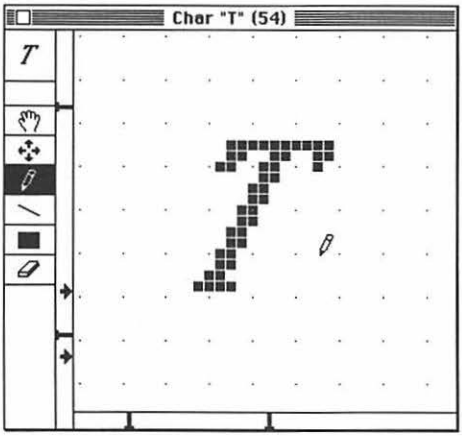
With a symbol font, you can recall electronic symbols or furniture layouts (Commercial Interiors, from Hayden) with a keystroke. The symbol fonts are especially valuable in MacPaint and MacDraw.

Many application programs cannot change fonts; some will handle only a single font at a time, so you cannot mix them. Other restrictions may apply. MacWrite, for example, always maintains at least a one-pixel spacer between lines, a nuisance with scientific notation. Microsoft Word can be set to butt two lines together (use the line-spacing command in paragraph formats).

Some programs are limited by screen size in the number of fonts they can handle; you cannot install more than 20 fonts in MacWrite or MacPaint because you will run out of menu space. Microsoft Word will allow more, since it uses a scrolling font menu. Some programs can only handle fonts up to a particular size. Large-sized or too many fonts create problems because they take up so much space. 128 KB Macs and single-sided system disks run into trouble fairly quickly.

Font Editors

FONTastic (version 2.0, from Altsys, Plano, TX) is an excellent utility program that lets you manage, edit, and create screen fonts. It duplicates all the font functions of Apple's Font/DA Mover program and much more. You can copy and paste characters from font to font, change selected characters, resize fonts and smooth over rough edges with a set of MacPaint-like tools. If you need special symbols, you can create or paste them into a standard font, perhaps replacing a foreign-language character you never use.



FONTastic

The screen fonts use algorithms to create the italic, boldface, and other styles. The results are pretty weak, especially for the italic forms. If you have the patience, you can create a special italic font with FONTastic. However, creating new fonts, or even modifying an existing font, is time-consuming and requires some skill.

So that the system can keep track of them, each screen font is assigned a number. It doesn't really matter what number goes with a particular font, only that the installed fonts have distinct numbers. FONTastic automatically changes numbers when required to prevent confusion. It also lets you assign a font to a particular program rather than to the system file, so a decorative font can be linked to MacPaint, for example, and it won't get in your way when you use MacWrite. In addition, FONTastic lets you paste images from MacPaint into characters so you can create your own symbol library.

A request

With all this freedom, many users are tempted to go overboard, using too many fonts. Please use the fonts with care. Don't make all your correspondence look like kidnappers' ransom notes; don't commit crimes against typography. Look at well-designed books and magazines and note how few fonts and typestyles they use.

The only use for the San Francisco font is as an example of bad taste.

Screen Fonts and the LaserWriter

Fonts designed for the Imagewriter and screen will print on the LaserWriter. You can select a smoothing function, but things will rarely look just right and the fundamental sizing and spacing remain set for 72 dots per inch. For the best results, use fonts designed for the LaserWriter.

LaserWriter Fonts

With only 300 dots per inch, a laser printer does not create typeset quality but it can do credible simulations of traditional typefaces. The sequence of constructing fonts on the LaserWriter, and the corresponding screen fonts, were described in Chapter 16.

Times Roman *Times Italic* **Times Bold**
Times Bold and Italic Outline
Helvetica *Helvetica Oblique* **Helvetica Bold**
Helvetica Bold and Oblique Shadow
`Courier` *`Courier Oblique`* **`Courier Bold`**
`Courier Bold and Oblique`
Symbol: θωερτψπΣΠℵ±∨⇔⇐⇑⟩|∫⟩ | |⊕⊗

variable size Big

Laser fonts are specified as a set of outlines (analogous to MacDraw) rather than a set of pixels (analogous to MacPaint) as with screen fonts. Italic and boldface fonts have their own outlines, a great improvement over the simple algorithms used by the screen fonts. But there is only a single outline for all sizes, optimized for the 12 point size; large sizes look a little out of proportion. In comparison, the designers of the best traditional fonts adjust the proportions subtly for different sizes. Standard phototypesetting machines also use a single master, but that is one of the things wrong with phototypesetting. Of course, the power to scale fonts to an arbitrary size is fabulous, even if imperfect, but we should never accept a compromise merely because it is the easy way out for some computer. Ultimately, a separate set of outlines could be made for the larger display sizes.

The intercharacter spacing in the LaserWriter follows a simple geometric rule with changing type size. For small sizes—those under 10 points—the geometric rule sets the text too tightly. Some software lets you expand the spacing for legibility.

With 300 dpi fonts, you can use typographic touches. Kerning adjusts the spacing between specific letters—particularly useful in headlines. For example,

WAVES becomes WAVES with kerning.

The LaserWriter permits kerning in 1/300-inch increments, but depends on the Macintosh application software to send suitable positioning instructions. A complete kerning capability will automatically adjust spacing for specific letter pairs, taking the font and size into consideration.

For clean results with right-justified text, particularly if you use two or three columns on a page, you will need to hyphenate the text, adding word breaks to suppress loosely set lines. The editing program should permit conditional or discretionary hyphenations so words will be broken with a hyphen only when necessary. Hyphenation is a tedious process, so many companies have developed automatic hyphenation programs for typesetting systems, using a spelling list to insert conditional hyphens. But no program can be completely automatic. The noun *pres-ent* breaks differently from the verb *pre-sent.* The best hyphenation programs flag such homographic sounds and warn you if there will be more than three hyphens in a row. A separate spelling list adjusts for the variations between American and British hyphenation.

As this is written, no hyphenation program has appeared for the Macintosh; until one does, you can either hyphenate manually or use a ragged right. Legibility studies have generally shown that ragged right is easier to read.

For an excellent survey of typography and aesthetics, see *The Typ-Encyclopedia* by Frank J. Romano (R. R. Bowker, 1984).

Adobe's laser fonts include two ligatures, the connected letters fi and fl. Unfortunately, ff, ffi, and ffl are missing. To handle ligatures properly, the editing software should create and split such characters as required.

Font editors for laser fonts will be available; generally they will be useful mainly for modifying existing characters. Making new characters calls for considerable skill; most fonts for the many laser printers on the market look crude. The first good laser-printer fonts are just beginning to appear; the best of these are designed specifically for 300 dpi xerographic rendition rather than simply copied from another medium. For a good discussion, see "Digital Typography" by Charles Bigelow and Donald Day in *Scientific American,* August 1983.

Typographic Composition

For several years, many people have used microcomputers to prepare text for typesetting equipment. Special typesetting commands are embedded in the text file with a word processor. The typesetting commands usually make the text nearly illegible and the process is not interactive; you cannot see what you are doing until the page comes back from the typesetter. The TeX typesetting language by Donald Knuth and JustText in LaserTools work in this fashion; a command interpreter converts the code into PostScript form for a LaserWriter or typesetter. Some programs using embedded commands may let you "print" to the screen to preview the finished page.

Some software that drives the LaserWriter won't position material more accurately than the screen resolution allows. This resolution is adequate for routine work but not exacting enough for precise layouts. The most flexible programs will magnify pages on the screen for greater accuracy and let you specify positioning with typeset precision, far exceeding the capabilities of either the Imagewriter or the LaserWriter.

For the most detailed work, however, you may need to send commands directly to a typesetter. You can embed commands within PostScript for this purpose; a PostScript processor in the typesetter takes the appropriate action. (For more on typesetters, see the final section of this chapter.)

Ultimately, a modified form of PostScript will drive the CRT itself so the system will have to do less work translating images from one form to another.

PAGE-MAKEUP SOFTWARE

Most Macintosh word-processing programs can mix text with graphics on a single page, and the more sophisticated can print in multiple columns. For complete freedom in designing a page, however, you will need a page make-up program. Such a program lets you mix headlines, text, and graphics anywhere on a page. Text can be broken into several parts and continued on successive pages. Graphics can be freely intermixed on a page and text can be adjusted to print alongside a picture.

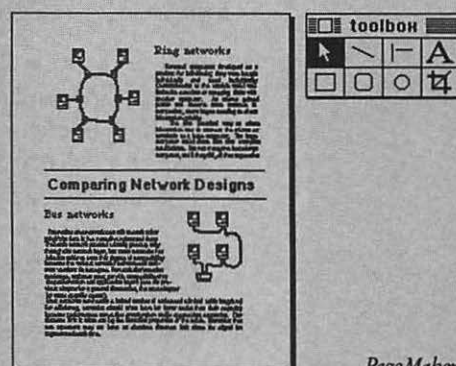
Page make-up programs perform the electronic equivalent of paste-up, a process whereby a graphic designer lays out all the components of a page. In addition to text and pictures, a page may contain rules (lines that separate page components), decorative borders, and initial caps, the large initial letters used to set off a portion of text. Not all page-makeup programs can do every function, however. Three of these programs are described briefly in Table 17-1.

TABLE 17-1. PAGE-MAKEUP PROGRAMS

ReadySetGo: From Manhattan Graphics; quick and easy to use—drag a rectangle on the page dummy, then fill it with text or a picture; built-in editor lets you change fonts and sizes in the text. ReadySetGo will not read a MacWrite or Microsoft Word file directly; you must go through the Clipboard or Scrapbook and reformat the text; Version 2.0 handles 16 pages at a time.

MacPublisher (version 1.26): From Boston Software Publishers; can handle 32 pages at a time; can print articles at 133, 150, and 200 percent of the final size, so appearance will be improved by reduction in the final printing; transparent ruler tool lets you measure off lengths of text or size a picture. As with ReadySetGo, includes its own editor and only accepts text from other files via the Clipboard/Scrapbook; editor has limited formatting; each text block must be in the same font and style; a text file can be cut with a scissors tool and continued on other pages. You can define standing elements, text, or graphics that will appear on each page.

PageMaker (pre-release): From Aldus Software; more expensive than the others; deals with 16 pages at a time; you can specify left and right page layouts; will accept formatted files from MacWrite and Microsoft Word. Text can be run onto continuation pages; you can pull up or push down one part, and continuation pages are adjusted accordingly. You can view the pages on the screen in several sizes, including a two-times magnification to check positioning.



PageMaker

FINISHED COPIES

For the best results with an Imagewriter, print the page oversized and reduce the image for the final copy; the reduction decreases the effective dot size and increases the visual density of the dots. Even the LaserWriter can benefit from a reduction process. Reductions work fine if you want a smaller finished size to begin with, but can be a pain if you must paste up pages again, the very process that all this software and hardware was supposed to replace.

TYPESETTING

For really clean results, only typeset quality will do. Allied Linotype has announced a typesetting system for Macintoshes. The system includes a raster image processor (RIP) that connects to AppleTalk. The RIP receives information in PostScript codes and converts the information into a raster for two typesetters:

- The Linotron 101 ($30,000, including RIP) creates finished copy at a resolution of 1440 pixels per inch, with a maximum line length of 11.7 inches and page depth of 45.2 inches. Its 68000 CPU has 512 KB of memory; it creates multicolumn pages at 180 lines per minute.
- The Linotronic 300 ($57,000 including RIP and required CRT terminal) achieves a maximum resolution of 2540 dots per inch, with a 12-inch line length and 25.8-inch page depth. It also uses a 68000 CPU, but has 1.5 MB memory; it operates at 225 lines per minute for multicolumn pages.

Both units use a helium-neon laser writing directly onto photosensitive paper or film. The photosensitive material must be handled in light-tight cassettes and requires photographic processing, so these and other typesetters are not a general-purpose substitute for conventional printers. The results are clean, with smooth, even black areas, and free of the stray toner that mars the white areas in laser printer output. The dot size can be adjusted by varying the exposure. Laser typesetters achieve much higher resolution than laser printers since the dots are not smeared during the toning and fusing steps of a xerographic process.

Modems Chapter 18

Modems transmit data by converting a computer's digital signals into modulated audio tones that can travel on a telephone line. They also do the reverse, demodulating the audio tones from a distant computer and modem back into digital form for your Mac. ("Modem" is contracted from MOdulator/DEModulator.)

MODEM TYPES

Modems for microcomputer use operate at three speeds: 300, 1200, and 2400 bits per second (bps). (The term "baud" is commonly and inaccurately used in place of bits per second. For a precise definition and a table of true baud rates, see Chapter 25.)

In terms of speed, 300-bps modems transmit 25 to 30 characters per second (cps). Thirty cps is slow, taking about a minute to fill a Mac screen with characters, or up to 12 minutes to fill it with graphics. Despite this slow operation, 300-bps modems have been popular among hobbyists because of price—from $50 to $200. Higher-speed modems use more expensive, more precise components; 1200-bps models cost from $150 to $600. But unless you are on a very tight budget, 1200-bps modems are worth the extra money. They operate four times as fast as 300-bps modems, transmitting 100 to 120 cps. Most 1200-bps modems, including the Apple model, will also operate at 300 bps. Continuing up the scale, 2400-bps modems are twice as fast as 1200-bps models, and Telebit's newest 10,000-bps modem offers great promise.

Compatibility with Other Modems

Compatibility among the many modems and communications schemes available is an extremely complex subject. For now, I will simply say that 300- and 1200-bps modems sold for microcomputers in North America use the same signaling frequencies, so they are compatible as long as they operate at the same speed; 2400-bps modems are a little more complicated. I discuss this in depth in Chapter 25.

Modems follow certain rules—a protocol—that specify such details as which audio frequencies will carry the information. All 300-bps modems use the Bell 103 protocol, while most modern 1200-bps modems use the Bell 212A protocol. Bell Laboratories originally set the standards and gave the protocols their names. Avoid other 1200-bps protocols, at least in North America.

The newer 2400-bps modems use the V.22 bis protocol in both Europe and North America. However, few manufacturers follow every detail of the V.22 bis recommendations, so both subtle and glaring problems can occur. If you use 2400-bps within a company, you should adopt a single modem model. In addition to setting the correct modem protocol, you must also make sure that several other layers of protocol are compatible. In Chapter 8 I gave a simple guide to setting these protocols; Chapter 25 covers them comprehensively.

Most microcomputer modems only operate asynchronously; they do not support synchronous communications.

Modem/Software Compatibility

Most modems contain control circuitry able to dial phone numbers, check the phone line, and hang up. To take advantage of these features, your communications software must send suitable instructions to the modem. The most popular instruction set was established by Hayes for the Smartmodem series; Apple and most other modems now sold for microcomputers use the Hayes instruction sets. Unfortunately, many, including the Apple modems (see Technical Note at the end of this chapter) fail to emulate the Hayes exactly. The minor differences do cause some problems, for some software will run only on a specific modem model. But this compatibility problem affects only the commands between the software and the modem, not modem-to-modem compatibility.

Well-designed communications software includes installation procedures for different types of modems. Even so, whenever you buy a modem, make sure that it will work with your software; as always, be wary of compatibility claims. Many companies claim that their modems work just like the Hayes, but they often don't.

MODEM FEATURES

Beyond the now-standard Hayes commands, some modems come with additional features. Callback modems are a good choice for security and privacy; see Chapter 24 for details. Some modems include a built-in memory for storing telephone numbers to dial but, since most communications software can now store the telephone numbers on disk, having independent storage in the modem doesn't matter.

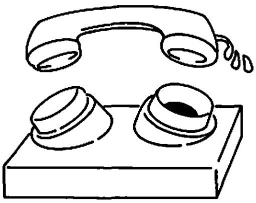
A more useful feature is a memory buffer that stores incoming messages. The memory is active even when the computer is not turned on. With such a buffer, you don't have to be present when a message comes in; the buffer behaves like a private electronic mailbox. Anchor Automation's Signalman Computer Mailbox and the Prometheus ProModem can store up to 64 KB of incoming information. The Hayes Transet 1000 buffer works in conjunction with a modem and can store up to 256 KB. However, you must dedicate a phone line to the modem; in most cases, a commercial electronic mail service is simpler and cheaper.

CONNECTING THE MODEM

Whenever possible, the modem should connect directly to the telephone line through a modular jack. If your phone uses a four-pronged plug, you should buy a high-quality adapter plug; the cheap adapters aren't reliable.

Modems should not be connected to party lines. Another party picking up a telephone will disturb the computer link; in an emergency, another party cannot ask for the line.

Some telephone convenience features, such as call waiting, that produce an audible click may interfere with computer links. If interference is rare, you may be able to live with it; otherwise you should disconnect these telephone features. A modem will hang up if the signal is interrupted. A few newer modems can tolerate and ignore the click, but the remote modem may still hang up.



Acoustic cups fit over a standard telephone receiver.

If you are traveling, you may not be able to connect your modem directly to the phone line. You will have to resort to an acoustic modem with cups, with a small microphone and speaker that fit over a standard telephone receiver. Acoustic modems are less reliable because of noise and other problems. Always use a direct-connect modem when possible.

USING MODEMS

The efficiency of modem communications varies with phone-line quality. Modem connections using local calls generally work satisfactorily, but long-distance calls frequently run into problems because of noisy lines. You may find that some long-distance services have noisier lines than others. The line quality can also vary with the time of day; during peak periods the lines will have more cross-talk (leakage from other conversations) than they will at nighttime.

Also bear in mind that the lines of many long-distance services simply won't carry information at 2400 or even 1200 bps. If you have trouble at 2400 bps, try another line or switch to 1200 or even 300 bps and try again. In some cases, the local phone company can check your line for transmission quality.

Dialing

Most modern modems now include automatic dialing and answering, so you can dial the telephone number from the keyboard (or use numbers stored on disk) and the receiving modem can automatically answer the call. Take care to turn off your modem's auto-answering feature unless incoming calls on the line are exclusively from computers; otherwise a human caller will be greeted with an irritating, high-pitched tone.

Hayes-style modems can dial either tones or pulses; you select one or the other depending on whether you have touch-tone or rotary-dial service. If the dial tone does not stop when the modem starts dialing, check if your telephone service can handle touch-tone dialing. If so, find a way to reverse the red and green wires of the phone line.

Long-distance charges and billing

Whenever possible, you should make modem connections to other computers purely electronically: An auto-dial modem should call an auto-answer modem. In some cases, however, you may need to start the link by calling the operator. A typical case might be:

- Dial the phone number manually through an operator.
- When the operator answers, arrange for a collect or credit-card call.
- Wait for the connection to be completed.
- If the call is collect, the other party must accept the charges.
- Arrange with the other party to switch to data mode.
- Switch together to data mode.

Unfortunately such switching is difficult or impossible to do with many hardware/software combinations. You'll need software that lets you send a switching command to the modem.

You don't always have to deal with a human operator for a credit-card or an alternative long-distance service call. The exact procedure varies by area, but generally you do one of the following:

Credit-card call (touch-tone dialing services only; not yet available in all areas):

- Dial 0 and the complete long-distance number.
- After two or three rings, you hear an acknowledgment tone.
- When the tone sounds, dial your credit-card number. (If you wait, the special tone stops and a human operator comes on the line.)

Alternative long-distance service call:

- Dial a local access number.
- After two or three rings, you hear an acknowledgment tone.
- Dial your account number.
- Dial the complete long-distance number.

To perform the entire dialing procedure with an auto-dialing modem, you need to time the delay between completion of dialing and the start of the acknowledgment tone. For Hayes-style modems, insert one comma into the dialing string for each two-second delay that you need.

For example, if you are dialing from inside a company through a private branch exchange (internal switchboard), creating the dialing string 9,02135551234,,,1222-5551111 will cause the modem to:

- Dial 9, and wait two seconds, allowing time to receive an outside dial tone (Apple and Hayes modems do not detect dial tones).
- Dial 0-213-555-1234, indicating to the phone company that you are making a long-distance call with assistance—but will not actually use operator assistance.
- Wait six seconds, long enough for the call to clear the phone company's exchange, ring the operator twice, and get an acknowledgment tone.
- Dial 1222-5551111, the credit-card (calling-card) number during the acknowledgment tone. The operator will not come on line if the card number is dialed at this time.

For the exact sequence, consult your modem and communications-software manuals. You can change between pulse and tone dialing with simple commands.

Making the Connection

If you initiate the call, your modem will operate on the originate frequencies and the receiving modem on the answer frequencies. The two frequencies let each party send information simultaneously without interference.

Most Hayes-style modems contain an internal speaker so you can hear the connection until it switches over to data communications. With two 1200-bps auto-answer modems, this is the exact sequence:

- Your modem dials the other modem.
- The other modem answers the line with a 300-bps answer tone.
- Your modem responds with a 1200-bps, phase-modulated originate tone.
- The other modem answers with a 1200-bps answer tone.
- Your modem switches to data mode and disconnects its speaker.

If the remote modem is a 300-bps unit, it will not respond to your modem's 1200-bps originate tone and so your modem will switch to the 300 bps speed.

If a person answers the phone, you should pick up an ordinary telephone and hang up the modem electronically.

Given the frequent problems with getting communications to work, you will probably find having a second phone line handy. With it, you can talk to the person operating the other computer while setting the parameters. Once you have established a link, store all the parameters. The connection should be easy next time, unless the line conditions have changed.

A TECHNICAL NOTE

The Hayes commands not supported by Apple/U.S. Robotics modems: AT C series, AT O, and AT R. Of the AT S series, Apple/U.S. Robotics recognizes only AT S0, AT S2, and AT S7. The Hayes "escape" and hang-up code (pause, + + +, pause, AT H) is simplified to (pause, + + +, pause).

Macintosh Software Issues

Chapter 19

This chapter discusses several different software topics. If you have used other microcomputers or already have some experience with the Macintosh, you should find this grab bag of topics useful. The topics discussed here include:

- The User Interface, a comparison of visual interfaces with earlier designs.
- Other Operating Systems, a discussion of the advantages and disadvantages of running other operating systems on a Macintosh. This section also discusses the hardware necessary for using some other operating systems.
- Choosing Software, a selection of general guidelines on how to buy software that best meets your needs.
- Copy Protection, an overview of the different types of copy protection you may encounter.
- Exchanging Information Between Programs, a look at what you can and cannot do when transferring information.
- The File Structure, a collection of notes on how Mac stores information.
- Why the Macintosh Is So Slow, a discussion of how software affects Mac's operating speed and what can be done.
- Miscellany, a brief look at some types of software not covered elsewhere in this book.

THE USER INTERFACE

At any given time while operating a microcomputer, you make choices—to open a file, close a file, insert or delete text, and so forth. Software designers face two key problems:

- How does the computer show you what choices are available?
- How do you indicate your selection?

Software designers have developed several types of interface as solutions to these problems. First, I will briefly survey earlier solutions, then discuss Mac's visual interface.

Conventional Interfaces

All conventional interfaces are based only on text because many computers cannot handle graphics well.

Command lines

With a command-line interface, the screen shows a prompt, such as A> or $, indicating that the program is ready to accept a command. There is no other information; you must know what to type next, and you must type it exactly.

Command lines are the easiest interfaces to program and the hardest to learn, but they offer great flexibility because the program can recognize any keyboard entry. Examples of command-line interfaces include CP/M-80, MS-DOS, and traditional Microsoft BASIC.

Menu initials

With a menu-initial interface, you are presented with a menu of single-letter abbreviations for possible choices. Given a display such as B C D E F G I M, you must know which letter represents your choice. Often your choice results in the display of a second-level menu with more letters prompting a further choice. Still fairly cryptic, menu initials are nevertheless a little easier to use than command lines. Examples of menu-initial interfaces include the early versions of VisiCalc and SuperCalc.

Menu words

With a menu-word interface, instead of initials, you are given a list of words representing the possible choices. Again, selecting an item may take you to a second-level menu with more words prompting a further choice. Less cryptic than menu initials, menu words take more space on the screen. Examples of menu-word interfaces include non-Macintosh Multiplan, Lotus 1-2-3, and Symphony.

Menus

With menu-driven interfaces, you are presented with a complete menu from which to make your selection. The menu takes up the full screen; with each choice a new menu appears until the selection process is complete. Menu-driven interfaces are good for the novice, since there is enough space for full instructions, but they are exasperating for experienced users because of slow operation. Examples of menu-driven interfaces include Wang word processors and many traditional accounting packages.

A New Solution: The Visual Interface

Beginning in the 1960s, computer research groups began looking for ways around the user-interface logjam, setting as key requirements ease of use, consistency, and familiarity. Xerox's Palo Alto Research Center (PARC) developed the mouse/icon/pull-down menu interface, which is the most successful yet. The interface can provide many choices without interfering with ongoing work. You select a menu category from a menu bar, and "pull down" the menu with a mouse. The work area is obscured only while you are making the menu selection. (The menu bar could have been designed to run along the screen bottom and "pop up," but this would cover up your most likely active area, the bottom of your document.)

The PARC interface has had tremendous success, and all new microcomputers are using a form of it. Apple built the PARC interface into its Lisa and Macintosh. Companies as diverse as AT&T, Atari, and Commodore are adopting the PARC interface. For the IBM microcomputers, Digital Research's GEM and Microsoft's Windows install a PARC interface purely in software. Even IBM's primitive Topview is a simplified PARC interface using text only.

The impact on microcomputers

The widespread use of the visual, or PARC, interface means that the microcomputer industry is adopting an interface standard by consensus. So when you learn to use one PARC interface machine, you have pretty much learned how to use another. You won't necessarily understand the details, but the major operations such as opening a program, editing text, and saving a file are nearly the same. Programs that use other interfaces will become increasingly harder to sell.

Nevertheless, many programs designed in an earlier era will linger on as the industry changes. A few companies have adapted their programs to run on the Macintosh exactly as they do on other microcomputers. Avoid these programs unless they meet a specific need; even then you should probably replace the program when a Mac-style equivalent appears.

Other software companies will rewrite their programs, taking advantage of the visual interface. You will use Mac's interface, but the programs will still retain their unique features and capabilities. In many cases, the Mac version will use the same file format as versions for other computers so that it can exchange information with them via modem; for others there will be a file-conversion program. Chapter 27 discusses information exchange in more detail.

Software for the Mac doesn't have to use the Mac interface, however; a programmer can write an entirely different interface or, more likely, add new elements to the present one. Since such measures can confuse users, changes should not be made lightly. There is nothing sacred about the Mac interface, but programs that modify it should have good reason for doing so. Thus far, the programs that change Mac's interface have not had reasons important enough to justify the changes.

Is Mac's visual interface the optimum one? No one really knows; the next ten years should see much more development of interfaces than the last thirty.

OTHER OPERATING SYSTEMS

Can you run another operating system on a Macintosh? In some cases, yes, but....

With a few exceptions, other common microcomputer operating systems cannot run directly on Mac. Most operating systems were designed for other CPU chips and are incompatible because each chip type uses a different set of instructions. For example, the operating systems for the Apple II, IIe, and III run on the 6502 chip; TRSDOS and CP/M-80 run on the Z80 chip; and MS-DOS/PC-DOS run on the 8088, 8086, 80186, and 80286 chips.

Many microcomputers besides Macintosh use the Motorola 68000 chip, but the other models use operating systems that generally require a different arrangement of hardware than Mac's.

Even if you can get another operating system running on a Mac, the interface will be an old-fashioned, non-visual one. This single issue makes other operating systems superfluous unless:

- You have a large existing inventory of programs you use occasionally and cannot afford to replace.
- You have written many programs in a form not transferable to the Macintosh.
- A program uses accessories not supported by Macintosh software.
- You need to work directly with files from another computer system.

UNIX

The UNIX operating system, developed for minicomputers by Bell Laboratories, has been the most widely discussed operating system for sophisticated microcomputers. Versions of UNIX can run on microcomputers with 68000 CPUs. Mac's 512 KB is a little tight for running UNIX directly, but a Mac with more memory and a hard disk drive could run UNIX.

UNIX's main features are many special tools for program development and multiuser operation, in which several users can share the same CPU. Both features are largely irrelevant to current microcomputer use. From a user's standpoint, little software is available thus far in UNIX. There is some software for computer-aided design and other specialized technical applications, as well as some fairly powerful database programs with poor interfaces. Most programs use a text-only display, for UNIX has no graphics standards.

Several companies supply software and hardware that turn the Macintosh into a terminal for a remote UNIX host computer; see Chapter 26.

UCSD p-System

Softech Microsystems developed a version of the UCSD p-System for the Macintosh. The p-System was an early effort to create a single operating system standard so that the same program could run on a variety of computers. Unfortunately, this universal approach led to sluggish performance and so many compromises that it has been a failure; the few available p-System application programs offer little to Mac users. Although Softech has withdrawn from the microcomputer market, another company may publish the p-System for the Mac.

CP/M-68K

CP/M-68K from Digital Research, Inc., a version of CP/M designed to run on the Motorola 68000 chip, is available for the Macintosh from I.Q. Software (Ft. Worth, TX). It is a traditional, cryptic, command-line operating system with virtually no application programs and few languages. All the well-known CP/M programs run under CP/M-80 and will not run directly under CP/M-68K.

Emulation: CP/M-80

Although CP/M-80 was written for the Intel 8080 and Zilog Z-80 CPU chips, the 68000 has enough power to emulate those chips and can thus run CP/M-80 programs such as WordStar and dBase II without additional hardware. Most CP/M-80 programs will run on the 8080; a few require the slightly more advanced Z-80.

There are at least three ways to run CP/M-80 programs on the Macintosh without additional hardware. Logique (Laguna Niguel, CA) offers CP/MAC, the only one that works within the Macintosh operating system and Finder. CP/MAC is a CP/M emulator; it is not licensed from Digital Research. CP/M programs appear in a window, working in their usual way, and Mac desk accessories are available at the same time. Unlike the other schemes, CP/MAC emulates the Z-80, uses Mac's standard disk format, and can run on hard disks. On a 128 KB Mac, the transient area it sets aside for programs and data is 48 KB.

I.Q. Technologies has EM80, an 8080 emulator running under its CP/M-68K; version 1.0 requires a 512 KB Mac and runs too slowly for most applications—a modest typist can get far ahead of the screen. I.Q.'s package emulates the ADM3A terminal; common operations such as deleting a line require rewriting the entire screen, and thus slow things further. I.Q.'s entire package is also fairly expensive.

Island Software (Providence, RI) supplies a licensed CP/M-80 version 2.2 in a completely standard form. This package emulates an 8080 acceptably and runs on a 128 KB Mac. The transient area is 62 KB; a RAM disk is planned for 512 KB Macs. Terminals emulated include the ADM3A, TeleVideo 950, and VT 100. The Island emulator runs significantly faster than I.Q. Technologies' package, fast enough to be usable but still slower than a normal CP/M microcomputer.

None of the CP/M systems can deal with the Apple II CP/M variants. If you need to run a lot of CP/M-80 software, you might find it simpler to buy a low-cost microcomputer, such as a Kaypro, designed for the purpose. A Mac could be connected as a terminal to a Z-80 computer that lacks its own terminal, but a Kaypro or equivalent is much easier and usually cheaper.

Emulating the Apple II and the 8088

A Macintosh might be able to run Apple II software by emulating the Apple's 6502 CPU chip, but there are some practical problems. Unlike CP/M-80, the Apple II operating systems are closely held proprietary secrets, so third-party developers find it difficult to create viable emulation software for the Mac. Also, the numerous copy-protected Apple II programs would be hard to convert from floppy to microfloppy disks (few CP/M-80 programs are copy protected).

Mac's 68000 can emulate the relatively simple 8080, Z-80, and 6502 chips without undue difficulty, but would have a much harder time emulating chips like the 8088 that perform midway between those 8-bit chips and a 68000. An 8088 emulation is possible, but the performance is likely to be poor. To run 8088 programs, a hardware accessory is more practical.

Hardware accessories: MS-DOS/PC-DOS

MacCharlie, from Dayna Communications (Salt Lake City, UT), is an 8088 computer that connects via the Mac's printer port and uses Mac's keyboard and display as a terminal. MacCharlie does not accept IBM PC-type expansion boards directly, but can connect to an external expansion chassis. MacCharlie will run text-only software designed for the IBM PC; it cannot handle graphics. To run a program, you load the MacCharlie terminal program on the Macintosh, and the IBM PC application appears in a Macintosh window. The mouse works only for Macintosh functions such as the desk accessories; it is not functional for MS-DOS programs.

The Clipboard still works, but one way only, from an IBM PC program to the Mac. Text files on disk can be sent back and forth. During normal Macintosh operation, the memory inside MacCharlie acts as a printer buffer.

MacCharlie and products of its type cost about as much as a complete, freestanding computer. Their parts cost less because they lack a keyboard and display, but these savings cannot offset the production economies of mass-produced, complete computers, especially the cheapest IBM PC clones. If you need to run MS-DOS software, a separate computer is much more flexible and, whether IBM PC or PC-clone, it will have better software and hardware compatibility. Most PC clones can display both text and graphics and accept IBM PC-style hardware accessories.

You can mimic MacCharlie's Clipboard function by sending a screen dump from the MS-DOS computer out a serial port to the Macintosh; use a desk-accessory communications program to capture it without leaving a Mac application program. You can even move IBM PC screen graphics to a Mac by connecting a video digitizer to the PC's video signal (most digitizers work with the standard IBM color/graphics display card and equivalents, not with the Hercules card or the IBM Enhanced Graphics Adapter). MacCharlie's only real advantage is to save desk space.

If you need MS-DOS operation, you may want to get a more modern 80286-based computer (IBM PC-AT and compatibles) rather than the obsolete 8088/8086 designs.

CHOOSING SOFTWARE

For all the important computer uses, there will be many competing software packages. For each software type, most packages will include the essential features. So how do you choose among them?

The choice may depend less on specific features than on how well the program meshes with your other work. Software selection between competing packages rests on such issues as personal preference, an analysis of your needs, and, increasingly important, compatibility among programs.

Do You Need the Software?

Longtime computer users can show you a shelf full of programs they bought but now ignore. Lots of ideas sound terrific but turn out to be impractical. The classic example is a calendar program. There are many calendar programs that let you enter appointments and notes and have the computer beep at you on cue. But none of them works as well as a small pocket calendar that is available at all times, not just when you are at a desk. And, unlike a computer, an alarm in your wristwatch alerts you wherever you are. For appointments, how do you enter a last-minute change in a phone booth? Some people, such as doctors, do need comprehensive scheduling software, but the common calendar programs aren't good enough for such needs.

Past experience

If your main goal is to get some work done, rather than play with the computer, try to minimize the learning period with any new software. Programs styled after another package you have already mastered become the first choice. Although most Macintosh programs use a common interface, there are many variations in details, such as command-key assignments and file compatibility. Here are some considerations to think about.

Growth. Will the software grow with your needs? For example, can you increase the complexity of your models in a spreadsheet program? Can you incorporate information produced by other users in your spreadsheet?

If a program cannot expand with your changing needs, it could be a dead end; you may have to start over with a new program. A well-designed, complex program will let you use its simpler features without fuss; you can start using the more sophisticated features when you need them.

Manuals. Is the program's manual well written and effectively organized? Does it have a decent index? Can you find information quickly? Does the manual use the same wording for operations as Apple's manuals or this book? Apple provides a writer's style guide, so anyone who writes about Macintosh software should use the same terms, such as *choosing* a menu item and *selecting* some text.

Support. Can the dealer from whom you bought the software help you adequately if you have a problem? Can a friend or business associate help out? And does it look like the software company will remain in business? For game programs, this doesn't matter; for a major database program, it is vital.

Compatibility. Will this program work with your other software? Can it transfer files to and from your other programs and other computers? With many programs you need to work over ("massage") the files before the information is ready to use.

Although software selection has traditionally rested more on personal preference than any other criterion, programs will increasingly be judged by their ability to exchange information with other programs and other computers.

COPY PROTECTION

Software developers often take special steps to make their programs impossible to copy by ordinary procedures. They do this to protect against thieves who make copies without paying for the program. Unfortunately the copy-protection process makes life more difficult for the legitimate user, who clearly should have both the security of back-ups and the freedom to use hard disks without cumbersome procedures.

Practical Issues

You may want to choose software partly on the basis of the protection scheme. Taking the user's point of view, a gallery of choices, from the best to the worst, follows:

- Not protected. This is the least constraining choice, since you can make back-ups freely.
- "Validated" copies. The program lets you make one or two working copies, including one on a hard disk. The copy is completely self-sufficient except that you can't make any more copies. The best systems let you de-validate a copy so you can move a working copy if you change hard disks, for example.
- Key information read once. This scheme requires a copy-protected "key" disk when you start the program but not again as long as the computer stays on, even if you quit and restart the program. Aside from the initial brief check for the key information, the program runs from another floppy or a hard disk. Some programs may require the key disk only once a week or every 20th time you run the program.
- Key information read every time. The program requires the key disk every time you start it, but the program itself can reside on another disk.
- Key information read randomly. A few programs ask for the key disk at random times. This is disruptive because it's unpredictable; you may have to swap disks.
- Key disk used for program loading and running. This worst-case situation forces you to run the program from the protected floppy disk. A hard disk can serve only as a data disk.

Many utility programs can copy key disks with a variety of techniques. Advanced copy-breaking schemes modify the program itself to bypass the checking routines used for copy protection, so you can run programs from a hard disk without using key disks.

Although the Macintosh does not have any special built-in copy-protection hardware, the disk-controller design permits use of many different copy-protection strategies, so there will be a protracted, never-ending battle between the designers of key disks and the authors of copy programs.

Ethical Issues

In the debates that rage over software theft, many arguments are specious. Price isn't an issue; low-cost software is probably stolen as often as the expensive variety. Although a common attitude condones stealing from large, successful companies, the smaller companies and garage shops are hurt as much and perhaps more. Several small companies and individuals distribute on the honor system, making their software freely copyable but requesting a modest payment from users. The majority of users do not pay the money requested.

Regardless of the specific legal issues, many people are unethical about copyrights, whether for audio/video recordings, printed matter, or software. If any progress is to be made in this area, no one should think that theft of intellectual property is justified just because other people do it.

If you don't like some software package because of poor features, poor support, high price, or copy protection, don't use it. Don't buy it, don't steal it; ignore it. The market drives this industry; the vendors will get the message.

EXCHANGING INFORMATION BETWEEN PROGRAMS

Macintosh programs can exchange information in five ways:

- Through the Clipboard and Scrapbook.
- As "live" or linked information between program components in an integrated package.
- With a screen dump.
- By reading document files.
- Through special links between two programs.

The Clipboard and Scrapbook

Whenever you put something in the Clipboard (stored in RAM, usually) or the Scrapbook (stored on disk), the information is saved in up to three forms:

- As a data file, using the format specific to the application.
- As an ASCII text file, without formatting.
- As a picture file, in the QuickDraw format used by most Macintosh graphics programs.

Not all programs or situations produce a Clipboard or Scrapbook in all three formats. In some cases involving large amounts of information, the program may give you a choice of how to store the Clipboard/Scrapbook.

The originating program stores multiple formats because it cannot tell where the information is going; the safest strategy is to give the receiving program a choice. When you paste something from the Clipboard or the Scrapbook, the receiving program checks the file formats and uses the most suitable format it can understand. Some programs may only understand their own Clipboard and you won't be able to paste information from another program into them; a few programs cannot read any Clipboards at all.

As examples, a text file could be converted into a pictorial image of the text and merged into a graphics file, or a picture could be treated as a block and placed in a word-processing file to be printed along with the text. Because the files were created in such different ways, in most cases you won't be able to modify the picture file with the word processor or modify the text file with the graphics program. MacWrite can use MacPaint information pasted in from the Clipboard or Scrapbook, and vice versa. MacWrite can also perform some simple changes to a MacPaint file included in text, such as moving or stretching it, but for any substantive changes, you have to go back to MacPaint.

The amount of memory available at times imposes limits on how much information you can put in the Clipboard. You may have to move the information in several blocks. The Scrapbook size is limited by available disk space.

The Clipboard remains intact when you switch disks, as long as you don't cut or copy something else into it. The Clipboard is also preserved between applications with the Switcher if you hold down the Option key while switching.

The Clipboard and Scrapbook do have a major limitation: They can transfer static or "dead" information only. If you change the source information, you will have to transfer it again.

Live Links

With a live link, changes you make in one task are automatically made in all linked documents. Suppose you create a graph and incorporate it in a memo; a live link lets you change the graph and the memo will change immediately. Live links are much easier to do in an integrated package than across separate programs, but no integrated package thus far permits live links in any direction. Jazz, for example, lets you link spreadsheet, graphics, and database information to the word processor, but not the reverse. However, the Clipboard is still available for transferring dead information, within Jazz or with other Macintosh programs. Excel can set up live links among spreadsheets.

Screen Dumps

If a program will not create a QuickDraw file for the Clipboard or Scrapbook, you can usually save a screen dump (the program's screen image) on disk by typing Shift-Command 3. This operation creates a QuickDraw graphics file, called Screen0, on the disk. A second screen-dump file will be named Screen1, and so on up to Screen9, when you must rename earlier screens before you can continue.

You can look at, modify, and print the screen image with MacPaint. Within MacPaint, you can select the image and put part or all of it into the Clipboard. For the best printed results on the LaserWriter, avoid screen dumps if the program you're using can create a QuickDraw file; it produces cleaner results.

(You can print just the top, or active, window by typing Shift-Command 4, or print the whole screen with Caps Lock-Shift-Command 4. This won't work with the LaserWriter driver version 1.0.)

Reading Document Files

While nearly all programs let you move some information with the Clipboard and Scrapbook, more complete information exchange requires that two programs be able to read the same document files.

Most Macintosh programs store files in a document format unique to the program. Document files come in many formats that have not been standardized, and no single standard format is likely or possible. Some formats are shared by several different programs and some programs can generate multiple document formats.

Special document files include formatted text files, such as those created by MacWrite or Microsoft Word for storing text with formatting information—the typeface and size, margins, and spacing, for example. The two programs do not create compatible document files, although Word can read MacWrite files, and page make-up programs such as PageMaker can read files from either word processor.

Other program types create their own specialized document files. Spreadsheet programs use their own unique formats for the formulas and interrelations between the cells in a spreadsheet. Structured data programs, such as Microsoft Chart, store graphics in their own special formats, again usually incompatible with other graphics programs, even if they incorporate some elements of QuickDraw.

Whether a program uses a particular file format is an important consideration in selecting Macintosh software. All the programs that you use regularly should be able to exchange suitable data. Document files can be transferred at several levels:

- Most desirable: Two programs can read each other's files directly.
- Next most desirable: Two programs use a common special file format, such as SYLK, that preserves all formulas and relationships.
- Better than nothing: Two programs use a common special file format that preserves current values, or all text-formatting information.

Unless specifically promised as an application-program feature, you should not expect any program to read another program's disk files. Common file formats only make sense for some classes of programs, however: A collection of accounting programs might read a common file format; a word processor and a graphics program are unlikely to share files.

Special Links for Information Transfer

Where complete file compatibility does not make sense, two programs might have special provisions to pass along specific information. For example, when you construct a spreadsheet with Multiplan, you can copy values you want plotted to the Clipboard. Then, with Microsoft Chart, you can paste these values into the graph data.

Chart lets you choose whether you want to paste data just once, or whether you want to establish a link between files. With a link, you can change the Multiplan values and Chart will automatically look at the same spreadsheet locations for new values to plot. Similarly, Mega-Merge creates mailing-list files that can be merged with MacWrite documents.

A successful program will attract other programs that can use its data, although complete file compatiblity is often difficult without co-operation from the developers. The most complex programs may create fragile data files, files that cannot be easily modified by another program.

THE FILE STRUCTURE

The operating system keeps track of all the file names and disk names. Usually you select a disk by opening a disk window or by clicking the disk button in a mini-finder, but you can also specify a disk by putting a colon between the disk name and the file name. Typing Disk2:Myfile in a Save as... dialog box will save Myfile on Disk2.

Files on the Macintosh may contain up to three major sections. All files—programs and documents—have a header and a resource "fork" or a data fork or both. A fork is merely a portion of the file.

The Header

The header is physically stored in the disk-directory tracks, and it records basic information such as the file name, date, and size. When a disk is put into Mac, the header is read into the disk-directory buffer; that's why you can still open a disk window when the disk is ejected. The header specifies whether the file is an application (that is, a program) or a document; it also holds the name of the program that created the file. Thus, if you open a document file, the operating system identifies the creator and looks for that program, first on the same disk and then on other active disks. Programs are considered their own creators.

The header also identifies the general data type: text, picture, and so on. This lets programs other than the creator look at a document file. When you ask to open a file in a word-processing program, for example, the program asks the operating system for all suitable document files. It can ask for any text file, or it can be restrictive, asking only for files it created, or even only for files created on a certain day.

The header does not store the file icon; all file icons are stored in the invisible Desktop file, which assigns icons according to the file's type and creator. The Desktop file is updated along with the header whenever changes are made to a disk. The header also contains pointers to the physical locations of the resource and data forks, which the system treats as two separate files stored under the same name.

Apple has a utility program called the Resource Editor, which can change header information. With it you can make a program read another program's file, change the icon assignments, and do other housekeeping chores.

When Apple changes to its planned hierarchical file structure, the operating system may read header information into RAM from a specific set of files instead of from the entire disk. This would help owners of hard disk drives, which can store hundreds of files; there is no point in constantly searching among files that you are not using.

The Forks

The resource fork holds everything that is not conventional data. An application-program file usually consists of a header and a resource fork, without a data fork. Resource is a very general concept that includes the actual program code, dialog boxes, alert boxes, menus, fonts, icons (for files to be created), and just about anything else that is not a document, although some user data ends up in the resource fork.

The data fork consists mostly of document information. A pure text file is a data-fork-only file, but many documents are hybrids because formatting information may go into the resource fork. The distinctions are not absolute; sometimes all the document information is in the resource fork.

WHY THE MACINTOSH IS SO SLOW

Much of the time you work with the Macintosh, you can't do anything except stare at the wristwatch icon on the screen. I discussed the disk-controller speed problems in Chapter 14. Software creates further speed problems.

The Macintosh performs many more functions than the traditional microcomputer; when you start Mac, the system must load not only the Finder but other essential information, such as the system fonts and icon shapes. The Finder often recreates the Desktop file for all disks currently known to the system. Older microcomputers don't have equivalents of these functions; they use a single built-in font for everything and use few or no graphics. A visual interface will always take longer to load and draw than a text-only interface because icons and windows contain more detail.

But the system also does operations you don't want it to do. It frequently checks disks that are not being used for anything. The process is particularly aggravating when the Finder insists on your inserting a disk you last had in the Mac some time ago and one to which you are making no changes. In this respect, the system has moved beyond user-friendly to user-suffocating.

Many utility programs speed up specific steps. Fast Finder (from Tardis Software) does away with the visual interface in the Finder altogether and replaces it with an old-fashioned, command-line interface. You will have to type in commands, but the skilled user will find the Tardis product faster than the standard Finder in some situations.

Apple has also modified its original Finder to increase operational speed, taking shortcuts that avoid changing or recreating the Desktop file whenever possible. Portions of the Finder and system fonts once found only on disk have been moved into the new Macintosh ROM planned for early 1986.

Hard disk drives also improve speed considerably, because they are inherently faster than floppy disk drives and because you don't have to swap disks constantly.

MISCELLANY

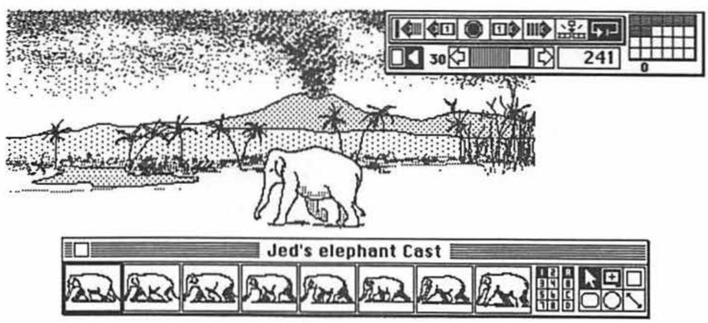
Game software is really outside the scope of this book, but I can't resist commenting on Sargon III (Hayden Software). It is an excellent chess game; you can set different levels of play, ask for a suggestion or swap sides with the computer if you are losing, and set up and play end games. Sargon III might not challenge the most accomplished chess players, but it is quite capable of beating the rest of us.

On all present computers, you must know how to type to get useful results from most programs. Two typing programs that work well are Typing Tutor III (Kriya Systems/Simon and Schuster) and Typing Intrigue (Macware from Forethought). Both incorporate a game; Typing Tutor has a shooting gallery activated by letters, Typing Intrigue has a mystery to solve.

Three worthy programs let you process music: MusicWorks (MicroMind/Hayden) is a single program that lets you compose and edit music on the screen, change the key signature and tempo, and listen to the results. In addition to tunes supplied with the program, many others are available on bulletin boards and public-domain software. ConcertWare (Great Wave Software, Stanford, CA) has three separate programs for creating and editing music, designing an instrumental sound, and playing the music. As separate programs, ConcertWare is a little harder to use than MusicWorks, but each component is more sophisticated; ConcertWare can play longer pieces and the sound quality is better. Neither program uses full music notation, but Professional Composer (Mark of the Unicorn) does include further features for the serious musician.

All the music-program authors plan to add support for the MIDI musical-instrument interface (see Chapter 24).

VideoWorks (also from MacroMind via Hayden) is an excellent animation program. You can build up frames one by one, and then play them back as an animated movie. Like MusicWorks, VideoWorks has an exceptionally well thought-out interface.



VideoWorks (drawings by Jed Schwartz)

Chapter 20 Problem-Solving and Getting Help

Your Macintosh will need only minimal maintenance and probably few, if any, repairs. If you encounter a problem that you suspect may be a hardware or software failure, try to reproduce the problem on another Macintosh. If you can reproduce the problem, it is either in the software or in the disk itself; try a different disk (not a copy of the suspect disk). If the problem persists, you probably have a software bug; if not, the first disk was probably faulty.

If you cannot reproduce the problem on another Macintosh, you may have a hardware problem. I will discuss some of the simplest and most common hardware problems; for more information about these and more serious problems, check your manuals, consult your dealer, or contact a user group.

FAULTY CABLES

For any computer, the most failure-prone parts are the plugs and connectors. If you are having problems with any external component, try swapping cables first. Look at the plugs carefully. Are any pins bent? Some pins may be missing; this is normal, but compare with another unit if you have any doubt.

Before you swap cables, turn off the power to Mac and the accessory equipment. If you have problems when you are in the middle of important work, you may not want to turn off the power and thereby erase all the data you have in RAM. It is safe to unplug and check the cables to the mouse and the keyboard without turning off the power;

serial-port cables can usually be checked, but follow the manufacturer's recommendation if in doubt. You should always turn off the power before unplugging an external disk drive.

GLITCHES

All computers are subject to glitches—minor problems that cause temporary setbacks but are not necessarily symptoms of disease. Often the best cure for a glitch is to turn off the computer and peripherals and go away for awhile; the problem may have disappeared by the time you start work again. If you turn the power off and on quickly, however, the power may not be off long enough for some equipment to go through a power-on procedure (the capacitors in the power supply continue supplying power momentarily). Turn the equipment off for several seconds before turning the power on again.

Static electricity can be an occasional problem. Many companies sell anti-static pads to put under your chair or to stand on, or pieces of metal you're supposed to touch before operating the computer. These products are effective only if grounded properly. They aren't usually necessary unless the air is so dry that you are personally uncomfortable and you set off sparks every time you touch a doorknob.

Any computer will occasionally seize up so that you can't get anything to work. With Mac, this kind of glitch can be disconcerting because the mouse pointer may still work; it uses a separate portion of memory from the programs. Sometimes you won't be able to pin down the exact cause. If it is a software bug, the same thing should happen if you repeat each step exactly, but you won't always remember the steps exactly; something you did hours earlier could be the cause. Otherwise, it might be the power line or static electricity; or a bit of dust can make a disk temporarily unreadable. If the disk-drive head has gone on to read another part of the disk, it may be able to come back and read a track that was faulty minutes earlier.

Some problems simply have no explanation. You may sometimes run into a problem that does not repeat, and you will never know why it occurred. The semiconductor memory chips used in all computers are susceptible to errors resulting from bit changes induced by cosmic rays, though such errors are rare—much less than one error per year. Rather than spend hours trying to figure out a problem that you cannot get Mac to repeat, accept that these things happen once in awhile, even with the largest, most expensive computers.

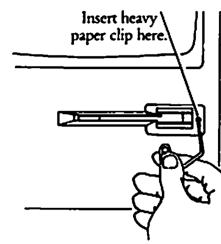
If some programs lock up frequently, install the INTERRUPT RESET buttons on the left grille; see the Mac manual. Press RESET to restart the computer. The computer will restart as if you had just turned it on. The switch is essential for programmers, since unfinished programs often behave badly and lock up the computer. If you install this switch, take care not to have anything push against it or you will get inadvertent resets.

See Chapter 29 for a list of common system error codes.

IF A DISK GETS STUCK

If the computer seizes up and you can't get any response, there are three ways to remove your disk (two of them will erase RAM and all of them may occasionally make the disk or parts of it unreadable):

- First try pressing Command-Shift-1 or 2 to force the external or internal disk to eject.
- Turn off the power, press and hold down the mouse button, and turn the power back on. The disk drive will eject the disk in a moment.
- If the power has failed, you can remove the disk mechanically by inserting a heavy pin (a heavy paper clip bent open) into the small hole just to the right of the disk-drive slot.



Removing a disk using a heavy paper clip (if the power has failed).

POWER-LINE PROBLEMS

Many companies sell power-line conditioners, costing $15 to $100, that eliminate short, sharp voltage pulses or the high-frequency interference that can induce errors in a computer (they rarely cause lasting physical damage). Whether you need a power-line conditioner depends on the quality of your power source and on whatever else is connected to it. If your Mac locks up or produces errors whenever an appliance is turned on in the building, try a power-line conditioner.

If you have a choice, do not plug your Mac into an electrical circuit that is used for any heavy equipment, such as air conditioners, refrigerators, power tools, copy machines, or vacuum cleaners. The motors in these devices can generate a transient power-line spike that occasionally causes trouble. Vacuum cleaners are especially insidious, since they move around, unlike refrigerators. If you see a vacuum cleaner coming your way, save your file to disk. Eject and restart the disk, then continue working; you probably won't have a problem, but it's best not to take chances. Once you're sure a potential hazard creates no problems, you can ignore it.

Brown-outs

A Macintosh is fairly resistant to brown-outs, or low-voltage periods. Although the announced specifications call for the Macintosh to operate on 105 to 130 volts, it actually runs satisfactorily on as little as 95 volts (some units even run at 75 volts). If you need a little more latitude, the international version has an even more tolerant power supply, specified for 90 to 130 volts in the low range, switchable to 180 to 260 volts in the high range. This version will actually run on anywhere from 85 to 135 volts (low) and 170 to 270 volts (high).

The power-line frequency doesn't matter; Mac's specifications call for 47 to 64 Hz, but the power supply will actually run on anything from 30 Hz to 20 kHz. The leakage current increases with higher power-line frequencies, so make sure your computer is properly grounded. Hard disk drives may not be as tolerant as the computer. The built-in clock/calendar uses its own quartz oscillator and does not depend on the power-line frequency.

Severe Electrical Storms

In case of a severe storm, unplug the computer and disconnect the modem from the telephone line.

Power Failures

Power failures are another threat. Whenever the power fails, everything in Mac's RAM disappears, and any open file remaining on disk may be lost as well. There are several steps you can take to combat this threat; three involve the purchase of more accessories, and the fourth involves the development of cautious work habits.

Uninterruptible power supplies

You can buy an uninterruptible power supply (UPS). These come in two forms, both based on a battery, a battery charger, and an inverter (which converts the DC power from the battery to AC). A true UPS always runs off a battery that is continuously recharged, so the power stays constant regardless of what happens to the AC power supply. A true UPS system is also fairly expensive, for it must have well-designed circuitry to regulate charging and discharging without damaging the battery.

With a cheaper switcher UPS, the computer operates directly from the AC supply under normal conditions. If the AC power fails, the unit switches to the battery so quickly—generally in less than 15 milliseconds—that the computer does not notice the change. (The capacitors in the Mac's power supply can keep the logic circuitry alive for up to about 100 milliseconds).

When the power fails, you must be present; the battery usually stores sufficient power for 15 to 20 minutes' work, enough time for you to close all your files and turn off the machine gracefully. Some UPS units have much larger batteries, but for Mac, the UPS should be capable of supplying 150 watts of power (more if you have a hard disk). A UPS normally includes power-line conditioning as well. If you use a UPS, you needn't plug the printer into it; just protect the critical components—the Mac and any hard disk drives.

Permanent battery operation

Continuous battery operation may be wise if you are working in an area with exceptionally unreliable power or if you have no conventional power source. When you aren't using the computer, the battery is charged from an AC power supply or generator. An inverter with a square-wave power output works satisfactorily (but in rare cases an isolation transformer may be necessary to smooth the square-wave slightly). Because of battery maintenance problems, you probably should not use a battery unless you have to.

Recovering disk files

A few programs write the information in RAM to disk whenever you haven't entered anything from the keyboard for some time. Automatic disk writing is a powerful feature if the information written to disk can be recovered after a power failure.

You can also buy a sector-reader program, a program that can read individual sectors on the disk, even if the file is unreadable by normal means. Sector readers often require some skill to use and cannot always recover a file. MacZap (Micro Analyst, Austin, TX) has a complete set of tools for sector reading and file recovery.

Cautious work habits

One of the most undramatic and yet powerful ways to ward off the threat of power failures is to make a habit of safeguarding your work. The following are two habits to get into.

Saving on disk frequently. You should save your work on a disk (from RAM) frequently. If there is a power failure, you will generally only lose whatever work you've done since the last save. Be prudent and save after every complex operation, especially when moving information between programs. *Caution:* Occasionally it is not enough to choose Save from the File menu in some programs, since the disk buffer (which contains the directory and other such essential information) is not always written to disk. Few software manuals tell you exactly what the disk-writing routines do.

If the program itself is faulty and unable to print your document, you can sometimes load the program and the document file, then save everything you can to the Scrapbook; then at least you have a record.

Printing back-ups. For some types of work, a printed copy can act as a back-up. Although not ideal, since you will have to retype everything, at least this mechanical procedure is simpler than having to rethink it all.

DISK BACK-UPS

You should make copies of important data regularly. Programs should be backed up at least once, but beyond that there is little point in backing them up again. For most people, the document files are much more valuable than the programs, which can be quickly replaced. For information on hard-disk back-ups, see Chapter 14.

MAINTENANCE

Although the Macintosh is not like an automobile that requires periodic tune-ups to stay in shape, the following are some tips on maintenance.

Routine Cleaning

If you don't use a dustcover, dust will eventually build up inside Mac. Modest amounts of dust won't affect the computer. With or without a dustcover, Mac's exterior will probably need occasional cleaning. Dust with a damp cloth; use sparing amounts of a mild, nonabrasive household cleaner if necessary, but take care not to get anything dripping wet. Clean the screen with a glass cleaner; again, avoid drips (for optical-glass anti-glare screens, use lens cleaners only). See Chapter 13 for information on cleaning a mouse.

There is little agreement about cleaning disk-drive heads. Many users of microcomputers have never cleaned their disk-drive heads in years of heavy use and have had no problems. Others claim that regular cleaning is essential. Certainly you should not have to clean the heads more than every few months under normal conditions.

Long-term Maintenance

With long-term use, the Macintosh's mechanical parts may require service and alignment. The disk drives and the printer may need such work after several years. An aging CRT will begin taking longer to start up and the screen will get a little dimmer. The rest of the computer should never require service unless a part fails.

Mac's internal design makes it very difficult for casual tinkerers to repair. Hobbyists should note that, of the integrated circuits, all but the ROM chip are soldered in. The boards have four-layer circuits; the chance of damage is very high when unsoldering components with an ordinary soldering iron.

Service Contracts

Vendors like service contracts because they are a high-profit item. Service contracts rarely make sense, though; micros are relatively cheap and do not break down often. If your company has multiple micros, you should buy a spare unit instead of multiple service contracts. If something breaks, you just swap computers. But the spare computer almost always winds up in constant use, so you have to get another spare. . . .

PHYSICAL PROTECTION

Aside from a dustcover, a Macintosh in normal usage needs little if any protection from the elements. Water from a sprinkler system or foams from fire extinguishers will damage any electronic equipment. The fire-fighting gases Halon 1301 or Halon 1211 can put out fires without such damage.

If theft is a problem, several companies make locking devices for the major components.

GETTING HELP

Almost every computer user needs help at one time or another. Whether you're trying to find out what you did to make that valuable document disappear or how to connect some esoteric accessory, there are resources that may help you.

Manuals

Your first source of information should be the manuals that come with Mac or with the piece of hardware or software that's giving you the problem. Apple's manuals are among the best in the microcomputer business and bear careful reading.

But manuals in general have never been a strong point of the microcomputer industry; good ones are so rare that a case can be made for selecting software on the basis of the manual's quality alone. Generally, a company that puts the effort into a good manual is also a company that cares about its customers.

Hotlines

Many microcomputer companies do not have a hotline and insist that you direct queries to the dealer; they will accept technical questions only from dealers. Your questions will often be misunderstood in the process. The best established companies maintain hotlines that are manned by trained staff who are able to answer technical questions. Hotlines usually work out better for the user. Where possible, buy only products supported by a hotline.

Hotlines are expensive to staff and maintain. If you use a hotline, you do have some responsibilities. You should have made a reasonable effort to find the answer to your question in the manual, and you should expect information only about the products produced by the company you are calling. If you are calling because a program failed ("crashed"), be specific about exactly what happened, any error messages you might have seen, and any action you might have taken that could have caused the problem. If you are using any nonstandard component—hardware or software—note that as well.

For some products, the company may charge for hotline access; such charges may be fair. See whether competing products have such charges and compare carefully.

Computer Stores

Of course, you can always go back to your dealer; however, dealer competence varies very widely. Some dealers are helpful far beyond the call of duty; others are unable to answer even the simplest, most straightforward questions. Generally, salespeople should be able to answer simple queries and check the operation of your unit against others. Beyond that, it's hard to tell you what to expect.

If possible, ask your friends where they bought their Mac, accessories, or software, and whether they are satisfied with dealer support. If your friends are pleased, then you should buy from that dealer also. Avoid dealers with a poor reputation for service, even if the price is a little lower. Computers are complex enough that most people will need help from their dealer at some time.

As with any other fast-growing industry, the microcomputer industry has attracted fly-by-night operators. Check with the local Better Business Bureau or consumer groups if you have doubts about a dealer. Always be cautious if you are asked to pay in advance for any item not immediately available. A small cash deposit may be reasonable for an unusual item, but you should never have to put a deposit on anything a store regularly carries.

A salesperson unusually determined to sell you a particular product may be getting a "spiff," a direct sales commission from the manufacturer, above and beyond the normal dealer discount. Spiffs distort the sales process but unfortunately are common.

Mail-order Companies

Computer magazines are full of ads from mail-order houses, generally promising quick service at low prices. They can be a good deal or a disaster. Many mail-order companies accept your order, charge your credit card, and do nothing for as long as possible. The honorable ones ship your order when promised and submit the charge only when they actually ship the product. It's hard to tell the two apart; word-of-mouth helps, but a company's quality may change over time.

Most mail-order houses offer little or no support for the products they sell. Once you've bought something, they don't want to hear from you until you place another order. By eliminating support, they can sell for less than normal dealers.

There are a few good mail-order houses that offer excellent support by telephone—much better, in fact, than most dealers. If you find one, you're lucky. Tell your friends (but don't be surprised if the quality of support withers as the company grows).

Whatever you do, don't use the resources of a local dealer for information and then place the order with a mail-order house. It isn't fair, and discourages the competent stores who deserve a return on their efforts. A dealer offering useful information will also provide useful service after the sale.

Publications

Magazines and newsletters started quickly for Mac. The quality of information in publications varies widely like everything else. The computer press grew so rapidly that desperate publishers hired many naive writers and editors. The best magazines are staffed by technically sophisticated journalists; they bring you the latest information and perhaps a little hype.

The weaker magazines can be identified by their style: articles that consist mostly of quotes from computer-store sales personnel—genial, uncritical, and uninformative. A few magazines have gone so far as to ban all criticism of products. Paradoxically, some of the worst magazines make the fewest errors because they offer so little information in the first place: The more information in a publication, the greater the chance of error.

The quality of computer books varies widely also. Books inevitably take longer to produce than magazines, so you shouldn't look to books for the latest information. Instead, the best books offer insight or provide a reference. They anticipate questions and problems, and if they don't always solve a problem, they can give you the tools you need to find your own solution.

There are dozens of books on Mac, many quickly written and superficial. The introductory-level books are less useful for Mac than they are for other computers, because they don't tell you much more than you'll find in Apple's excellent manuals.

User Groups

Throughout this chapter, I've stressed the value of word-of-mouth recommendations. You can greatly expand your circle of Macintosh-user acquaintances by joining a user group. Macintosh magazines carry lists of such groups so you can locate one in your area—or you and your friends can start your own.

A good user group is probably the best single source of information about computers. Club members run the gamut from rank beginners to computer engineers, and everyone shares information. If you can't get a direct answer to a question, you will probably find someone with a similar problem, and together you can find the answer.

Often the most sophisticated users are the "hackers," people who spend every spare moment bent over their computers delving into some obscure hardware or software feature or bug. Such enthusiasts are like hot-rodders—they may be a good source of information about repairing your carburetor, but their ideas of how to choose and use cars may have little in common with your needs. Yet the best hackers will understand the computer's innards far more thoroughly than any dealer and they can be an invaluable information source.

If you are thinking about buying a Mac, you may get advice from owners of competing micros. Many people who spend several thousand dollars on a computer system develop an emotional attachment to their investment and lose all their objectivity. If you buy a Mac, you may fall prey to the same syndrome. Keep in mind that there is no perfect microcomputer; all designs are compromises. Most major current models do at least a passable job for average applications. Every computer, including yours (whatever model), will be obsolete some day.

Electronic Bulletin Boards

Electronic bulletin boards can be another source of information. These bulletin boards are stored inside a computer and you gain access to them through telephone lines, with a modem. The simpler ones run by user groups generally don't charge for the service; the more complex Macintosh bulletin boards on the national electronic database services run by CompuServe and Delphi cost $5 to $20 per hour to use. Bulletin boards can be effective for short, clearly defined questions as well as for tips on many subjects.

Electronic bulletin boards have generally been designed for hobbyists, and all of them, including the commercial services, work badly. The Macintosh offers the possibility of an elegant and simple interface between you and the messages, but development and acceptance of such an interface will take time.

The Macintosh versus the IBM PC-AT — Chapter 21

At the beginning of this book is a statement that the views expressed here are mine and mine alone. That statement applies as much to this chapter as to any other. I do not speak for Microsoft, Apple, or any other company.

Comparing a Macintosh to the PC-AT isn't exactly fair, since the PC-AT costs considerably more. Forthcoming products—a lower-cost derivative of the PC-AT using an 80286 CPU and an enhanced Macintosh with some expansion capability—will make a more suitable comparison, but they weren't announced when I wrote this. (The successful but now obsolete IBM PC and PC-XT lack the computing horsepower to run modern graphics software effectively.) In this comparison, I've included some historical notes and comments on future developments.

THE PRODUCTS

Apple was the leading personal-computer maker when IBM introduced its PC in 1981. By the end of 1983, IBM had taken the lead, riding on a flood of software and attracting dozens of imitation PCs from other manufacturers. Probably no one, not even IBM, anticipated such success.

In 1984, Apple introduced the Macintosh and IBM introduced the PC-AT. Both are solid, effective microcomputers, yet they follow virtually opposite design philosophies.

The Macintosh is built around an elegant software concept, the visual interface, that is carried through all its software. The best Macintosh software, some of it written by Apple, is the best for any computer, regardless of size. But the Macintosh hardware is all too often frustrating. Many hardware details are well designed, even brilliant, but the Macintosh was seriously underconceived for the business and professional role it must now play. The Mac's lack of memory, slow floppy disk drives, and inability to handle hard disks gracefully irritate the serious user every day.

IBM, on the other hand, follows the long-established tradition of general-purpose computers: An IBM computer is simply a hardware base upon which software developers build application programs and a user interface. The PC-AT builds on the original IBM PC, a technologically lackluster 1981 design that succeeded mainly through good timing and marketing. The PC-AT can be made to work—at least the hardware does not place major barriers in your way. But software for the AT is another story. IBM itself has never developed great software for any size computer in its entire history. All the usable software for the IBM PC series has come not from IBM, but from independent companies, all of whom went off in their own direction. As a result, it's hard to transfer even simple text information between programs; transferring graphics is hopeless.

Both companies have major blind spots: Apple thinks it conceives great hardware, IBM thinks it can write great software. In time, both will start adopting the best features of its competitors: Apple hardware will become more flexible, PC-AT software will slowly be given a friendlier interface.

CENTRAL PROCESSING CHIP

IBM PC-AT: Intel 80286, a 16-bit chip running at 6 MHz.

Macintosh: Motorola 68000, a 16/32-bit chip running at 7.78 MHz.

The two processors are comparable in speed and power. From a programmer's standpoint, the 68000 is a better design, but these differences don't matter to the user. MS-DOS programs are usually much simpler for the computer to run, so the PC-AT is often faster for routine tasks. Macintosh software is much more complicated and Mac's limited memory and slow disk drives reduce overall efficiency. But as more sophisticated programs begin appearing for the PC-AT, its effective speed will slow down, too.

In many practical applications, disk-drive access times effectively limit the operational speed. Magazine articles often include speed comparisons, but these tests usually measure only one or two functions that are mainly interesting to programmers and don't really address overall operational speed.

RAM

IBM PC-AT: 15+ megabytes, although the average system probably has 512 KB; memory expansion easy and cheap.

Macintosh: 512 KB; memory expansion possible but difficult.

MS-DOS/PC-DOS (version 3.1) can use a maximum 640 KB of RAM for programs and data, but future versions of the operating system will use more memory; with present software, additional memory can be used for data or as a RAM disk. The UNIX or XENIX operating system can already use several megabytes of memory.

Mac's 512 KB memory limit is one of its most serious drawbacks. However, one-megabit RAM chips will increase the memory to 2 MB. With currently available 256-kilobit chips, memory expansion to 2 or 4 MB is possible, but requires modifications to the computer. Because of a shortsighted design limitation, the initial Mac design cannot address more than 4 megabytes of RAM; a fix is straightforward but causes unnecessary changes.

Neither Apple nor IBM has understood the importance of large RAM; the original Macintosh had only 128 KB and IBM announced a version of its PC with a mere 16 KB. IBM, however, makes RAM expansion easy. The next Macintosh model must have provision for at least 4 megabytes of RAM, preferably 8 or more. Contrary to conventional wisdom, more RAM makes operation easier for a novice user; an experienced user can deal more readily with limited memory.

INTERNAL EXPANSION

IBM PC-AT: Four expansion slots available.

Macintosh: No provision for internal expansion.

Of the eight expansion slots on a PC-AT, four are taken up by the display adapter, disk-drive controller, ports, and additional memory. New, lower-cost IBM computers will probably have fewer slots. The Macintosh does not provide for internal expansion except through modifications by independent companies; a future Macintosh will at least have expansion capacity for memory and fast, hard disk drives.

DISKS

IBM PC-AT: 5¼-inch, 1.2 MB floppy disks; 20 MB hard disk.

Macintosh: 3½-inch, 400 KB and later 800 KB microfloppy disks; 10 MB hard disk common.

IBM designed and built the first commercial floppy disk drives and the first small hard disk drives. For its personal computers, IBM uses off-the-shelf, industry-standard disk drives, readily available at low prices. The hard disks for the AT are high-performance linear-motor designs and communicate at high speed through the system bus.

Apple's clever, low-cost floppy disk controller design was fine for the Apple II, but is less suited to a Mac because of slow performance; it takes up excessive CPU time. Further, Apple has consistently had problems with disk drives. The original Macintosh prototypes used the unusual four-opening "Twiggy" disk drives with 750 KB of storage capacity, which were also used in early production Lisas; when those drives were canceled, Apple changed to a custom version of the Sony microfloppy drive. The single-drive version of the Macintosh is nearly unworkable, and most users have to buy an extra box with a second drive. Apple charges a premium for those drives; even the few alternate sources charge a high price.

Double-sided microfloppy drives with standard electrical interfaces were already in volume production by several manufacturers in late 1984, but Apple could not use them because of its custom design requirements. Apple chose new, redesigned track-seeking logic in the double-sided drives, which requires support from a new ROM, which led to further delays.

Two of the new, thinner, double-sided drives can and should be installed in a Macintosh case, but some Apple planners have argued—surprisingly—for staying with a single drive in the main case.

Hard disks are a severe problem on the Macintosh. Bus-connected hard disks, the only good configuration, require modifying the Mac and thus raise warranty problems. Port-connected drives work but not particularly well. Apple's hard disk connects through the disk port and is slow compared to an IBM PC-AT; disk and file servers on AppleTalk are even slower.

SCREEN GRAPHICS

IBM PC-AT: Low resolution, 640 by 350 pixels in color.

Macintosh: Low resolution, 512 by 342 pixels, monochrome.

With the Enhanced Graphics Adapter, IBM finally can display acceptable screen graphics on the PC-AT. But without a standard graphics format, few programs can share graphics information. Standardization will take considerable time in the unstandardized world of MS-DOS software.

Mac's standardized graphics are a major advantage over the PC-AT. Because the graphics capability is built into the computer, nearly all Mac programs use the same graphics routine and share information easily. The display quality is unusually good, but color awaits future developments.

For cost reasons, both the AT and the Mac have low-resolution displays. Medium-resolution displays (about 700 by 500 pixels) are somewhat expensive for mass-market microcomputers; high-resolution displays (1500 by 1000 pixels or more) are very expensive. ("Resolution" on a CRT customarily refers to the total pixel count rather than the pixel density.)

KEYBOARD

IBM PC-AT: Needs improvement.

Macintosh: Needs improvement.

The IBM Selectric typewriter had an outstanding keyboard; the IBM PC (for many users) had a strikingly bad one. The PC-AT fixed some problems and introduced others, misplacing the Escape and Backspace keys. The mechanical feel of the PC keyboards is fairly good, but the keys hit bottom too hard for most rapid typists.

The Macintosh keyboard doesn't have enough keys. In pursuit of an interface ideal, the Mac keyboard leaves out both function and cursor keys; in the real world, such keys help serious users greatly. The optional keypad can serve (a little awkwardly) for function keys. It has cursor keys, but they are poorly laid out. Most software does not make use of the keypad. The mechanical feel is poorer than IBM's; the keys hit bottom even harder.

SPACE REQUIREMENTS AND PORTABILITY

IBM PC-AT: 501 square inches of desk space; very awkward to carry.

Macintosh: 207 square inches plus space for the mouse; fairly easy to carry.

The PC-AT comes in three bulky pieces; one person alone cannot carry a complete system. The computer will not fit in an airline cabin. If optional software uses a mouse, then the desk-space requirements increase correspondingly.

Mac is among the smallest desktop microcomputers; it comes in four parts, including two small ones (the mouse and a second disk drive), so it needs a carrying case, but one person can easily carry a complete system. But a second floppy drive and a hard disk are additional boxes, unless they are installed internally. Taking a Mac on board an airliner is possible but usually awkward; it will fit only a few overhead bins.

USER INTERFACE

IBM PC-AT: None; interface determined by the software.

Macintosh: Visual interface with mouse and icons.

The PC-AT by itself offers no interface other than the hardware for a screen display and a keyboard. However, two visual interfaces are already delivered or announced. Digital Research's GEM is a literal imitation of the Macintosh interface; Microsoft's Windows is similar but uses adjacent rather than overlapping windows. IBM's Topview is much less sophisticated, omitting many key features including graphics. None of these interfaces will be as useful as Mac's until they have a critical mass of supporting (not just compatible) application software. Only with such software will the interface work smoothly and information transfer become practical. The three similar but incompatible interfaces illustrate both the profusion and the problems of MS-DOS software.

The Macintosh's most valuable feature is its built-in visual interface; programmers would have to do considerable extra work to replace it, and the results would not work like other Mac software.

SOFTWARE BASE

IBM PC-AT: Large, chaotic software base needs redevelopment.

Macintosh: Modest, growing software base.

In the early 1980s, hundreds of new software companies formed to develop thousands of programs for the original IBM PC. As many as 200 word-processing programs were written. 1984 was probably the high-water mark in numeric terms; time and financial realities have eliminated most of the start-ups. Investment money will probably never flow as freely in the future, and we may never again see so many word processors for a single computer model. Although the overall market will continue to grow, it will be a market dominated by fewer and fewer companies.

Neither the IBM PC-AT nor the Macintosh will ever have 200 word-processing programs written for them, but no one will care. The half dozen successful word processors will be better than any of the other 194. And the variety of software will actually increase; a larger proportion of programs will target specific niche markets.

(Some people may object to my numbers here, saying that you can buy those 200 word processors and run them on the AT. Actually, you can't; most of those programs are no longer available and the developers have disappeared.)

LEARNING TIME

IBM PC-AT: 10 to 30 hours.

Macintosh: 2 to 4 hours.

A key Macintosh advantage is its ease of use, both during initial learning and when starting work with a new software package. However, when the PC-AT adopts a visual interface, learning time for this computer will start dropping.

AVAILABILITY OF INFORMATION

IBM PC-AT: Excellent for hardware; fair to good for software.

Macintosh: Poor for hardware; good for software.

IBM has published technical specifications for its PC hardware, and interfacing information for MS-DOS, allowing many companies to build machines that are close copies. The copies have made the IBM PC and now the PC-AT industry standards.

Apple is less generous with its information and has taken legal action against companies that made copies of the Apple II. At least in relative terms, Apple's investment in the Macintosh is larger than IBM's in the PC, so Apple is understandably reluctant to share the details. Nevertheless, licensed versions of the Macintosh would help make it an industry standard.

Both companies supply needed information for software developers. IBM provides thorough documentation for the PC-AT; Apple gives comprehensive documentation with explanations for the Macintosh interface.

CORPORATE STABILITY

IBM: A corporate colossus with a foot in most doors; widespread sales force in place for large computers.

Apple: A large company equipped mainly for retail sales.

Both companies are in the personal-computer market for the long haul. Both have well-developed dealer and service networks. Their customers should be reminded occasionally that the main goal for both companies is to make money.

PRODUCT-LINE INTEGRATION

IBM: Essentially no integration.

Apple: Same.

IBM products in a specific category should work together, but often don't. Designed independently, most IBM office-automation products have little in common. For example, two IBM products with nearly the same design concept, the Displaywriter and the PC, work completely differently and don't even use the same character codes. However, the PC-AT does maintain substantial software and some hardware compatibility with the PC. But compatibility over the next few years will be rocky as the interface and graphics standards change. IBM has moved toward connecting its microcomputers with its larger machines as terminals. Topview contains some internal functions for this purpose. It's the wrong direction, since mainframe software nearly always has poor user interfaces. Instead, mainframe computers should be information servers to micros. The visual interface on the micro can then hide the workings of the mainframe.

The two Apple families—Macintosh and Apple II—have little in common. Macintosh products for the next few years should be largely compatible with the present model.

OTHER MICROCOMPUTERS

Three micros, newly introduced in 1985, might become established designs: the AT&T UNIX PC (68010 CPU), the Atari ST (68000 CPU), and the Commodore Amiga (68000 CPU). The major test for each will be how much software support they can attract.

Macintosh Developments Chapter 22

This chapter covers several topics:

- Modifying the Macintosh.
- Future Macintosh designs.
- Finding out about new products.
- Minimizing problems with accessories.

MODIFYING THE MACINTOSH

As delivered by Apple, the Macintosh has no internal expansion slots or unused connectors; you cannot add accessory circuit boards or cards internally as you can to an Apple II or IBM PC. But the Mac can be modified in several ways. Most of the internal modifications described here will void the warranty, but some are more serious than others. An innocuous modification is unlikely to cause problems or create incompatibility, present or future, but the more drastic modifications can create unforeseen problems. All modifications must be done by a skilled technician, of course.

Bus Expansion

Such important accessories as fast hard disk drives and additional memory must be attached directly to the 68000 bus. To do this, a technician must remove the 68000 CPU, attach a bus-adapter circuit card, and plug the CPU into the card. The connector can also piggyback on the 68000 chip. The accessories can then be mounted either internally or externally.

Internal expansion is compact, but space is tight. The additional circuit boards generate heat, which usually must be controlled with a fan. You must give up Mac's normal silence. And you cannot easily change any internal expansion devices or check operation by removing the accessories—a nuisance when troubleshooting.

An external expansion chassis would require more desk space but could hold many accessory circuit boards and other items. Internally, the modification would consist of a small adapter board attached to the 68000 CPU and a connector at the back of the Mac. The connector will be a large one because the bus has many pins and should be shielded against radio-frequency interference, a problem that will probably stop anyone from bringing the bus out through the right-side ventilation slots. Whether the expansion chassis has a fan or not depends on the layout and accessories. By disconnecting the chassis, you would be able to nearly restore the original configuration.

The microcomputer industry has already developed several fairly expensive bus designs, including the VME and Multibus, for 68000 computers, but companies doing bus expansion for the Mac seem likely to go off in their own incompatible directions. No clear expansion strategy has emerged; each company makes the modification in a different way. The best situation would be for the companies to get together, agree on a standard, and certify each company meeting the requirements. Then a third-party service firm could provide warranty work and other service.

Memory

The simple expansion converts a 128 KB Mac to 512 KB, a simple, innocuous conversion that only requires swapping the RAM chips and adding a few minor components. Don't attempt the conversion yourself unless you have experience working on multilayer circuit boards. Instructions are readily available through user groups and in *Dr. Dobb's Journal*, January 1985 (note the addendum on page 4 of that publication).

Some designers can't resist trying to improve on an existing product. General Computer, when it upgrades a 128 KB Mac to 512 KB, increases the speed of the memory slightly compared to the Apple design. The change usually works but can cause subtle timing problems in some applications.

Beyond 512 KB, further memory expansion is a little more complicated, raising both hardware and software problems. Additional memory up to 4 megabytes total can be attached to the 68000 bus (adding more will run into the ROM memory addresses). Internally, space considerations probably impose a 2-megabyte limit with 256 kilobit chips. Such memory upgrades also require a new ROM or a modest alteration to the operating system at boot-up time; the changes include moving the video memory to a high address and remapping several other memory locations.

Memory-expansion problems will be partly resolved with 1-megabit RAM chips. The main logic board can hold 2 megabytes of RAM with such chips, and expansion to 4 megabytes within the existing case is straightforward. Two-megabit RAM chips will allow 4 megabytes on the main board. Expansion beyond 4 megabytes requires remapping the ROM addresses.

Disk Drives

Only bus-connected hard disk drives can offer full-speed performance. General Computer attaches its drive to the bus internally and adds a ROM to permit booting the Mac from the drive. In the future, bus connections will also be necessary for efficient operation with optical disk drives.

Coprocessors and Array Processors

Special processors that can take some of the load off the main processor can be added to an expansion bus. A numeric coprocessor can perform fast numeric computation; array processors manipulate matrices of numbers. Both are specialized tools for users who do intensive numeric computations and fast graphics.

Video Output

This innocuous modification adds a video-output connector on the back panel. A small circuit taps into the video signals traveling between the logic board and the analog board. The circuit combines the separate video and synchronizing signals into a single composite signal, which is suitable only for high-quality monitors and video projectors. For more information, see Chapter 28.

Possibilities

In terms of the present Macintosh, one ambitious modification would increase the number of pixels in the video display with a larger CRT; color displays are possible as well. But software support will be a problem for such modifications, so probably only Apple will be able to make these changes satisfactorily.

Also, an external expansion bus for accessory cards could be attached to the serial ports. Such an external bus system would operate much more slowly (about 29 to 62 KB of information transferred each second) than an internal bus (typically a megabyte per second or faster). As a result, the most important accessories are not possible. An external chassis could contain expansion slots, possibly using the Apple II or IBM PC bus, even though those buses were designed for different microprocessors. It's been done before: The 68000-based Corvus Concept microcomputer uses an Apple II expansion bus.

One particular type of external expansion product is already available: Instrument controllers connect a variety of sensors and processor-control equipment (see Chapter 24).

The Expansion Question

Should the Macintosh have been an expandable system instead of being a "closed" system? Expansion comes at a price: additional circuits, leading to a larger power supply and chassis, and a cooling fan. These additions would increase the price significantly. Besides, the vast majority of Apple II and IBM PC owners use their expansion slots only for essential components such as memory, clock/calendar, input/output ports, and video-display adapters, components that should have been in their computers to begin with.

The problem with the Macintosh is not so much that it is closed but that it was closed too small. The Macintosh desperately needs more memory, a good hard-disk interface, and at least one or two more ports. Provision for these features, needed by all serious users, would not have increased prices dramatically.

But all this points the way for the next Macintosh.

FUTURE MACINTOSH DESIGNS

The next Macintosh will have much more memory, up to at least 2 MB on the main boards, with expansion up to 4 or 8 MB total. Hard disk drives will be supported properly, along with additional ports. The central processor may change to the Motorola 68020, a faster, more powerful chip. The 68020 is a true 32-bit design and can address up to 4 gigabytes (4096 megabytes) of memory; the 68881 numeric coprocessor works with it.

Future Displays

New models will have larger video displays. Overall pixel counts of about 700 by 500 are practical at reasonable prices; a more expensive display might have 1000 by 700 pixels, a premium one, 1400 by 1000. The emphasis will shift to full-page displays (oriented vertically instead of horizontally) and two-full-page displays. Most machines will continue to use monochrome displays for their sharpness and relatively low costs. The absolute resolution will probably stay in the 75 to 100 dpi range; CRTs with finer dots are too expensive.

Color displays for the Macintosh will appear as well. The Macintosh ROM has routines that can handle eight foreground colors and eight background colors. But color raises significant hardware problems. Color CRTs use tiny striped triplets, or dots of colored phosphors. To achieve 80-dpi screen resolution, the pitch, or size, of the triplet must be 0.31 mm, the pitch of the best CRTs produced in quantity today. To produce a square-looking pixel, the electron beam must illuminate phosphors in four triplets, requiring a pitch of 0.15 mm and exact alignment. Such color tubes have been built only experimentally. Ordinary CRT magnets are not accurate enough to control such a fine electron beam. To line up the color accurately (convergence), these CRTs require a sophisticated controller that stores regional nonlinearities in the deflecting magnets; a processor adds the corrections to the video signal. These are expensive measures, and even with them the picture will not be as sharp as on a monochrome display.

Because color can convey more information than monochrome, you may choose to accept the fuzzier image. But you will probably use color only for graphics, not for text. Because the lens and cornea in our eyes lack color correction, distinctly different colors fall on slightly misaligned portions of each retina. This produces a "color-stereo" effect that makes the colors on-screen appear to be at different depths; reading such text induces eye fatigue.

If you plan to use color routinely in business, you will need to consider the implications. You may need to get a color printer; very few photocopiers work in color, sharply restricting your ability to share your results.

Printing Technology

Printers have changed rapidly in the past few years, moving from dot-matrix and daisy-wheel to laser technology. Here are some areas of continued development.

Dot-matrix printers

The best impact dot-matrix printers will print from 200 to 300 dpi. Speeds have already reached 200 characters per second, and future units will run even faster. Print quality will improve but still will not be letter quality.

Where it is technically feasible, new dot-matrix printers will become bidirectional graphics devices, incorporating a scanner for artwork. With a small vertical sensor array, a page could be scanned in as little as one minute.

Ink-jet printers

Ink-jet printers squirt tiny drops of ink onto the paper. Many current ink-jet printers are in the low-cost, low-quality range, but a new type with an array of ink jets will appear. The best will achieve medium resolution and some will print in color. Such printers are quieter and capable of better results than impact dot-matrix color printers. But they will not compete directly with laser printers because of their relatively large ink-drop size.

Thermal-transfer printers

Competing against the ink jets, thermal-transfer printers press a plastic sheet coated with color resin against the paper, and tiny heated pins melt the resin onto the paper. The resolution and print quality are generally better than that of ink-jet printers, but some people object to the glazed image. Ribbon costs are high. (Don't confuse thermal-transfer printers with the low-cost, low-quality thermal printers that use specially treated paper.)

Other technologies

Laser printers will reach about 800 dpi resolution. This appears to be the limit. Although laboratory experiments have created electrostatic images with over 1000 dpi, the toning and fusing steps squash the toner particles, cutting the resolution. At 800 dpi, one page contains 59,840,000 bits, or about 7.5 megabytes, placing great demands on the control electronics and communication link between computer and printer. Toshiba has described a prototype color laser printer with three laser and toner assemblies arranged around a single photoconductor drum.

We will soon see wider use of page-printer technologies similar to a laser printer. Most use a xerographic drum with toner development, like a laser printer, but they do not need moving parts to form the image. Arrays of LEDs and LCD shutters can create images; the Delphax printer uses a unique array of ion guns to squirt ions onto the photoconductor drum.

Magnetic printers use an array of tiny magnetic heads to record a magnetic image on a magnetic belt. Toner particles, themselves containing magnets, adhere to the belt and are transferred to the paper. Unlike other page printers, the magnetic image remains on the belt until erased, so many copies can be made from one magnetic image.

FINDING OUT ABOUT NEW PRODUCTS

You will have no difficulty finding out about new products for Mac. The problem is sorting out the good from the bad. The microcomputer industry is putting out a deluge of announcements touting new Macintosh accessories and software. Claims and counterclaims crowd the magazine advertisements.

Pre-announcements

Many companies engage in pre-announcements, saying they will do something months—sometimes a year or more—in advance of the first delivery. They may want to preempt the field and discourage people from buying competing products; or they may simply be optimistic—even competent companies can run into last-minute snags.

Magazines encourage pre-announcements by trying to be quickest with the latest news. For major products, a magazine may strike an unspoken deal with the company, trading a splashy article or cover in exchange for early information. Few magazines publish corrections when the hot news turns out to be mostly hype.

Computer engineers tell many funny stories about last-minute crash projects to simulate a product for the press or investment analysts. The demonstration may have nothing to do with the product—usually it's a lot of fancy screen graphics—but it almost always convinces the gullible.

One semi-legitimate reason for pre-announcements is the result of a scheduling problem. Magazines typically plan articles and sell advertising space for each issue three to four months ahead of time. Many products will fail if they do not start selling successfully within a period as short as a few months of the date they are released; if sales are delayed, competing products will appear. So some companies start the announcement process months before the product itself is ready, to make the publicity coincide with the product's release date.

The best companies take a more conservative approach, announcing products only when they are ready to ship. All other things being equal, these companies deserve more support than those that regularly jump the gun.

Rumor mills also generate many stories about forthcoming products. Most rumors are inaccurate, based on wishful thinking. When a company plans a new computer, it will usually design several alternate configurations; most will never be built for sale, but they will generate additional rumors.

Problems with New Products

Should you always look for the latest and most glamorous product? I recommend that you resist the temptation, especially if you depend on the Macintosh in your work. Instead, concentrate on products and companies that have an established reputation. Ask other people what their experience with a particular product has been. You might not always be the first with something new, but you won't be an unwitting guinea pig either. Far too many computer products appear in an unfinished form; some companies regard their first customers as the last step in the product-engineering process.

New software often does not work quite as well as the developer planned. Bugs, or errors, are common, particularly in early versions. Some bugs you can learn to live with, especially if they afflict only little-used features. A minor glitch may make a particular function awkward but not detract greatly from a program's larger benefits.

More serious are major bugs that affect the program's usefulness. A database program with a delete-record command that doesn't work, for example, can only lead to frustration and unpredictable results.

Such major bugs aren't unusual; several best-selling programs for other computers have had such faults for years. Sometimes the seller will try to explain away the bug as an "undocumented feature."

The chance of bugs increases with program complexity. Often, very complex programs are never completely debugged; it becomes a question of how severe the bugs are and whether it is worth the effort to fix them. Frequently the bugs are fixed only in a new version that adds features—and new bugs. Nevertheless, the most highly developed programs should be bug free for all practical applications.

MINIMIZING PROBLEMS WITH ACCESSORIES

Although you will generally have few interface problems if you use only Apple or Apple-approved products, there are often good reasons not to use them. Other products may be cheaper or have more or better features, or you may already have an accessory, such as a modem, and not want to buy another.

One disadvantage of using other products is that your system then has components from different vendors who may have no interest in helping you get their product working with those made by other companies. And Apple usually can't help you either; it cannot possibly keep track of all the products that work with Mac. As a result, you may find that a printer which offers many features with one particular program proves awkward or even impossible to use with another.

Another disadvantage: When Apple updates system software, such as the Finder or operating system, it will probably update everything in Mac's product line that is affected by the change. If you are using other products, there will probably be delays in getting updates for them, resulting in a period when things won't quite work together. The healthy independent companies will, however, update their products quickly.

How do you tell a healthy company from an unhealthy one? A long-term reputation helps. With the rapid growth of the microcomputer industry, many ill-planned business ventures have started up. Most of these will fold. In the early years of microcomputers, many successful companies, including Apple, started in the proverbial garage. Unfortunately, those days have now passed, at least for the development of major products. Few important hardware or software products can succeed now unless they are supported by millions of dollars in development, packaging, and promotion.

Buy with care from a company you have never heard of; you may never hear from it again. Many smaller companies—and some bigger ones—will announce and ship a new product, only to discover problems and decide that the product is too difficult to redesign or support further. Customers who bought the product will be left holding the bag. These companies may have made an honest mistake, but if you've paid for the product, you will probably be stuck with it.

The Future of Microcomputers
Chapter 23

What does the future hold for microcomputers? This chapter looks at future hardware and software developments, not just for the Macintosh but for other microcomputers as well. Unless otherwise noted, the projections are for the late 1980s.

Although the computer industry, especially the microcomputer industry, is changing rapidly, the short-term future is fairly easy to predict. All the manufacturers use the same technologies; there are only a handful of important microprocessor chips and only a few ways to make a screen display. Everyone works with the same memory or disk-controller chips, and the same disk drives. Every hardware component that will be commercially available in the next four to five years already exists in some kind of prototype today.

Within five years, every successful computer company will offer a full family of models, from powerful desktop designs to under-4-pound, battery-operated portables. All the various models will work together, using as similar an interface as the specific hardware allows. With little effort, files will move back and forth between any members of the family. Whenever you connect two computers, the operating system will automatically update files and make backups by checking the time and date of each file.

CENTRAL PROCESSING UNIT

The follow-on to Mac's 68000 processor, the 68020, is already appearing in prototype computers. Although the 68020 and other new processors will operate faster than those currently in use, software will become more complex; overall speed may not, therefore, appear to increase: If the new processor runs twice as fast but a complex new program does twice as much work, there may be no speed increase from a user's standpoint.

MEMORY

Memory prices were dropping at the rate of 4 percent per month for the last two decades; then suddenly in late 1984 and early 1985, memory prices dropped sharply—over 80 percent in six months.

The Macintosh currently uses 256-kilobit RAM chips (the early 128 KB Macs had 64-kilobit chips). Prototype production of megabit RAM chips has already begun, and Japanese companies are working hard on 2-megabit chips for television frame buffers. If the television market for memory chips takes off, it could dwarf the computer market and produce dramatic price reductions.

The next round of computers will handle 20 to 50 megabytes of memory. By present standards, this is a lot of memory—so much that the short-sighted claim that we will never need such power. Future software will use all of it and, like today, advanced users will demand even more memory.

DISPLAY

The long-established CRT will remain supreme in desktop computers through this decade. No other display technology has equaled its relatively high contrast and resolution for a sufficiently low manufacturing cost. Only in portable computers will alternative flat-panel technologies dominate—and then only because size, weight, and power constraints are more important for portables than resolution or price.

CRT resolution will improve steadily. Mac's 75-dpi screen will give way to a 150- to 200-dpi black-and-white screen, about the resolution limit for CRT designs with present technology. More screens will display an entire 8½- by 11-inch page; some will display two pages, enough for effective window operations. In addition, some displays will move away from simple black and white to handle grays. Switching from digital to analog video modulation will permit more precise positioning for better-looking fonts and graphics.

High-resolution color screens are much harder to manufacture at reasonable cost than black and white, but color screens with Mac's present resolution will start appearing for microcomputers.

The higher the resolution, the longer it takes to compute the display's contents. Most present microcomputers, including the Macintosh, use the main microprocessor for computing the display. Future micros will incorporate a special graphics processor to handle the display, freeing the main processor for other tasks.

MASS STORAGE

Computer storage capacity has been growing fast—even faster, relatively, than the Library of Congress has been growing. Microcomputer storage capacity is growing much faster than mainframe capacity, but cost restrictions are likely to keep mainframe storage 1000 to 10,000 times the capacity of micros.

Floppy disks will remain widely used for years to come because the disks themselves are so inexpensive to make. Using new technologies, storage densities will make a big jump, to about five megabytes. Such densities will bring some problems. Dust will be a bigger nuisance than ever, and the mechanical alignment of the drives will need to be much more precise.

Meanwhile, hard disks will become standard as the retail prices for 10- and 20-megabyte drives drop to just a few hundred dollars apiece. More expensive drives will store over 200 megabytes on a hard disk the size of a floppy disk drive.

Then, however, hard disks will lose their luster in the face of optical disc drives. CD-ROM (compact disc-read only memory) disc drives will be common by 1986. A CD-ROM, using a read-only format borrowed from audio compact discs, can store 550 megabytes on a platter 4.7 inches in diameter. Manufacturing costs are low for both players and discs, for they are essentially the same as audio compact-disc equipment.

You cannot record on a CD-ROM, however, so they won't serve for general-purpose storage. They will be outstanding for mass storage of reference information—an encyclopedia or a massive font library, for example.

Other optical-disc technologies now available can write information permanently on a disc once and then read it back; they cannot erase information. But laboratory prototypes of erasable optical discs have already been constructed. Erasable optical discs with a capacity of 2 to 10 gigabytes should be available for a few thousand dollars in a few years. Since one gigabyte can hold about 2000 books, entire libraries could reside on one disc.

NETWORKS

Networks will become common and interlinked by the end of the decade. Present local area networks will include gateways to the telephone system and, thus, to other networks. Electronic mail and file sharing, whether with the next office or another continent, will be embedded in computers as an operating-system function. You won't have to keep track of where a particular file is located; the system will automatically find the most recent version.

TELECOMMUNICATIONS

Unfortunately we appear to be in for a period of low-quality telephone lines because of deregulation and the competition to provide low-cost service. But the future does hold promise.

Many electronic telephone exchanges already in use can accommodate digital phone lines capable of transmitting 56 kilobits per second. These lines will, in the near future, be wired to most of our offices and homes, offering decent communication speed at last; a double-spaced, typewritten page will transfer in ⅓ second. Even the most complex Macintosh screen will transmit in 3 seconds; most will take much less time. Conversion to high-speed phone lines will take time and a massive investment in optical-fiber links between exchanges. Many present local loops, from the local exchange to our telephones, already can pass 56 kilobits, sometimes even a megabit, per second.

New, experimental systems (from Pacific Telesis, among others) take advantage of such capacity by combining voice and several data channels on the same phone line, using a coding device at the telephone and a complementary decoder at the local exchange. Multiple data channels can be useful for burglar alarms and other sensing devices. The phone companies are delighted; they can charge you for several calls simultaneously—your own and those made automatically by the sensors.

Information services available by telephone will finally become easy to use. With higher transmission speeds and better graphics, these services will attract more people, in turn bringing the cost down low enough to attract even more people.

Everyone also hopes that the advent of the new high-speed links will help standardize communications protocols. Standardization is possible but unlikely. New software will at least insulate individual users from the protocol problems by automatically detecting which protocol is required and making any necessary adjustments.

GETTING INFORMATION INTO THE COMPUTER

In this decade, the keyboard, with help from a mouse, will remain the dominant method of entering information into a computer. Everyone wants a computer that will recognize human speech, but the problems are formidable.

With present technology, lower-cost units can recognize a limited vocabulary of isolated spoken words, but these units are speaker-dependent—they must be "trained" to recognize individual speakers. Expensive systems can interpret limited connected (but not continuous) speech with a vocabulary up to 500 words.

A powerful system able to understand standard spoken business English will probably not be available until the 1990s. Such a system will require a sophisticated program to reduce the many ambiguities inherent in spoken English.

SOFTWARE

Although future software will still do present tasks, the ways in which the software does the work will change.

With more memory, computers will store more programs in RAM at the same time. When we switch tasks, we will not have to wait for disk drives to store files and then load new ones; the switching will be instantaneous.

Live links already work in a limited way in Lotus' Jazz; changes you make in one file are automatically reflected in all dependent files. New software will extend live links not only between files in RAM but on disk as well. If, in a law office, you use a boilerplate paragraph in several documents, and then change the boilerplate, the system will not only ask if you want all documents changed, but whether you want to send a new copy to all previous recipients.

A second major software direction will be toward natural language—ordinary English. To retrieve information from a data base, we will no longer have to learn a rigid series of commands. We'll be able to type a question in plain English; the software will analyze the syntax of our question (parse it) and figure out what information we want. Parsing is akin to grade-school, sentence-diagramming exercises combined with a dictionary of meanings. With parsing, computers can for the first time be said to "understand" language.

Microcomputer data bases that accept simple natural-language input began appearing in 1984 (for the IBM PC), but that's just the beginning. The twin steps of parsing and linking open the way to a new era in software.

DATABASE MACHINES

All of us work by building upon old data. But people are usually disorganized; computers need to work with organized data. With today's computers, we spend an inordinate amount of time organizing our information so the computers can understand it.

Letters, for example, begin with an address and other information that is already stored in an address book or file. With present software, you must find the address or an old letter by explicitly searching for it. Future software will search automatically. The program will parse your input as you type. If you type the name John Doe, the program will immediately put John Doe's address and telephone number in a secondary window. If there is no John Doe, the program will wait until you type in the new information.

Regardless of where or how you type the address, the program will detect an address by scanning for "Ave.," state abbreviations, and so forth, and then look in nearby text for further address information. The information will automatically be organized into address format; you will have a chance to check it and then it will be added to your name-and-address data base.

If your letter begins "In response to your letter of last week. . . ," the program will automatically go off, having figured out that "last week" means September 8 through 14, and check for letters from Doe. (All letters will arrive via electronic mail, of course.) If no such letter is found, the program will expand its search, looking farther back in time. If you regularly underestimate the date, the program will automatically start looking earlier.

If you refer to a company name in the letter, all the information in your files about that company will be available instantly. The program will anticipate all common data needs, accepting natural-language guidance for less common requests. ("What is the telephone number of the hotel closest to the meeting?")

The software will be unstructured. You will not have to ask for a word processor or a data base when you start working; one large, seamless program will analyze and suitably organize your work. All program functions will always be available without arbitrary boundaries between them. The software will analyze and adapt to your specific working habits. If you enter calendar dates in a specific way, the program will adopt the same format, while retaining the ability to read dates in any format from other sources. On-line dictionaries will always check spelling and offer corrections for common mistakes.

This is extremely complex software; it can only be written by large, well-organized teams of programmers.

MICROS AND MAINFRAMES

Microcomputer development takes much less time than mainframe development. The most time-consuming period in microcomputer design is for the central processing chip; once the chip is available, a complete microcomputer based around it takes only about a year to design and build.

A new mainframe design takes five to six years. The major mechanical problem in mainframe development has become figuring out how to get rid of the heat generated by the circuitry. Ultimately, micros are likely to have about one-tenth the power of mainframes and a far superior cost/performance ratio.

How does a microcomputer compare to a mainframe in power? The design differences are too large to make simple comparisons possible. Minicomputers and mainframes must usually deal with multiple users simultaneously and therefore have far more sophisticated input/output arrangements than micros. The most common figure of merit for computing power is in millions of instructions per second (MIPS). The MIPS figures are an interesting but by no means comprehensive measurement; the figures tend to downplay the overall throughput differences among computers. Of the current computers, a Macintosh performs about 0.4 MIPS and the most powerful microcomputers about 1 MIPS. A DEC VAX does 1 to 4 MIPS, the larger

IBM mainframes around 15 to 30 MIPS, and the most advanced supercomputers reach 150 MIPS. Large computers will improve by 3 to 10 times by 1990. Micros will grow faster, possibly reaching 5 to 7 MIPS in 1988 and 15 to 25 MIPS by 1990.

Micros have modest computing power but are easy to use; mainframe computers have lots of raw computing power but are extremely difficult to use. In 1984, about 75,000 mainframe computers were installed, roughly a week's production of microcomputers. In the long run, mainframes will become utility devices for micros, serving for complex programs and large database needs. Users will only see the microcomputer interface; it will hide all the problems of mainframes.

A CONCLUDING NOTE

Does this chapter sound too futuristic? It isn't (except perhaps the predictions about voice recognition). We already have, or will have within months, the hardware able to perform these marvelous functions. The software will take longer, so we'll probably be forced to wait until the end of the decade for truly powerful microcomputers.

Sectio

Special Macintosh Topics

Four

This section contains material for people with specialized needs or interests. Some of the material is highly technical, written for specialists and intended primarily for reference.

A Macintosh Medley

Chapter 24

When I finished writing this book, I found I had lots of useful bits of information left over; they didn't seem to fit logically anywhere. Believing that more information is better than less, I couldn't bring myself to leave them out; the result is two chapters, this one and Chapter 29, that should rightly be titled Miscellaneous One and Miscellaneous Two.

Chapter 29 is a compendium of technical tidbits that might interest experienced users; this chapter contains several unrelated topics that might be of wider interest. They are:

- Specialized Accessories.
- Using the Macintosh in Moving Vehicles—including motor vehicles, boats, and airplanes.
- Privacy.
- The International Macintosh.
- Notes for the Handicapped—for those with hearing and vision impairments, those with motor-control limitations, and the one-handed.

SPECIALIZED ACCESSORIES

Telephone management systems. A telephone management system dials numbers and tracks your telephone usage. If you make many phone calls in a business, particularly if you need to bill the calls and their duration to different clients, you may find a system like MacPhone (Intermatrix) useful. For ordinary situations, most people will find a conventional stand-alone telephone dialer simpler; it will be much quicker than digging out a disk and loading a program just to dial a number.

Video accessories. Videodisc and videotape controllers provide all the functions of random-access tape and disc players. The video image goes to a standard television or monitor, not to the Mac screen, which continues to work in the normal way. Training software can use the Mac screen for text and high-resolution graphics while putting a normal, full-motion video image on a television screen. Other programs will use videodiscs for games; you use the Mac's screen for the computer interface, but watch video action on a separate television.

A Macintosh will also control video tape recorders and film-editing equipment, with a hardware interface that reads and generates standard SMPTE time codes specified by the society of Motion Picture and Television Engineers. A video digitizer will grab frames you designate, creating miniature images on the screen. You can edit by manipulating the images, a great improvement over traditional systems with which you must memorize the images; the software will automatically assemble the finished tape by issuing commands to the recorders and video-effects equipment.

Sound and music accessories. The Macintosh by itself can synthesize music sounds adequately for teaching and demonstration purposes, but serious musicians will want to use the Mac to control professional musical instruments, keyboards, and synthesizers. Many electronic instruments use the Musical Instrument Digital Interface (MIDI) for electronic controls and interconnections. MIDI is a 31.25 kilobaud serial protocol, supporting 16 polyphonic channels on each circuit.

Total Music (pre-release, from Southworth Music Systems, Inc., Harvard, MA) is an outstanding music composition, editing, and playback system; you can compose on music staffs with the mouse, specify voices on the synthesizer, set and change tempi and key signatures, and print the music notation. You can pick out the notes one at a time and record them on a MIDI keyboard, and Total Music can correct the timing on playback. Total Music includes interface hardware that accepts two MIDI inputs and generates four outputs.

Many other MIDI products (all pre-release) are appearing for the Macintosh. ConcertWare and MusicWorks will have MIDI output. MIDI hardware interfaces for the Mac are planned by Assimilation, Opcode, and Musicworks (the latter is unrelated to the MusicWorks software package). Pre-release software from Opcode includes sequencing and some editing; from MusicWorks, the DX7 Librarian which can read and write voice cartridges for the Yamaha DX7 keyboard synthesizer. There are also some non-MIDI music interfaces: Digidesign has software for controlling the E-Mu Emulator synthesizer; Kurzweil had a sound and sequence storage system for its keyboard.

A Macintosh can also control multitrack tape recorders and a digital mixing board in a recording studio, marking and displaying sound fragments as a compressed spectrogram; again, as with video accessories, you can edit graphically.

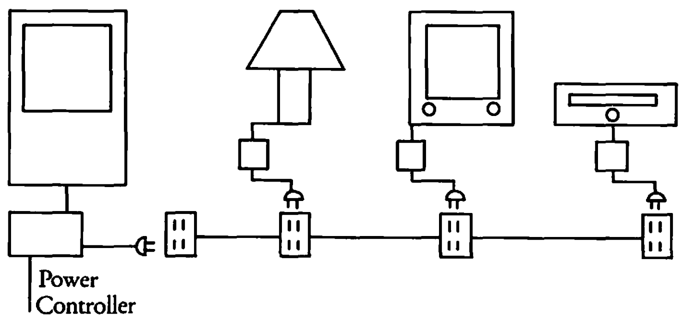
Instrument controllers. Instrument controllers let you collect and analyze data from a wide variety of instruments. An A-to-D converter changes an analog voltage from any source (thermometers, machinery, fuel-tank gauge, and so on) into a digital signal for the computer. A digital-to-analog converter works in the opposite direction, turning a digital signal into an ordinary analog electronic circuit. These products are designed mainly for laboratory or manufacturing applications.

IOtech (Cleveland, OH) has Mac488, an IEEE 488 bus controller for a serial port. The IEEE 488 bus can handle up to 15 devices on the bus. All the equipment can be controlled with commands sent out through the serial port; programming can be done in any language. Metaresearch (Portland, OR) sells software and hardware developed by Reed College, including an icon-driven interface for controlling an instrument network where each device has an Intel 8031 processor.

Many companies make controllers that connect through RS-232C serial ports. There are several products that have explicit software support for the Macintosh: Macquisition and MacControl software (from Small Business Computers of New England, Amherst, MA) interface the Mac to Taurus One data acquisition and control peripherals. The software can put the acquired data directly into a Multiplan spreadsheet. A second, unrelated product, MacCONTROL software, handles the ADC-1 A-to-D converter peripheral (both from Remote Measurement Systems, Seattle, WA).

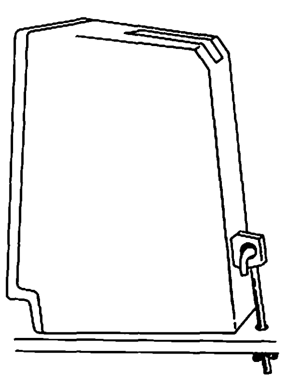
A digital oscilloscope attachment will turn the Mac into an electronic tool to analyze waveforms. Using the computer, you will be able to store, compare, and analyze electronic signals. A logic analyzer will let you trace digital circuits. Signal generators will use the screen to display the waveforms; analyzers will display the distortion products. An entire electronic testing set will be centered around the Macintosh. These will mostly be tools for engineering.

Power controllers. Power controllers working with X-10 switches will let you turn appliances on and off with your computer. Computer-operated controllers make sense for industrial applications and for the handicapped; they are less useful for ordinary offices or households. MacCONTROL with the ADC-1 can operate X-10 switches.



A computer-operated power controller can turn any standard electric device on and off.

USING THE MACINTOSH IN MOVING VEHICLES



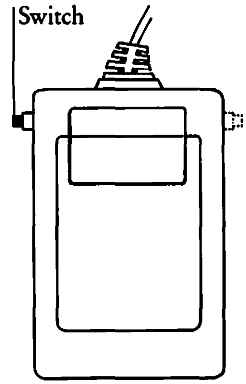
Fasten down the computer with the keyhold lock from Apple's security kit.

If you have a source of power and a table, you can use Mac while traveling—in a boat, a large motor vehicle, or an executive airplane.

You may have a few problems if you run into heavy seas, rough roads, or turbulence that can upset the main computer unit or toss the mouse around. Since the main computer unit is a little top-heavy, fasten the computer to the table. The keyboard can be held down with its security lock or you can simply tape it down.

The mouse might create problems if it rolls around while you are using the keyboard. Some accessory pads will restrain the mouse. As an alternative, you might consider a touchbar or a trackball. However, since the mouse isn't active unless you push its button and the pointer never runs completely off the screen, you can probably live with a loose mouse.

You can modify a mouse so it is active only while you are holding it. Add a small push-button switch to the side of the case (the left side if you're right-handed). The switch should normally be open (that is, you should have to push it to make a contact). Wire the switch in series with pin 2 in the mouse cable, the pin that carries the 5-volt supply for the LEDs in the mouse. When the switch is open it will extinguish the LEDs and the motion sensing. When you palm the mouse and press the switch, you will activate the LEDs and restore mouse operation.



A switch on the mouse can activate motion detection. (The mouse button is not affected.) Switch should be momentary-contact, push-to-close, wired in series with pin 2 of the mouse cable.

Although fairly compact, the Macintosh is not ideally packaged for carrying around. Colby Computer (Mountain View, CA) rearranges the Macintosh into a more convenient traveling package; everything packs up into a self-contained, 26-pound case. Colby's case also has space for the second floppy disk drive, a modem, and a hard disk.

The Macintosh is certified to be sufficiently free of electromagnetic radiation for home use (FCC type B), but there are no certification procedures to qualify a device as safe to use on aircraft. For lack of space and power, this isn't an issue on commercial airliners, but you may want to use your Macintosh on a private aircraft. Consult with the companies that supply the avionics to see if they have advice or tests concerning possible interference from the Mac. In any event, prudence suggests that you not use any device that might radiate interference during takeoff or landing procedures, or while flying near potentially hostile airspace.

An unpressurized aircraft is one of the few situations where a Macintosh might be used beyond its altitude limit—15,000 feet or 4500 meters. (See Chapter 29 for an explanation.)

PRIVACY

Many people are much more careless with their disks than with their papers. If you have confidential information in a computer, keep it on a floppy, not on a hard disk, and lock up the floppy when you aren't using it. If you must put the information on a hard disk, erase the file afterward with a utility program that not only changes the directory entry but magnetically wipes the tracks clean as well. The Purge function of P/C Privacy (MCTel, Bala Cynwyd, PA) does this.

You can also use an encryption program such as P/C Privacy or Lock-It (Assimilation) to encrypt the information so that a password is needed to decrypt or read the file. You must go through the decryption and encryption process every time you use the file and if you forget the password, you've lost the file. If you are the only user of the file, locking up the disk in a drawer is probably simpler; if several people use the file or it must be sent somewhere, encryption is better.

Encryption programs can also help with electronic mail. If only you and the recipient know the password, eavesdroppers cannot decode the message even if the file is on a public bulletin board. Of course, encryption schemes can also be broken, just as locks can be picked, but such programs will deter most snoopers.

If your computer is connected to a network, disconnect it for privacy, especially if your network software makes every storage device available to the network. Password protection on network file servers is usually weak; the files themselves are not usually encrypted and the passwords must be stored somewhere for checking. If you must store a sensitive document on a network device, use a freestanding encryption program that uses a password as the basis for the encryption algorithm but does not actually store the password anywhere.

Instrusions through telephone lines are rare, since microcomputers are rarely left permanently connected to the phone line and callers cannot, in most cases, gain access to a file directory or the operating system. If you do set up an unattended system for remote access, you should at least use a password system. If you have problems, use a system that keeps an audit trail of activity and automatically disconnects on repeated efforts to gain access. The best simple protection comes with dial-back modems. When you call up such a modem, you get no sound, just silence. You then enter a touch-tone code number and hang up. The modem checks your code number against an internally stored list of numbers and calls you back, this time establishing a modem connection. Since it only dials numbers already held in memory, the system is hard to break into. Call forwarding can be a weak link, though; consult with data-security experts if the stakes are high.

Most computers generate spurious electromagnetic radiation that a sophisticated snooper can interpret. Careful shielding can reduce or eliminate the radiation (the military Tempest specification is a maximum spurious-radiation standard). Again, you should consult with data-security experts if you need such protection.

Also, watch what you throw in trashcans. Most printer ribbons create no security problem, but single-strike carbon ribbons, such as the IBM Selectric ribbon used by the Juki 6100 printer, can be read after use. If you do not have a paper shredder, you can tear paper into strips and throw it away in two locations. Finally, if your hard disk needs service, think about its contents before sending it out; you may find it is cheaper to replace the disk than to leak its contents.

The serious fear monger can go on and on, but I will stop here.

THE INTERNATIONAL MACINTOSH

The Macintosh is available with the Finder, MacPaint, and MacWrite in many languages: North American English; United Kingdom English; French; Canadian French; German; Spanish; Dutch; Japanese in Kana, with some Kanji software; Italian; Flemish; Swedish.

The international Macintosh has a slightly different keyboard in some countries. The Return key is in a slightly different place, and there may be an additional key for symbols to the right of the left Shift key. The character codes are mostly the same, even if the keys are rearranged; most screen and laser fonts have all the common characters used by most European languages.

Many programs are available in several languages. If you share files internationally, look for translated software in each language of interest. Some programs have special arrangements to deal with different currency symbols. International versions of Microsoft Multiplan, Chart, File, and Excel store the language version with each document. If you open the document with a different-language version of the program, it asks you to check the currency symbol. Microsoft Chart additionally asks if you want to enter a currency-exchange rate and rescale the values on graphs.

NOTES FOR THE HANDICAPPED ON USING MACINTOSH

This section offers a few suggestions on ways that the Macintosh can be adapted for use by people with certain physical handicaps. For more ideas and information about products for handicapped computer users, contact the organizations listed at the end of this section.

For the Hearing-impaired

If you are deaf or hard of hearing, you will generally have little difficulty using Mac. You can set the speaker volume to maximum with the Control Panel desk accessory or use an auxiliary amplifier and speaker. Other than for programs specifically designed to produce sound output, the speaker is used only for warning tones. These warnings are nearly always accompanied by a dialog box or an alert-box warning message, so missing the warning tone does not matter. For a very simple visual indication of the tone, you can build a small circuit with a resistor and an LED attached to the audio output port.

TDD/Baudot terminals (sometimes called TTY) are widely used by the hearing-impaired to communicate with typed characters. To use Mac as a TDD terminal, you will need a communications program and a 110-baud modem that can handle five-bit Baudot code. These items should be available from independent vendors.

There is no practical way to use Macintosh to decode closed-captioned television programs; the cost would be higher than the present decoders available for use with television sets.

For the Vision-impaired

Macintosh can be adapted more readily than other microcomputers to work for people with certain forms of visual impairment. People with acuity limitations will benefit from the variable type size used by word processors and some other programs. The font sizes offer the computer equivalent of large-type books; even larger sizes of type are also available. Not all programs offer flexible type sizes, however.

With MacWrite, use Font, FontSize, and Style to select a typeface and size for comfortable reading.

The small type used to name each icon in the Finder may be a problem. However, since each icon stays in position unless you move it, you can keep track of positions without reading the file names. Programs should be available that let you redesign the icons; use these programs to make bolder icons. Similarly, a program could create larger file names on screen.

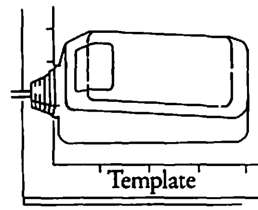
The on-screen pointer for the mouse is limited to 16 by 16 pixels. Since the shape is under software control, the pointer could be redesigned for greater legibility.

In some cases a larger screen will help. See Chapter 28 for information about adapting alternative display screens to Mac.

The speech-synthesis feature will help the totally blind. Since the speech synthesizer can read any standard ASCII text, the Macintosh could serve as a general-purpose reading machine. The problem is Macintosh's visual interface, which is less suitable for a totally blind person than an old-fashioned command-line interface. If your associates or relatives prefer a Macintosh for its ease of use, however, you can still use the Macintosh as a reading machine by circumventing the main problems.

To get around the desktop and its icons, you can use FastFinder (Tardis Software, Pebble Beach, CA), which substitutes a typed, command-line interface for the Finder. You can also use the standard finder by setting up disks so that icons appear in fixed locations on the screen. Do this by resizing the disk window to the full screen, then put the icons in a specific order and choose Clean Up from the Special menu. Eject the disk without closing the disk window. Make a plastic template with notches marked off for the mouse movement corresponding to the standard spacing of icons within the Finder.

To use the disk, insert it and move the pointer to the upper left corner of the screen by rolling a long distance up and left. Then pick up the mouse and put it into the corner of the template. To select an icon, run your thumb against the template and feel for the notches while you move the mouse. You can also use a digitizing tablet for selections; use a cover sheet printed with a screen dump of the disk window and identifications in braille.

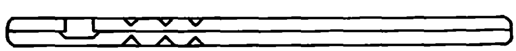


If you are blind, you can find icons by first rolling the mouse to the left and up; then by rolling the mouse along the template and counting the notches that mark each icon's position.

Use as many keyboard commands as possible during operation. Within a program, the Mac-Tracks keyboard enhancer can simplify menu selections. You can eject a disk at most times, even when the Finder is not on the screen, by typing Command-Shift-1 (internal disk drive) or Command-Shift-2 (external disk drive), but sometimes the system will want the disks again later. Disk ejection via the Finder is more reliable.

Disk identification is another problem. The thin disk-drive slot doesn't leave enough space for attaching raised lettering or braille labels to disks. You can open the slot a little by filing on the upper edge; remove the front panel from the computer first.

Another way to identify disks is by cutting small notches along the edge. Use a knife rather than a file so plastic dust won't contaminate the disk; microfloppy disks are not sealed along the edge. Cut at a 45-degree angle instead of with a vertical cut; this gives you two more edges for coding.

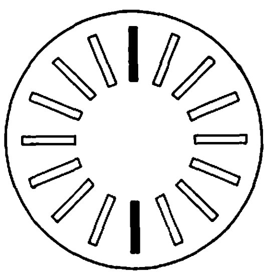


Identifying disks by touch: Cut small notches at 45-degree angles along the left and right edges. Don't cut vertically, since the edges are not sealed and shavings might slip inside.

For the One-handed

The three keys that perform shift functions—Shift, Option, and Command—are used not only with the keyboard but also with the mouse. For one-handed operation, special software will let you toggle these keys, turning them on and off through a keyboard command. Since a software toggle might not show the status of each key, the switching could be done by a conventional on/off switch, wired in parallel to the key. There is space along the top of the keyboard to mount such switches.

For Those with Motor-control Limitations



If you want to change the "gearing" of the mouse permanently, you can block optical-vane slits in opposing pairs in the mouse. This is not an easy modification to make.

If you have difficulty with fine motor control, the mouse may present problems. Sometimes the solution may be the same as that for some graphic-arts problems: Change the "gearing" of the mouse. Such changes, however, are only easy for factor-of-two alterations. A permanent change can be made by opening the mouse and covering some of the slits in the optical vanes that interrupt light between two pairs of LEDs and phototransistors. The slits must be covered in opposing pairs. Doing this is a little tricky. Modified software could be designed to register only smooth mouse movements and ignore sudden or jerky motions. The MacNifties optical mouse that reads the grid lines on graph paper for motion sensing may help out as well.

Some users may find that moving the mouse button to a different part of the mouse or out of the mouse altogether makes it easier to operate. An alternate switch can be installed between the mouse and the main computer unit (a diagram of an accessory switch is shown in Chapter 17). In some cases, a trackball, joystick, digitizing tablet, or touchbar may help; see Chapter 13. For many programs the mouse isn't necessary; both the Mac-Tracks keyboard and the FastFinder program can help out. Also see the description of head-mounted pointing devices in Chapter 13.

Sharing Good Ideas

If you have found a useful way for a handicapped person to use Mac or any other computer, publicize it. Write an article for a magazine or a letter to the editor. Put suggestions on electronic bulletin boards and pass them on to organizations for the handicapped.

Some information sources about computers and the handicapped:

Organizations:

The Trace Research and Development Center on Communications,
Control, and Computer Access for Handicapped Individuals
University of Wisconsin
314 Waisman Center
1500 Highland Avenue
Madison, WI 53705

Council for Exceptional Children
1920 Association Drive
Reston, VA 22091

Access Ability, Inc.
(Mac modifications for the spinal-cord injured)
60 East 6th Avenue
Vancouver BC, Canada V5T 1J4

Newsletters:

Closing the Gap Newsletter
(includes Computers for Disabled Individuals Directory)
Route 2
Box 39
Henderson, MN 56044

Sensory Aids Technology Update
(newsletter for the visually impaired)
Sensory Aids Foundation
399 Sherman Avenue, Suite 12
Palo Alto, CA 94306

The Sloane Report
P. O. Box 561689
Miami, FL 33256

Job training program for the handicapped (emphasis on IBM PC):

Association of Rehabilitation Programs in Data Processing
Physically Handicapped Training Center
University of Pennsylvania
4055 Chestnut Street, 3rd Fl/T7
Philadelphia, PA 19104

Communications Chapter 25

Communications are difficult and complex and this chapter is necessarily complex as well, ranging far beyond the scope of most books about microcomputers. If your communications needs are straightforward, you may not need to read this chapter at all; instead read Chapters 8 and 18 to learn the essentials of the most common types of communications. Read this chapter if you need more information about specific topics or want a broader understanding of the subject. Chapter 26 takes up AppleTalk and other local area networks; Chapter 27 discusses what to do with the information you move between computers.

THE VARIETIES OF COMMUNICATIONS

Communications include many overlapping areas. The answers to three key questions guide us through this difficult terrain:

With whom do you want to communicate?

- Another Macintosh.
- Other microcomputers.
- Large computers.
- A dial-up data base (Dow Jones, THE SOURCE, CompuServe, and others).
- Facsimile.
- Printers.
- Typesetting machines.

How will you communicate?

- Direct connection to another computer—
  Asynchronous link with or without error correction;
  Synchronous link, including error correction.
- Ordinary telephone lines—
  Asynchronous;
  Synchronous.
- Special leased or "dedicated" telephone lines—
  Asynchronous;
  Synchronous.
- Local area network (LAN) shared by many computers—
  Networks with Macintoshes only;
  Networks with various computer models.
- Exchange disks.
- Printed text and optical character reader.
- Facsimile.

What do you want to communicate?

- Unformatted text.
- Word-processing files with formatting intact.
- Graphics—
  Using Apple's QuickDraw format;
  In North American Presentation Level Protocol Syntax.
- Data files from spreadsheets or data bases—
  Text output only;
  Values or numbers only;
  Complete file with all formulas and relationships intact.
- Programs—
  Source code in ASCII;
  Binary file in machine language or a special communications format.

Each of these areas could fill an entire book, and many of them have. This chapter examines the basis of communications by describing the level of protocol associated with each area, and then looks at some specific topics, such as using a Macintosh as a terminal to a large computer.

THE BEGINNINGS OF CHAOS

Communications between Macs may not be difficult, but problems begin when you try to use a Macintosh to communicate with other computers. The problems have nothing to do with the Mac or Apple specifically; the headaches belong to the whole computer industry. Occasionally you can get a transfer between computers to work quickly; more often it takes hours, sometimes even days, to establish a connection the first time.

The problems lie in the profusion of protocols (the specific form of the signal) and hardware standards and non-standards that litter the microcomputer industry. Furthermore, the words that describe communications are often used ambiguously, adding to the confusion.

The ISO Protocol Layers

Considering how complex communications are in the world in general, the chaos in the computer world may not seem so surprising. In speech, for example, we follow protocols about who should speak first, forms of address (Mister, Ms, Sir, Madam), ways to keep others from talking ("... aah ...," "you know"), and so forth. For most of us, these protocols are ingrained as part of our social training. Keeping the many levels of computer protocols straight, on the other hand, requires some concentration.

The International Standards Organization (ISO) has developed a model with seven protocol layers to describe communications among different computers. The following explanations of these layers include computer examples—which will be explained later—and analogous concepts from other forms of communication, principally the postal system. This list starts at the lowest hardware layers and works up to the highest software layers. Each successive layer builds upon the lower layers.

The physical layer. This layer comprises the mechanical, electrical, and functional arrangements necessary for a physical connection. Examples: Computer cables, telephone lines. Analogies: Trucks, railroads, postal-delivery personnel.

The data-link layer. This electronic protocol is used to convey a unit of information from one node to another. A node is any device—a computer or printer, for example—capable of sending and/or receiving information. This layer includes flow control—who sends data and when—and some forms of error control. Examples: An asynchronous serial connection, a 1200-bps modem. Analogy: An envelope.

The network layer. This layer determines how information from the sender is routed to the correct receiver. This step is usually missing if only two devices are involved. Example: A token-passing network. Analogies: The address and the mail-sorting process.

The transport layer. These are the steps taken to ensure high-quality network service, including confirmation that the information has reached its destination and has been read without error. This layer is not always used; many computers simply send information out without knowing where it goes. Examples: The error-detection and correction features of many communications programs. Analogy: The sender requesting a return receipt from the post office or acknowledgment from the receiver.

The session layer. This is the procedure by which the two communicating computer devices coordinate action. Example: The sending computer requesting that the receiving computer open a file, accept information, and close the file; the user sitting at the receiving computer does not need to intervene. Analogy: The sender asking the receiver to perform an action they both understand.

The presentation layer. This protocol details all the formatting and code conversion necessary to make information from the sending computer intelligible in a such a way that the receiving computer can act on it. Example: Supplying graphics information in a form that the receiving computer can display, the SYLK data file format. Analogy: A lawyer sending a letter to a client that explains a court decision in plain English.

The application layer. This layer ensures that the information is sent in a form that can be used directly by an application program running on the receiving computer. Example: Moving a Multiplan file from an IBM PC to a Macintosh. Analogy: A lawyer sending a legal analysis of a court decision to another lawyer.

Many computer practices cut across several types of protocol, so things are not quite as tidy as this list implies. Nevertheless, we will use these concepts to guide us through the quagmire.

For your computer and another to communicate at all, you and the other user must agree on the physical and data-link protocol layers; for the computers to communicate smoothly, you must agree on the transport and session layers; and for integrated operations, you must agree on the presentation or application layers. There is no shortage of protocols at each of the seven layers—and the problem is that they're all different.

THE PHYSICAL LAYER: CONNECTING THE HARDWARE

For the physical connection, the choices include:

Direct connection, using a simple set of wires tying two computers together via their serial ports. The wire is sometimes called a null-modem because it replaces a pair of modems. See Chapter 29 for a wiring diagram of Macintosh-to-Macintosh and Macintosh-to-other-computer null-modems.

Connection via a local area network (LAN), an electrical connection that links many computers and accessories, such as printers and high-capacity hard-disk drives. Unfortunately, two devices that can attach to the same physical network may not be able to talk to each other unless the upper-layer software protocols permit. The better networks accept a variety of computers; the restrictive ones accept only a single brand or model.

AppleTalk is a simple, low-cost network; many other types of network are available. Some can move data much faster than AppleTalk, although the Macintosh may not take full advantage of the speed because of hardware restrictions.

Connection via telephone lines. Two computers, each using a modem, can communicate through the phone lines. To accomplish this, both computers must be ready at the same time; one then calls the other. Depending on the software and hardware, an operator may not need to be at the answering computer, or even at the sending computer; everything can be run by a timer under software control.

The telephone line may be:

- The same one we use for routine conversation; the quality of these voice-grade, dial-up switched lines varies greatly and appears to be getting worse since the deregulation of telecommunications.
- Leased or dedicated lines with higher bandwidth and lower noise. Because these lines are tested for performance and sometimes have permanent electrical compensation added, they are also called conditioned lines.
- Via cellular radio from a moving car. This connection will need a robust error-detection and correction scheme.
- Air-to-ground and ship-to-shore radio links. These connections are too unreliable for any but the most determined users with the best error-correction protocols.

- Switched data lines (Data-phone Digital Service or DDS) that you can dial up much like present voice-grade lines. Available now, these lines will handle 56 kilobits, or 7 kilobytes, per second. The forthcoming Integrated Services Digital Network (ISDN) supports two 64-kilobit-per-second channels of either digitized voice or data.

Connection via a third computer. When connection between two computers proves too elusive or schedules don't match, then an indirect method may work. The first micro communicates information to a third computer; the second retrieves the information. Most electronic mail services follow this scheme.

By using an intermediate computer for storage, the two micros don't have to be available at the same time. The third computer is usually a large one, often able to handle multiple protocols or at least a well-defined protocol. Getting a micro to communicate with such a large computer can be easier than getting two micros to talk directly with each other. Data formatting problems generally restrict this method to text, although you can code any file to a form that looks like text as far as the third computer is concerned.

Radio transmission. On some frequencies, the FCC authorizes data communications by radio amateurs. The reliability varies tremendously with location, time, and frequency.

THE DATA-LINK LAYER: CODING THE SIGNAL

The form taken by the signal passing through the physical link depends on both hardware and software. The most common form is asynchronous ASCII communication: Each character is sent separately, preceded by a start bit and followed by one or two stop bits and possibly an error-detection bit. Because the receiver can always tell from the start and stop bits when a character starts and when it is complete, the sending and receiving computers do not have to be locked to the same clock—they are asynchronous, with variable timing between characters. They must, however, work at the same speed.

Putting in all the start and stop bits takes time. If the sending computer transmits not just a character at a time but a block of characters (typically 256) at a time, the transmission rate improves considerably. The communications line carries a clock signal whenever it is not carrying data. To transmit a block, the sending computer must put synchronizing information at the beginning of each one. The receiving computer uses the synchronizing information to measure off each character in the block. Through the synchronizing information, the two computers operate synchronously, with fixed timing between characters and variable timing between blocks. If you use synchronous communications while typing, and you are a slow typist, there may be only one character in a block; the rest of the block simply marks time. In some systems the block length is variable, a valuable feature for interactive communications (where short blocks speed things up) and for coping with noisy transmission lines (short blocks help with intermittent noise; normally, long blocks are more efficient because the overhead is relatively smaller).

Synchronous communications are used mainly with mainframe computers and computer networks, such as AppleTalk. I'll come back to synchronous communications later and concentrate now on the more common asynchronous form.

Asynchronous Communications

The quickest way to sort out data-link layer problems is to find someone else who has already used a Mac to communicate with another computer and do the same thing. If you're on your own, this section covers each step. It doesn't matter whether the other computer is large or small; the same principles apply.

First, you need communications software at each end. If possible, get software for each computer from the same company. Although the programs themselves may be different in order to run on different computers, they are more likely to be compatible at several levels, sometimes as far as the session-layer protocol, provided that both programs are set up with the same parameters. If you cannot get such software, then you must configure two different communications programs to use the same data-link protocol.

Some programs only send and receive with a special protocol operating at the transport-layer level. These will only work if the other machine is equipped for the same protocol.

The protocol may restrict the kinds of information you can send. Between different types of computers, the transmitted information is most often ASCII text. Although the strict definition of an ASCII code includes all possible 7-bit characters, when used in communications, an "ASCII file" normally means only printing characters: letters, numbers, and punctuation, plus tab, carriage return, line feed, and form feed. The remaining characters, including most control characters, are normally non-printing. Many communications programs transmit some control characters; Control-g is useful because it will ring a bell on most terminals (Command-g on the MacTerminal program from Apple). Other control characters, such as the end-of-file marker, are usually removed before transmission.

Even if all you want to do is send an ASCII file, both machines must agree at the data-link level. The hardware determines some parameters, the software determines others; still other parameters can be set by either. The most important parameter is the speed.

Speed

Baud is a term from 19th-century telegraphy that measures how fast information is sent. By the strict definition, one baud is one signal event or modulation change per second (a "symbol" in communications theory). At 300 baud, one baud—one signal event—corresponds to one bit per second; at higher speeds, one signal event codes two, four, or more bits. A "1200-baud" modem actually sends 600 events per second; each event codes two bits.

Although common usage says "baud" when the correct term is bits per second, or bps, I follow strict usage here. The table later in this chapter gives, among other things, the data rate and the true baud rate for all common modem types. Depending on the exact data-link layer protocol, 300-bps transmission carries 30 to 33 characters per second and 1200-bps carries 120 to 132 characters per second. By coincidence, the number of words per minute is about the same as the number of bits per second.

Although the bit rate can be continuously varied in some cases, the industry has settled on several standard speeds and protocols. The speeds and protocols have been largely set by either Bell Telephone Laboratories (North America) or the International Telegraph and Telephone Consultative Committee (CCITT, first in Europe and Japan, now worldwide). If you use modems to communicate, they must operate at a common speed and protocol. For computers wired together directly, you can select the highest speed permitted by the equipment and the length of the wire; the longer the wire, the slower the speed. If in doubt, start at a slow speed and increase it until you encounter errors; then back off one speed step. (If you start at a high speed, you can't always tell if the problem is excessive speed or some other problem.)

Low-speed modems

75 to 110 bps. Used by teletypes, Telex machines, and other systems supporting older printers, and some radio applications. Also used for TDD communications by the hearing-impaired, with Baudot code instead of ASCII; you'll need a translation program and a 110-bps modem to connect to TDD devices. The 5-bit Baudot code has uppercase characters only.

300 bps. The traditional home-computer speed. At 25 to 30 characters per second these modems are slow, taking a minute to fill a Mac screen with characters. Most people can read text comfortably at this speed (up to 300 words per minute). Now almost obsolescent, 300 bps is giving way to 1200-bps modems. The standard protocol for 0- to 300-bps communications is Bell 103 in North America, CCITT V.21 in Europe and Japan; they are not compatible.

1200 bps. Most common in business and increasingly common with home computers. Transmission at 1200 bps is too fast for most people to read—a Mac screen fills in 15 seconds—but it is effective for scanning. If you're using a commercial database service, don't read everything while you are connected; save everything to disk and read it later; it's much cheaper.

The standard 1200-bps protocols are Bell 212A in North America and CCITT V.22 in Europe and Japan; the two are not compatible. The Racal Vadic 3400 protocol is still occasionally used, particularly by computing centers; most modems that use the Racal Vadic form can also use Bell 212A. These three protocols are full-duplex and work on ordinary phone lines. Avoid products that use Bell 202, a lower-cost, half-duplex protocol. (Full-duplex is simultaneous, two-way communications; half-duplex is one way at a time but the direction can be changed. Simplex is one way—and one direction—only.)

Medium-speed modems

2400 bps. The CCITT V.22 bis protocol is a new worldwide standard. It is full-duplex over dial-up phone lines, and it is adaptive, adjusting both sending and receiving characteristics to the line conditions when starting communications. The V.22 bis protocol is complex, requiring much cleaner, noise-free telephone lines than 1200 bps does; many long-distance phone lines aren't good enough.

V.22 bis modems are supposed to fall back to the 1200-bps V.22 protocol if the line is too noisy, but some units don't, or they fall back to the incompatible 212A protocol instead. If you set out to connect at 1200 bps with a V.22 bis modem, you must determine whether it is set for 212A or V.22. When communicating with the same model V.22 bis modem, you should not have these concerns—if you are on the same continent. But because of differences in European and American telephone practices, there are several potential problems when communicating internationally with modems that nominally follow V.22 bis. The European answer tone (2100 Hz) is different from the American one (2225 Hz); calls from America to Europe generally work, but the reverse isn't always true.

These problems arise from the way the standards were set. Bell Telephone actually built and sold a model 212A modem at a time when it controlled the American telecommunications network. Companies could buy it and check their own unit against it. The CCITT, on the other hand, is a voluntary body that can only make recommendations. There is no actual V.22 bis modem, only a pile of paper that manufacturers can and do interpret in their own ways.

The CCITT V.26 ter protocol also runs at 2400 bps. It is technically superior to V.22 bis and can cope with noisier telephone lines. It is also much more expensive, using echo-canceling rather than frequency-division multiplexing for full-duplex operation. V.26 ter will increase the chaos among 2400-bps modems, but V.26 ter units are much more compatible with the higher-speed V.32 4800- and 9600-bps modems and could be a better investment in the long run.

4800 to 9600 bps. These higher speeds have generally been used for synchronous communications with mainframe computers, but can be used asynchronously as well. The newer CCITT V.32 protocols are full duplex with echo-canceling on dial-up lines. Group III digital facsimile machines use half-duplex 9600-bps modems with automatic fall-back to lower speeds for noisy phone lines.

Table 25-1 gives characteristics of all standard modem configurations up to 9600 bps. Systems that are half-duplex with a two-wire telephone line can often be turned into full-duplex with four wires or two telephone lines. Leased and dial-up lines are to some extent interchangeable; entries give the common usage. There are also many other proprietary protocols used by a single manufacturer and suitable only for closed communications within an organization. Outside of North America, Europe, and Japan, the protocols in use depend mostly on who has the greatest economic impact. Many countries still officially ban any connections to the phone except by the central telephone company. Telephone-line quality varies widely; some phone systems are unusable by modems at any speed.

High-speed modems

Over 9600 bps. The new Telebit modems (also sold by DCA as the Fastlink) operate at over 10,000 bps on ordinary dial-up phone lines. The ingenious Telebit technology, using up to 512 carriers simultaneously with inherent error correction, could leapfrog the 2400-, 4800-, and 9600-bps protocols. Traditional over-9600-bps modems are used for specialized applications. Some short-haul models can operate at a megabit per second. Special digital communications links may use microwave or television circuits. Few computer users work directly with this class of modem.

TABLE 25-1. MODEM PROTOCOLS

Protocol	*Data rate* (bits/sec)	*True baud rate* (signal events/sec)	*Half/Full Duplex*	*Duplex type*	*Async/ Sync*	*Dial-up/ Leased lines*	*Area*
Bell 103	300	300	F	FDM	A	D	North America
CCITT V.21	200-300	200-300	F	FDM	A	D	Europe/Japan
Bell 202	1200	300	H	-	A	D	North America
Bell 212	1200	600	F	FDM	A/S	D	North America
CCITT V.22	1200	600	F	FDM	A/S	D	Europe/Japan
Bell 201	2400	1200	H	-	S	D/L	North America
CCITT V.22 bis	2400	600	F	FDM	A/S	D	All
CCITT V.26 ter	2400	1200	F	ECT	A/S	D	All
Bell 208	4800	1600	H/F (2/4 wire)	-	S	D/L	North America
CCITT V.32	4800-9600	2400	F	ECT	A/S	D	All
Bell 209	9600	2400	H/F (2/4 wire)	-	S	L	North America
CCITT V.29	9600	2400	H/F (2/4 wire)	-	S	D/L	All

FDM: frequency-division multiplexing   ECT: Echo-canceling technique (see text for explanation)

Noise on the phone line

Noise limits the capacity of any communications channel. The standard pre-divestiture AT&T phone line, measured under modem communications conditions, had a 24 db signal-to-noise ratio (S/N) on long-distance lines, a 27 db S/N for local calls. Table 25-2 lists the S/N required for different modem protocols. The figures were compiled by Ken Krechmer (Action Consulting, Palo Alto, CA); see his article in *Data Communications,* April 1985.

TABLE 25-2. SIGNAL TO NOISE RATIOS

Speed	*Protocol*	*Best-case S/N (db)*	*Typical S/N (db)*	*Safety margin w/typical S/N (db)*
300 bps	103	4	6	16
1200 bps	212A/V.22	7	14	10
2400 bps	V.22 bis	14	21	3
2400 bps	V.26 ter	10	17	7
4800 bps	V.32	16	21	3
9600 bps	V.32	19	—not available—	—not available—

The best-case S/N gives the theoretical performance of the protocol, while the typical S/N is the measured performance for commercial products. The safety margin is calculated for a pre-divestiture long-distance line. The best industrial-grade modems perform distinctly better on noisy lines than mass-produced consumer units, but on good telephone lines there is no functional difference. Unlike analog signals (such as hi-fi sound), digital communications are not affected by noise as long as the bits are demodulated accurately.

The real situation may be worse than the table shows, however. Simple signal-to-noise ratio figures do not include many problems, such as pulse noise. Furthermore, consumer modem quality varies widely; some cheaper designs have much worse S/N figures than the average and can only work on local telephone lines at 1200 bps. Some modems contain design errors that degrade performance when they communicate with a different modem model.

The S/N ratios given here measure one-way links only. Working modems in full-duplex links need an additional 3 db, effectively wiping out V.22 bis's safety margin. The competition to offer telephone services at lower rates has already degraded the signal quality on many long-distance lines, and the situation will probably get worse. With the faster modems, use a local call whenever possible; call a nearby gateway for a dial-up service rather than try to connect directly by long-distance lines. (The signal-to-noise ratio for most other computer components is irrelevant under normal conditions.)

Measuring modem performance is a complex topic. Few manufacturers have complete facilities, and at present no computer magazine has the tools. As a result, articles about modems almost always focus on operational features rather than technical performance.

You have only two choices with noisy phone lines: switch to a slower speed or adopt error correction (described shortly).

Other parameters

After setting matching speeds, both sides must also agree on these other data-link layer parameters:

Character width or data bits. This is the number of bits in a character, either seven or eight. If you are communicating with another Macintosh, eight bits can carry the entire character set; seven bits cannot. For ordinary text communications, the number doesn't matter as long as there is agreement.

Stop bits. Either one or two bits signal the end of a character. This setting frequently doesn't matter; after counting off the data bits, a computer can ignore everything until the next start bit.

Parity. Even, odd, or none. A single parity bit lets the systems determine if there is a one-bit error in the received signal. For even parity, the computer adds up all the bits in the character. If the sum is even, the parity bit is 0; if odd, the parity bit is 1. Either way, the final sum is always even. If a one-bit (or three-bit or other odd-numbered error occurs, the sum will no longer be even. A two-bit error will not be detected since the sum remains even. Odd parity works the same way except the sum is always odd. Some systems specify Mark or Space, variations on no parity.

What the receiving system does upon encountering a parity error depends on the software. The parity bit is simply an error flag letting programs signal for retransmission—if the sending computer is set up to recognize the signal.

Duplex. At 300 and 1200 bps, most phone links are full-duplex—information can go both ways at the same time. With a full-duplex link, the two computers use different frequency bands (frequency-division multiplexing). So that both don't try to use the same band, one is set to the originate band, the other to the answer band. Which is which doesn't matter as long as both sides agree. With all modems and software, the calling party is set to originate. At the higher speeds, the telephone line doesn't have enough bandwidth for reliable transmission with two distinct frequency bands, so more advanced modems use echo canceling, a technique that allows both modems to use the entire frequency band. Each modem senses the effects of its own transmission and cancels them, leaving the incoming signal free of interference.

With a half-duplex link, information travels only one way at any given time. The line can be reversed with signaling codes, but frequent reversing slows the transfer rate considerably. If the application calls mainly for one-way transfers, there is no speed penalty.

Echoplex. When you are communicating with a remote computer, how do you know that your characters are getting through? With full-duplex and echoplex parameters, each character you send out is returned ("echoed") by the distant computer. Echoing is only practical with a full-duplex protocol (it takes too long in half-duplex), but not all computers operating full-duplex will echo. (This echo is a completely different issue from echo-canceling.)

If your echoed characters are occasionally garbled, the line is noisy. If the remote machine does not echo, your computer must generate the characters on your screen ("local echo"). If all your characters appear doubled, turn off local echo; if you don't see anything, turn on local echo. Some software and publications confuse duplex with echo.

For a second group of data-link layer parameters, communications will be more efficient if sender and receiver agree, but if the messages are sufficiently short or the receiver is willing to rework the received information later, complete agreement may not be absolutely necessary. These parameters include the following:

Handshake. The sending machine sometimes transmits faster than the receiving machine can receive. Handshaking can overcome this problem. The receiving machine first sends a signal (usually DC3 or XOFF, Control/Command-s) to the sending machine, which then stops until it receives another signal (usually DC1 or XON, Control/Command-q). Handshakes help when the receiver must write the transmitted information to disk. Unfortunately, many senders do not recognize handshakes. (On short communications lines, such as one between computer and printer, a separate electrical circuit in the connecting cable may carry the handshake signal; this is called "hardware handshake.")

Line ends. The Macintosh, in its internal files, uses a carriage return to indicate a move back to the left margin and a one-line paper advance. Some computers distinguish between a carriage return and a line feed and require both. Communications software for Mac should add line feeds to carriage returns if necessary and strip incoming line feeds when accompanied by a carriage return. If these steps aren't taken, you may see double line feeds in some cases, and the other party may see all your lines overwriting each other.

Filters and pauses. For some systems, you must remove certain characters or else you will see spurious characters on your screen. Some receiving computers require a short pause at the end of each line to allow them to process the line.

THE NETWORK LAYER: ADDRESSING THE MESSAGE

This protocol layer generally affects only local area networks, specifying the addressing scheme so a message goes only to the intended recipient. The layer is built into the network hardware and software; for example, it is part of the AppleTalk software.

THE TRANSPORT LAYER: ERROR-CORRECTION AND VERIFICATION PROTOCOLS

Protocols that send standard ASCII files in a continuous stream cannot send every possible bit combination because some codes must be reserved for marking the end of a file or signaling for the handshake. Transport-layer protocols get around this difficulty by sending information in blocks of a fixed size (called protocol data units in ISO jargon). Since sending and receiving computers both know exactly how many bits will be in a block, all bit combinations are permissible. The longer the file, the more blocks it uses.

Files that can contain any combination of bits are called binary files, because they are just a sequence of binary digits. Programs that run directly on a computer are binary, as are many types of data files.

For error detection and correction, the sending machine calculates an error-detection code from the data in the block and adds the code to the end of the block. The receiving machine takes each block, independently calculates the error code, and compares it with the code sent. If the codes don't match, the receiver requests the block again. Because of this additional information, the transfer rate is slower than sending straight ASCII without error correction. In some cases, the error detection is done at the data-link layer and the correction is done at the transport layer.

Transport-layer protocols with error correction are sometimes called protocol transfers or verification protocols.

Error correction is always handy but not always essential. English text survives occasional typos. For financial data or programs, on the other hand, any error—even a single bit—can be serious, so error detection is essential.

The computer industry has no standard transport-layer protocol; several dozen are in use, many for specific brands of hardware or software. Most work in a primitive way. Before receiving information, the receiving machine must open a file and let the sending machine know that it is ready to receive. The sender transmits the information, and the receiver closes the file.

The transport-layer protocol best known among computer hobbyists has been the Xmodem protocol used by many microcomputer communications programs, for many different computer models. Xmodem is effective (although it does not catch all errors) but works only at the transport layer; it needs modification for more sophisticated use. Xmodem is not particularly efficient, for it works in a half-duplex fashion: After sending each block, the sender waits for acknowledgment; on satellite transmissions, this takes a long time—80 percent of the time at 2400 bps. In faster protocols the sender transmits continuously, getting acknowledgments from the receiver while transmitting. When errors are detected, the sender can resend a block without pause. Xmodem's fixed block size makes it unsuitable for interactive communications when many messages are a single character. The newer protocols have more robust error detection.

The Kermit error-correction protocol developed at Columbia University also works at the transport-layer level. Because it was designed for universal use on any type of computer, it is inherently inefficient. The Kermit protocol is available for the Macintosh and will be used, at least initially, to connect systems in universities.

When a Mac uses a transport-layer protocol, the communications program usually sends only the data fork of a file. The MacBinary format (which overlaps into the session-layer protocol) specifies a way to send and store a complete Macintosh file, including header and both data and resource forks, on other computers. Such files can be used for electronic mail and general storage.

THE SESSION LAYER: COORDINATING ACTION

At the session layer, two computers can automatically take actions such as opening and closing files on command from one or the other. Thus the transmission not only moves the contents of a file, it essentially moves the file folder itself, neatly labeled and ready for use. Most local area network protocols work at this level.

With a session-layer protocol, the file type, icon, and other attributes are sent with the file contents when you are transmitting between Macintoshes. If the communications involve several computer types, session-layer protocols can keep track of which computers can understand which file contents—a Macintosh might receive a file with an icon, while an IBM PC might get just a directory name.

The Macintosh Xmodem transfer supported by MacTerminal and several other Mac communications programs works at the session layer, for it transfers a complete file—header and both data and resource forks. This Xmodem variant is not directly compatible with the traditional Xmodem on other computers.

For communications between a Macintosh and an IBM PC, MacLink (from DataViz, Norwalk, CT) operates at the session layer; you can see and change IBM disk directories and subdirectories from the Macintosh. MacLink also performs some file-format conversions (see Chapter 27).

At the session layer, however, compatibility in the computer industry pretty much falls apart. For dial-up lines, Microcom advocates its proprietary Microcom Networking Protocol (MNP) and Tymnet recommends its X.PC protocol. Both use much more thorough error checking than Xmodem, both can work synchronously to save time by eliminating start and stop bits (provided the modem can cope), and both have variable block size.

MNP is better for microcomputer use, although it is best supported by hardware in the modem. MNP automatically checks to see if the receiving modem understands MNP, and if the receiving computer has enough space on its disk drive to store the file; in each case it can take the appropriate action. X.PC is a subset of a protocol called X.25 and is capable of simultaneous communications sessions on the same line with a single remote computer (usually irrelevant for micros); it can be supported easily with software alone. X.32, which is an expanded form of X.25 for dial-up communications, may become widespread in the future.

No session-layer protocol seems assured of universal success and no single protocol will suffice for all possible communications. The intelligent communications software will select or detect the most appropriate protocol so the user need not fuss with the details.

THE PRESENTATION AND APPLICATION LAYERS: THE HIGHEST LEVELS

At the presentation level, the receiving computer can make use of the incoming information in a standard form. Presentation-layer protocols range across many applications.

- Graphics protocols use a standard way to write on the screen. The North American Presentation Level Protocol Syntax specifies a coding system for screen graphics, used in the videotext industry.
- A few data formats permit moving information between different programs and different computers.

At the presentation layer, software must still convert information before use, and a widely used protocol may not be able to take advantage of all the features of any specific computer.

At the application layer, the file is moved between two computers that can use the information directly. Until there are more standards, this level is achieved mainly when moving information between similar computers.

For specific information on moving files from another computer to the Macintosh, see Chapter 27.

COPING WITH THE DATA FLOW: CONCENTRATORS AND MULTIPLEXERS

Macintosh and other graphics-driven microcomputers create and manipulate far more information than earlier microcomputers. Files include not just ASCII text, but typefaces, icons, graphs, and drawings as well. These large files gobble up time in transmission and run up the phone bill.

If you frequently send large quantities of information, you should consider several ways to increase the effective transmission rate.

Higher-speed modems. Since higher-speed modems may not operate reliably on noisy dial-up lines, you may want to switch to leased lines. If so, the telephone line charges go up to about $1.50 per mile per month.

Data concentrators. An English-language text file contains much redundant information. A trivial example is the letter Q, which is always followed by a U. If sender and receiver agree, the U can be deleted from the transmission, saving time; the receiver inserts a U after every received Q.

The widely used Huffman coding algorithm compresses English text to about half the original length. A complementary algorithm restores the original text. You can use the algorithms manually, or use software that automatically performs compression and expansion; such software usually provides error correction as well (transport-layer level). A 1200-bps line with compression functionally runs at twice the speed. The compression ratio depends on the information; the more powerful compression techniques work with a variety of data, not just English text.

Multiplexers. A MUX (multiplexer) takes the incoming data stream from two or more computers and puts them on a single telephone line. Often a MUX includes a built-in data concentrator. Both sides of a communications link must have matching MUXs. Prices range from $1200 up. A combination MUX and concentrator can run two channels at effectively 2400 bits per second with a single 1200-bps synchronous modem. Concentrators and MUXs work at the transport layer; many can handle text only.

COMMUNICATIONS WITH OTHER COMPUTERS

When two computers communicate, their relative sizes often affect terminology more than the technology.

Nearly all computers consist of a processing unit and a keyboard/display unit. A microcomputer contains, in the same package, a processor and a keyboard/display unit. A mainframe computer is a large machine with very fast processing, large-capacity disk drives, and many megabytes of memory. Its processing unit fills big boxes installed inside air-conditioned rooms; its keyboard/display units, or terminals, may be scattered all over. The traditional, or dumb, terminal does not contain any processing power and can operate only when connected to its large computer host.

To a large computer, Mac is just another dumb terminal. But since Mac has processing and storage capabilities, it can act as an intelligent terminal, manipulating information before sending it on to the mainframe and storing the mainframe's responses for later use. For example, you can use MacWrite to prepare text before sending it to the mainframe (save with the Text Only option); you can also feed information from the mainframe into Multiplan for further analysis.

Terminals come in many forms, and suitable software will turn Mac into most of the popular types. The differences between terminal types involve many details, such as the code to erase a line on the screen. Although some terminal configurations truly offer special advantages, most came about because of arbitrary design decisions. Many communications programs emulate the Digital Equipment Corporation's VT52 and VT100 terminals. Other companies have programs to emulate the DEC VT125, Tektronix graphics terminals, and many others. Some of these programs are discussed in Chapter 8.

With so many terminal types, some systems fall back on the simplest one: an electronic version of the old mechanical teletype. A teletype has minimal features—just the ability to print along a line and advance the paper. It won't even erase. In a rare display of nostalgia, many flashy terminals, including Mac, can emulate a teletype.

When you buy software for Mac to emulate a terminal, make sure it emulates an intelligent terminal rather than a dumb one. Look for several key features:

- Disk storage and recording. The program should let you create information with Mac software, store it to disk, and then transfer the disk file to the mainframe, as well as store returning information on disk.

- Ability to store communications parameters on disk so you can set them once and recall them.
- Ability to suspend communications without disconnecting from the remote computer. This way you can use a Macintosh program for some task and return to the other computer without having to reestablish a connection.

SYNCHRONOUS COMMUNICATIONS

Earlier in this chapter we saw that synchronous communications work by sending information a block at a time, with timing information so the receiving computer can extract the characters from the block. There are, naturally, several variants of synchronous communications. The various physical-layer protocols can be shared by most forms, but the data-link layer changes. In its various forms, a block of data is also called a frame or a packet. Packet-switching systems are telecommunications devices that deal with blocks, rather than voice or asynchronous data.

Microcomputers have generally used asynchronous communications because it's easier, and the performance penalties aren't too serious for small quantities of data. Synchronous links are mainly used by large computers. For best performance, a microcomputer connected to a mainframe should use a synchronous link. The physical link between the mainframe and the micro may be a wire or a modem. Many modems made for microcomputer use, including the Hayes and Apple 1200-bps modems, will not handle synchronous links.

The conversion from asynchronous to synchronous communications can be performed in several ways. Although the Macintosh is normally asynchronous at its serial ports, the ports will accept external timing signals on pin 7 to clock data in and out and thus support common synchronous protocols. The Macintosh can therefore emulate the terminals that the mainframe software expects to see at the end of each cable. Several companies sell synchronous protocol converters for the Macintosh, including Apple (AppleLine), Avatar, and Winterhalter. Most units emulate the IBM 3278 terminal.

If several different Macs need to communicate only occasionally with a mainframe, a single converter can be switched among them. Some converters are cluster controllers, servicing several microcomputers or terminals simultaneously under software control. Or the mainframe may have ports adapted to asynchronous operation.

Virtually all microcomputers, including the IBM PCs, use ASCII. But many large computers, particularly IBM models and copies of IBM models, don't use ASCII for coding characters. Instead they use EBCDIC (Expanded Binary Coded Decimal Interchange Code); it's similar to ASCII and the conversion software simply looks up each incoming character in a table and sends out the translated code. The supporting software for all the protocol converters will perform ASCII to EBCDIC conversions.

System Network Architecture

System Network Architecture (SNA) is IBM's umbrella architecture for its present communications offerings. The communications protocol within SNA transmits data in blocks; it is known as Synchronous Data Link Control (SDLC), a version of the international standard High-Level Data Link Control (HDLC).

The IBM computers that work with SNA include the 370, 43XX, 303X, 308X, 3090, and 8800 series, as well as the System 32, 34, 38, and Series/1.

The most widespread type of IBM synchronous terminal is the 3270 series of interactive products, so most terminal emulators, including the AppleLine, emulate such terminals. Fewer microcomputer products now emulate the earlier 3780 batch-communications devices, such as the 2780 terminal.

Help with a Mainframe Connection

Getting micro-mainframe links to work successfully is often an involved process, but in most cases, an organization big enough to have a mainframe is also big enough to have a data-processing department that can help you get started.

Chapter 26
Local Area Networks and AppleTalk

Personal computers were developed as a product for individuals; they were bought individually and used individually. Communication with the outside world was limited to a modem or swapping disks. As micros gained power and became more common in companies, users began needing to share information quickly.

The first practical way to share information was to connect the micros as terminals to a large computer. The large computer could store files that everyone could share. But not everyone has a large computer, and if they did, all that expensive computing power could be put to much more important uses than shuttling files between micros.

Enter the local area network, or LAN. Networks consist of nodes connected together. A node can be a computer, a printer, a disk drive, or any other device capable of sending and/or receiving information. In most networks, any node can send a message to any other node; nodes not involved with a particular message ignore it. The computers in a network generally have equal access to network devices, so they can share disk storage, printers, and other resources.

Early developers envisioned networks of diskless microcomputers sharing a central hard disk drive that would provide all storage functions. This was supposed to lower costs, since no computer would need its own disk drive. Things have turned out differently; putting a hard disk drive on every computer can now be cheaper than the combined costs of installing network wiring and a shared hard disk. But for electronic mail services, or when files or laser printers must frequently be shared, networks make sense.

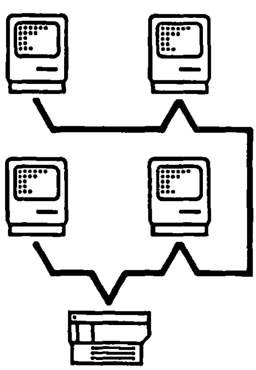
In some ways, networks can be a step backward, for they represent a return to central control. With a freestanding microcomputer, you are master of your own destiny. A network, on the other hand, must have a system administrator who sets up and manages it—and decides when to shut down a shared device. A breakdown of a critical shared component, such as a disk drive, could paralyze many users, a key problem with central computers. Careful system management, including frequent file back-ups, will minimize such threats. Many networks can be disconnected easily, restoring freestanding operations when necessary.

NETWORKS IN GENERAL

Networks come in a tremendous range of capabilities and prices. I will begin with a general introduction and then concentrate on the AppleTalk network.

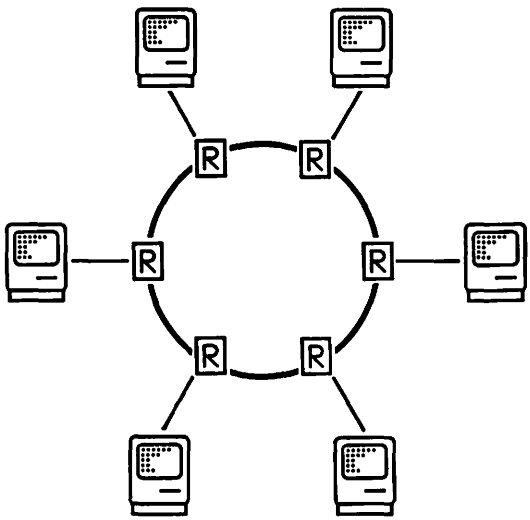
Network Topology

Connections between nodes can be organized in several ways.

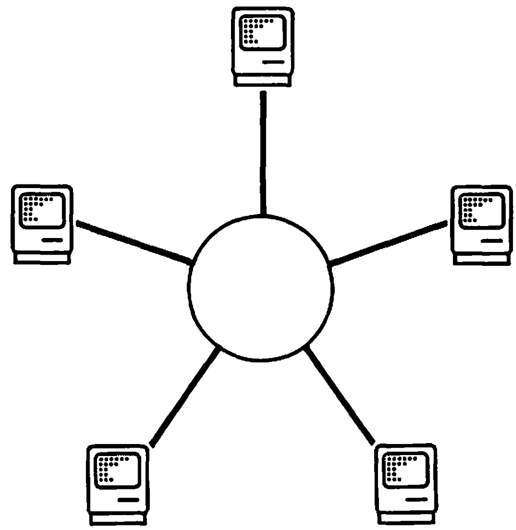


Bus network

- *Bus networks.* These have a single wire connecting each node; the wire does not close upon itself. Each node has a unique address and monitors the network for information addressed to it. In the most common designs, each connection to the network is a passive circuit so that a node failure does not disrupt the rest of the network. Adding nodes is simply a matter of tapping into the cable. Most commercial networks, including AppleTalk, Ethernet, Sytek (the IBM PC network), and WangNet, use the bus layout.
- *Ring networks.* These use a closed-loop cable; a repeater for each node relays information around the ring and picks off information addressed to its particular node. Ring networks can run over great distances because the signal is regenerated by each repeater, but if one repeater fails, the network fails as well. Adding a node requires breaking the connection between two nodes and inserting a repeater. Ring networks are used in some universities and for specialized applications requiring high performance.



Ring network; [R] *repeater*



Star network

- *Star networks.* These use a central hub that handles all traffic. The traditional mainframe computer serving multiple terminals acts like the hub of a star network. The hub is critical; failure means the network stops. Wiring costs are usually high, since every node must be connected directly to the hub. A network can also use the private branch exchange, or PBX, the telephone system within an organization. PBX networks follow the star layout with the central switch as the hub. Wiring costs may be considered relatively low, since every office is wired for a telephone anyway.

Information Flow

Within the network cabling, the information can travel in several forms. One essential distinction is whether messages are separated by time or frequency, or by cable. Speeds quoted here are the often-misleading raw data-transfer rates (discussed later in this chapter).

- *Baseband.* A baseband network carries a single channel of information at a time. If the network is busy, each node waits for the channel to clear before sending information; messages are separated by time. Peak speeds range from slow to medium—about 50 kilobits to 20 megabits per second. Baseband networks include Ethernet and AppleTalk.

- *Broadband.* A broadband network can carry multiple independent channels, much like cable television; messages can be separated by frequency. A channel may carry data, voice, or even video. Broadband can be high speed, well over 20 megabits per second, and is generally expensive; both the cabling and the network interfaces cost more than they do for bus-based designs. The most elaborate versions use fiber optics as the communications pathway. WangNet is a broadband network.
- *PBX-based.* Networks based on older PBXs must use analog signaling and are slow, typically limited to 9600 bits per second. New digital PBXs usually carry information at 56 kilobits per second, the data rate of a digitized voice channel. Because a PBX network handles many messages simultaneously, the aggregate communications rate can be high. Within a particular cable, the signal is usually baseband, but time-sharing is unnecessary because a PBX network separates messages physically with multiple cables.

Compatibility Within a Network

Two nodes on a network can talk to each other only if the data are in a mutually understood form. The basic network protocol usually goes up only through the network layer, but some networks have failed to achieve even this degree of compatibility because the various network hardware and software vendors do not agree. For useful information exchange, software must provide compatibility through the presentation and application layers (see Chapter 25 for a general discussion, Chapter 27 for some specific cases).

Gateways Between Networks

Most networks can handle a limited number of nodes and a limited length of cable. For efficiency, networks should usually have far fewer nodes than their capacity because performance can suffer greatly when traffic approaches saturation. The distance limit is often set by the electrical properties of the cable. Networks that use repeaters may not have a distance limit, since the signal is regenerated each time.

To circumvent these limits, a gateway, or network bridge, can connect two or more networks. A gateway is a node on both networks that accepts messages from one and relays them to the second. Gateways must usually provide addressing services, so nodes on one network can address specific nodes on the other. With suitable design, a gateway can connect two local area networks that are thousands of miles apart.

A second type of gateway connects two dissimilar networks, performing protocol conversion as required.

A backbone network is generally a high-performance network used for high-speed communications and gateway services among two or more lower-speed networks.

Network Speed and Efficiency

Network speed is a complex issue. The most commonly quoted speed figure is the raw data-transfer rate, the fastest rate that bits can move along the network cabling. But the information actually travels at that speed only in brief bursts. In real situations, the overhead associated with a message—address information, conversion of files into frames, message confirmation, and so on—takes up considerable time. On bus networks there is always dead time, even when several nodes want to send messages.

The true throughput speed—the rate at which useful information passes from node to node—is much less than the raw data rate, often as little as 20 or 30 percent; thus a "10-megabit-per-second" network may functionally pass only 2 to 3 megabits per second. As this is written, speed measures for AppleTalk were not available.

Some network designs become unstable with very heavy traffic; operation can even come to a complete stop. Careful planning is essential if a network must carry heavy traffic.

Messages and Speed

The common fixation on the raw data-transfer rate obscures the fact that the speed doesn't matter for many network applications. Fast networks are expensive, and many offices don't really do anything that would justify such an expense.

Activities where speed is usually not important:

- *Electronic mail.* Network speed is almost irrelevant; in most situations, it makes little difference whether the message gets to its destination in 0.1 second, 10 seconds, or even 10 minutes. For truly urgent messages you would probably call the person on the phone, anyway.

- *Sharing files.* Shared data files are most commonly treated like electronic mail, except complete data files instead of text messages are sent. Unless the file is unusually large, it will get to the recipient within a few seconds regardless of the network speed.
- *Printer information.* The throughput is more often limited by the printer speed than it is by the network speed, unless extremely complex graphics are involved.

Some network traffic is moderately speed sensitive:

- *Fully downloaded programs and data files.* If you work with software that is entirely resident in RAM (software that does not need to go back to the disk during operation), network speed is not always critical. The program will operate as fast as always, and any delays while reading the disk may not be too bothersome.

And some traffic is highly speed sensitive:

- *Disk-intensive programs and data files.* If the software must frequently go to the disk drive to read in more program code or data, network speed can be a factor, especially if heavy traffic forces additional delays.
- *Multiuser interactive software.* Some new software will let two or more users work on the same file simultaneously; several people might edit a memo or draw a diagram together. Everyone sees the same screen (although only one person at a time can make changes). Since all screen updates are sent via the network, rapid response helps greatly. Few such programs have appeared so far, and effective use may require changing working practices within a company.
- *Multiuser data bases.* When many people must have quick access to a complex, shared data base for transactions, network speed can be crucial. (Airline reservation systems are the extreme example, although they do not use standard network architecture.)

When a Network is Overloaded

There are three general solutions for an overburdened network:

- Break the network into two smaller ones, dividing the intensive users to cut down the traffic volume on each network. Use a gateway to maintain connections among all nodes.
- Give intensive users their own dedicated hard disks if they are taking up too much network time working with their own files on a shared disk; a dedicated hard disk works faster than any network.
- Reschedule the workload so that predictable periods of heavy usage don't overlap.

At this point, I will stop describing networks in general and deal mainly with AppleTalk.

APPLETALK

AppleTalk is a relatively simple, low-cost bus network that can tie up to 32 nodes together on a cable up to 1000 feet in length. The raw data-transfer rate is 230.4 kilobits (29 kilobytes) per second.

The physical connection between a Macintosh and AppleTalk consists of a small box containing an isolation transformer (to protect from electrical interference) and plugs for the network cables. AppleTalk uses a shielded, 78-ohm, twisted-pair cable, available in long rolls as well as short lengths. If you install the cable, install an AppleTalk connector at every likely location for a computer or other node even if you will not use the connector immediately. This strategy will make it easier to add nodes later. If you use continuous connecting cable, you will have to cut the cable and attach a connector.

An AppleTalk network can operate at peak speeds higher than 230.4 kbps, if you use special hardware that provides timing signals to each node. The AppleTalk interface within the Macintosh, however, is designed for the 230.4 kbps data rate, so further interface hardware will also be necessary.

Node Types

AppleTalk supports a number of different node types:

- *Computers.* AppleTalk can mix computer types; you can put Macintoshes, IBM PCs, and other computers on the same network and communicate between them. Tangent Technologies (Norcross, GA) was the first to offer an AppleTalk interface card for the IBM PC; several others will follow. All computers can share network resources, such as printers and disk drives, if they have suitable software added to their own operating systems. Of course, shared information must be commonly understood by all computers; simple text files will be understood by all, but an IBM PC will not be able to do anything with a MacPaint file unless it has a compatible graphics program. Eventually, sophisticated network software will perform automatic file conversions between dissimilar computers; see Chapter 27.

In network jargon, a device available to everyone on the network has the word *server* added to its name. Thus a *printer* is available only to an individual user, but a *printer server* is available to all:

- *Disk server.* A disk drive on the network can come in two forms, a disk server or a file server. The simpler disk server appears as a disk drive to network users; they can open the drive, examine directories, and read and write information. Often, disk servers use software that divides a single physical disk into several logical volumes that behave as if they were individual disk drives. In some designs, only one user can access a particular logical volume at a time; in others, several users can use a volume but only one user can access a particular file at a time.
- *File server.* The more sophisticated file server does all disk-server functions and more. Not only can several people use the same logical volume, but they can use the same file simultaneously with multiuser software. Multiuser programs employ record locking so that only one user can change a specific record at a time; until that person is finished, others are locked out although they may be able to see the record.

File servers are fairly complex devices that incorporate their own microprocessor and a multiuser operating system. The operating system and other details may be hidden from the users, who see only normal Macintosh operation. Some file servers may employ well-known operating systems, such as UNIX.

A network can have a single central file server or multiple file servers. Some network software will make all disk drives connected to any computer on the network available to all users. Although password protection may apply to specific disk drives, some thought should be given to privacy considerations.

- *Printer server.* A printer on the network is normally available to all users. Printing is a slow process, particularly for a long document; if the printer is busy, anyone else wanting to print has to wait. To eliminate the waiting, you can install several printer servers. Apple's printer-driver software can handle several LaserWriters at a time; you select the one you want to use. Or network software can store or spool printer information to a temporary disk file and automatically forward the data to the printer when it is free. Some complex file servers will incorporate printer ports for all common printer types as well as a print-spooling function.
- *Mainframe server.* Computers on the network can operate as intelligent terminals with a mainframe computer; a mainframe server performs protocol conversions and can handle several terminal sessions simultaneously.
- *Modem server.* In principle, a modem server would let network users—one at a time—share a single modem and telephone line to the outside world. In practice, this turns out to be difficult because standard communications links send characters at essentially random times. Such random timing is unsuited to networks, which are designed for sending blocks of information. The modem user would see erratic operation. As a result, few modem servers have ever been made. Since modem servers must have their own microprocessor and controlling software, they cost much more than a conventional modem.

- *Other devices.* Many peripheral devices are like a modem in that they are designed for a single user and work best when they have the full attention of the host computer. Thus, scanners and optical character readers usually don't work as network nodes on their own, but should instead be connected to a specific computer that can forward the results to the network.
- *Gateways.* Bridges in Apple's terminology, gateways connect separate AppleTalk networks and tie an AppleTalk network to other network types.

Network Addressing

Going from the highest level to the lowest, an AppleTalk address can have up to three levels: the network, the node, and the socket.

A network address is needed only when several networks are connected, in which case each network has its own address; up to 65,000 networks can be addressed.

Within a network, each node has a unique address. Some nodes have an address stored in non-erasable memory or on disk. If the node does not have a permanent address assignment, it gives itself a random address number when turned on. To ensure against duplicate numbers, the node then sends a message to its own address; if any other node accepts the message, the sending node changes its own number.

Users do not normally need to know the node numbers; they deal with familiar names—*Jane Doe* or *LaserWriter 2.* When you connect to the network, you give the AppleTalk software your name for storage on a disk. Thereafter, whenever you start a program that uses AppleTalk, your name is retrieved from disk. To send electronic mail, you get a listing of other users by their names; the software automatically attaches the correct node number to the message. Because the node numbers can change, the software polls all attached nodes to get a list of active users.

Within a node there can be multiple sockets. A socket is a logical concept, not a physical entity. Sockets generally specify particular application programs within the receiving microcomputer. Thus, a message can be sent to the electronic mail program and not to a data base. With the socket addresses, the node can distinguish an electronic mail message sent by another user from a data file sent by a file server.

Network Control and Protocols

Once connected to a network, how does a node know when it can talk? On AppleTalk and many other bus networks, there is no master controller; instead, every node can initiate a message on its own. The control comes from Carrier Sense Multiple Access with Collision Detection, or CSMA/CD. This mouthful describes a system with two simple rules:

- Any node currently using the network has priority. When a node needs to send information, it first looks at the network. If it senses a carrier, indicating that the network is in use, the node waits. If the network is not in use, the node sends the message.
- If two nodes start transmitting at the same time, there is a collision; both nodes then back off and wait a random time interval before trying again. Because of the independent random time delays, they won't collide on the second try.

If the network is free, a sending node first transmits a Request to Send message addressed to the receiving node. The receiver must acknowledge with a Clear to Send message within 200 milliseconds. If there is no acknowledgment the sender must try again later. Some network software may deal with any busy nodes in the manner of a printer spooler, automatically storing the message in a temporary disk file and forwarding it later without further user action. A message can also be broadcast, sent to all nodes on the network.

Frames

Information sent on the AppleTalk network travels in frames, or blocks, organized in the manner of SDLC/HDLC links (see Chapter 25); each frame consists of several components:

- *A synchronizing pulse.* This marks the beginning of a frame and is followed by two or more flag bytes.
- *The frame preamble.* This contains the addresses of the recipient and the source nodes and the type of information contained in the frame.

- *The data.* The data follows, up to 600 bytes in length. The data length is variable; the receiving node determines the length from the synchronizing pulses (alternatively, software can set a fixed length). Some frames, such as the Request to Send message, do not contain data.
- *The frame trailer.* This contains the Frame Check Sequence; sender and receiver independently calculate an error-correction value from the data in the frame. If the two calculated values do not match, the frame is discarded and the receiver asks for the frame to be resent.

Many messages are longer than 600 bytes, of course. Up to 12 frames can be linked together by a special identification code. If there are errors, the receiver need only ask for the specific faulty frames. Messages longer than 7200 bytes (12 × 600) must be broken into groups of frames. Breaking up long messages in this way prevents two nodes from hogging the network. In addition, the receiver needs only a modestly sized memory buffer for incoming information. AppleTalk can operate in the background so your work is not disturbed, at least for short messages; long messages may require some disruptive disk activity.

Some AppleTalk Products

Just about everyone who will sell AppleTalk systems will offer at least electronic-mail and disk-server functions, as well as printer control with spooling and conversion of ASCII text to PostScript for driving the LaserWriter with non-Macintosh micros. Most of the products described here are in pre-release form.

Several companies—including Banyan Systems Incorporated, Lutzky-Baird Associates, TouchStone, and Cadmus Computer Systems—use a UNIX-based computer as the centerpiece for a network. The UNIX computer functions as a network server, providing file facilities and electronic mail to Macintoshes and other microcomputers that are attached through an AppleTalk cable. To varying degrees, the designs try to hide the cryptic UNIX interface from the user. Some systems use UNIX completely in the background for file management; a user cannot deal with UNIX directly. Other systems have a mode that turns an attached micro into a UNIX terminal. The Cadmus product turns the Mac into a graphics workstation running on a UNIX host.

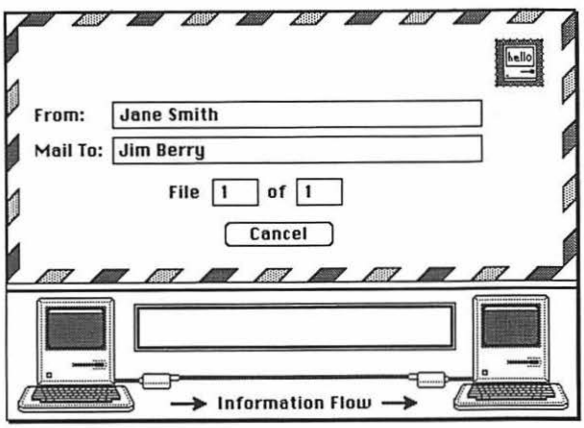
Sunol Systems offers its own disk servers for connecting several computer models including the Macintosh through AppleTalk.

Centram Systems West has TOPS, an AppleTalk-compatible network that does not rely on any dedicated disk or file servers; instead it can treat any disk drive or printer attached to any micro in the network as a shared device. Any micro can view the disk directory of any other micro, subject to privacy restrictions.

OTHER NETWORKS FOR MACINTOSHES

Corvus Systems offers its OMNINET for Macintoshes, using its hard disk drives. OMNINET can connect up to 63 Macintoshes on a 4000-foot cable; its raw speed is 0.7 megabit per second.

3Com specializes in Ethernet hardware and software. Its EtherMac uses a 3Server hard disk drive that connects to an Ethernet and to AppleTalk at the same time and acts as a gateway. The Ethernet network functions as the backbone network; IBM PCs are attached to Ethernet directly, Macs are attached to AppleTalk.



Mail Center, from Videx, Inc.

Moving Information Between Computers Chapter 27

For this discussion I assume that you have two computers talking to each other in some way. What kind of information can you move and what can you do with it?

PROGRAMS

The actual program in machine-language form will only be understood by another computer of the same type, so there is no point in moving a program except to another Macintosh. To move programs, use a transport-layer protocol, such as Macintosh Xmodem. If you move the files via a third computer, use MacBinary where possible or a hexadecimal (base-16) format.

Source code for programs—the original instructions in BASIC, Pascal, Logo, or other languages—can be transferred as an ASCII file (with Microsoft BASIC, you should save the program with SAVE "FILENAME",A). Depending on the program's origin, some adjustments may be necessary to run the program on Mac. And you will need the appropriate language interpreter or compiler to turn the source code into instructions for the Macintosh.

TEXT FILES

Most programs that create a text document store two kinds of information: the text itself and the text formatting—the left and right margins, page breaks, tabs, headers and so on. The formatting information is generally interleaved with the text and there is no standard formatting system, not even within the Macintosh world. What you can transfer depends on the kind of text information you have:

- *Raw document.* This is the native file created by a word processor, complete with formatting information. Some raw documents can be made intelligible with only light changes; others are so much trouble that you will have to find another way to transfer the text, or may even have to retype it.
- *Text-only document.* A raw document with all the formatting information stripped out leaves a text-only document. This form is generally easier to work with than the raw document. Many microcomputer word-processing programs can create text-only files, but there are many dedicated word-processing systems that cannot.
- *Final form or page-image document.* The files for some word-processing programs are so hard to work with that you should transfer a page image rather than the word processor's normal data file. A page image is the disk equivalent of the printed page, complete with headers, footers, and page numbering. Most powerful word processors let you make a page image (with the command "print to disk" or something similar). Page images are not particularly easy to work with, but at least they are free of embedded formatting commands. They do, however, have a carriage return at the end of every line. If you use Mac to do any work on the file, you will need to remove those carriage returns to restore word-wrapping.

If your word-processing program permits, here is the quickest way to get rid of unwanted carriage returns. You want to preserve the carriage return at the end of each paragraph, so search for all returns followed by three spaces (or a tab or whatever you use for paragraph indents) and replace them with a # or some other rarely used character or sequence of characters. Then replace every remaining carriage return with a space or with nothing, depending on whether the word at the line end runs together with the word at the beginning of the next line. Finally, replace the # symbol with a carriage return and the paragraph indent. Although this procedure may mangle a few lines, it's a lot faster than deleting carriage returns one by one.

A few word-processing programs—WordStar, for example—store their text using eight bits for each character instead of the more common seven bits; the extra bit must be stripped off before transferring the file. Some communications and utility programs can do this, and many computer hobby magazines have published programs for stripping the eighth bit.

Revisable Document

In a revisable document the formatting information of the original file has been converted into a form that you can edit immediately with the receiving computer and software. The formatting information may not be complete but preserves at least such essential parameters as the left and right margins, tabs, columns, and line spacing. The format coding on the converted document should be in the form of the receiving software so, for example, the left margin is set by the ruler in MacWrite, not by a group of spaces.

IBM's DISOSS

IBM specifies a standard interchange format, DISOSS, or Distributed Office Support System, for its office computer products. The overall strategy is two key format designs, or architectures:

- DCA (Document Content Architecture) specifies the format for transmitting text documents from one system to another. DCA defines such attributes as page width, tabs, headings, and so on. DCA itself comes in two forms: FFTDCA, or Final-Form-Text DCA, and RFTDCA, or Revisable-Form-Text DCA:

Final-Form-Text-DCA:	*Revisable-Form-Text DCA:*
Top margin location	Declare top and bottom margins
Left margin location	Number pages and lines
Line spacing	Specify space occupied by body text
Font definition	Specify page width and height
Justify text	Insert fields from external data records
Begin and end underscore	Include text from other documents
Begin and end overstrike	Keep specified text together on the same page
	Spelling verification control

- DIA (Document Interchange Architecture) specifies the communications protocols for sending DCA files. DIA covers not only the specific transmission structure but how requests are made for a file. It can create a document library for multiple users. DIA specifications describe text files only, although other file types can be specified by their extensions.

DISOSS, used as an the intermediate form between two otherwise incompatible systems, is becoming the *lingua franca* in office word processing. DCA and DIA are complex standards, and compatibility can be defined in many ways. Many companies have announced support for at least part of DISOSS, usually DCA. Some companies provide support only for the FFTDCA form because they fear that using the RFTDCA form may make it easy for their customers to change over to IBM or other RFTDCA-compatible products. These companies are more concerned with defending their own market share than with helping their customers. In some cases independent software developers will offer more complete DISOSS support than the original hardware vendors.

As this is written, no products for DISOSS support have been announced for the Macintosh, but several groups are developing such products. Many Macintosh features, such as font-size changes and graphics, are not supported within DISOSS; most other office computers either do not understand such information or use it in incompatible ways. You need only be concerned about DISOSS support if you must communicate with a DISOSS-compatible computer; the Macintosh with Microsoft Word is actually much better for editing words than any of the traditional office word processors.

Other file-conversion methods

Microcomputers have spawned a deluge of incompatible word-processor formats. Very few format-conversion programs have appeared, aside from a handful of simple file converters for the veteran WordStar word-processing program.

Nevertheless, you should use a program that can perform format conversion whenever possible. This can be a separate program or it can be part of a communications program. MacLink (DataViz, Norwalk, CT) can perform two-way, or bidirectional, transfers with some format conversion between MacWrite and WordStar or MultiMate (in MS-DOS). The format conversion cannot deal directly with font and type sizes, since WordStar and MultiMate do not have such features. Instead, the program changes the point sizes to different character densities, from 5 characters per inch to 13.2 characters per inch in MultiMate, or to normal and alternate pitch in WordStar.

Microsoft supplies a utility program (with Macintosh Word versions 1.05 and above) for bidirectional conversion of Word/Macintosh and Word/MS-DOS or XENIX files.

Among dedicated word processors, Wang systems remain the most widely installed. To move text files between a Mac and a Wang system, you can use a telecommunications option on the Wang. On an asynchronous line, the Wang will only send out text-only files. For a fully formatted file you will need a bisynchronous protocol converter such as the one made by the M/H group. The M/H group also supplies Wang-to-DCA conversion software.

Many service bureaus offer both media conversion (transferring files between physically incompatible formats) and text-file format conversion; costs are fairly high and the turnaround slow. Some companies offer format conversions over the telephone line using a remote computer to do the work.

A dozen companies make disk converters—usually a freestanding computer with a highly flexible disk controller and multiple disk drives in several sizes. The converter can read files in many different physical and logical formats and can write them as well. Software within the converter performs format conversion. A few companies have announced support for 3.5-inch disk formats, but you must check specifically for the Macintosh format. At the high end, the Shaffstall converter with a 3.5-inch floppy option costs over $30,000; low-end units such as the unit from Flagstaff Engineering (Flagstaff, AZ) can be as low as $3000 plus a host IBM PC. The Keyword 7000 system (about $11,000) offers particularly complete text-file format conversion and plans Macintosh support.

Optical Character Readers

If transferring a text file from another computer proves impossible, you can use an optical character reader (OCR) to scan a printed page. Get a clean printout—with a daisy-wheel printer and carbon ribbon—from the original computer. The printed page is the only way to automate input from electric and most electronic typewriters.

Check on any restrictions that may apply to the typeface and page layout. Some OCR systems can only read the special OCR A and OCR B fonts; others read Courier and similarly common fonts. Although some service bureaus offer a document-conversion service with OCRs, reliability and cost have traditionally restricted OCR use to planned, high-volume jobs.

OCRs have been finicky, unreliable machines; they usually can only read originals, not photocopies, and they can't read dot-matrix printer output. The best OCRs to date have used specialized computing hardware for operation; simply adding software to process a scanner's output has not worked well. The Dest 211 OCR (Milpitas, CA) has better-than-average performance; it costs $5000 and up, depending on how many fonts it can read.

At about $500, the Omni-Reader from Oberon International (Irving, TX) represents the first of a new class of low-cost OCRs. It is manually operated; you place the document on an easel and run the reading head along a plastic guide across each line of text. It takes an hour or two to get used to adjusting and moving the Omni-Reader's reading head. Firmware for reading four common typewriter/daisywheel fonts is built-in; other fonts could be added via software.

How well the Omni-Reader works depends on the document and on your manual dexterity. You probably will not be able to read documents consistently from a wide variety of sources. Also, all OCR devices make errors; how many errors you can tolerate depends on the material you are reading and your typing ability. A spelling program can help out for some documents.

OTHER NON-GRAPHICS FILES

Generally each spreadsheet, data base, or other program stores files in a unique way that other programs cannot read. The special symbols on the Macintosh (the ones you get when holding down the Option key) occasionally cause trouble. Some communications programs can strip out the special characters during file transfer.

File-Interchange Formats

Although several file-interchange formats exist, none is universally supported. The DIF format was created by Software Arts, the company that wrote VisiCalc (and is now absorbed by Lotus). DIF files work at the presentation-layer protocol level and contain printable data only; they do not store the formulas and relationships of VisiCalc or other programs. DIF has found its widest use in moving spreadsheet data to a graphing program.

The SYmbolic LinK (SYLK) format devised by Microsoft can store all formulas and relationships; several companies including Lotus (for Jazz) can now use this format. SYLK is the only common microcomputer data-exchange format that runs functionally at the application-layer protocol level. Because the format was designed for universal application, it is somewhat large and inefficient; it is best used only for transferring information, not for normal storage.

Few programs use SYLK as their native file format; you must ask explicitly for a SYLK file to be generated. Once generated, data files in SYLK can be moved to any other computer that has a program capable of reading SYLK files; all the functions remain intact—to the extent that the second program can understand them. 1st Port (version 1.0, from Desktop Software) is a communications and utility program that includes bidirectional conversion to and from DIF, SYLK, and ordinary ASCII, as well as the Macintosh Clipboard, and Desktop's 1st Base database program. 1st Port includes some formatting options. MacLink can also do bidirectional DIF to SYLK transfers and adds conversion of native Lotus 1-2-3 files (files with the extension WKS in MS-DOS) as well.

Both DIF and SYLK can be transferred as ordinary ASCII files; they do not require special handling, although transport- and session-layer protocols will make transfers much easier.

ASCII FILES

Many programs can produce an ASCII file on disk that is equivalent to printed output so their information can be added to a word-processing document. These ASCII files usually lack formulas and relationships.

For programs that contain information in ordered, discrete blocks, such as cells in spreadsheets or records and files in data bases, the ASCII output can often be delimited with a comma or tab placed between each block. Without delimiting, the information is only useful for transferring to a word processor. With delimiting, the information can be placed, block by block, into another program.

Tab-Delimited Files

One way to create delimiters with a spreadsheet program is to add many spaces to each column—more than necessary for any cell. Then save an ASCII representation that includes the spaces; search and replace a block of spaces with a tab. This may be a multistep process, as you must then remove excess spaces and take care that empty cells are not inadvertently deleted.

You can write a program in BASIC or another programming language that reads the original file and converts it to another form or creates a tab-delimited file.

Multiplan and some other spreadsheets on the Macintosh can read tab-delimited data; each block is placed in a cell and each tab moves the following block of data into the following cell. A carriage return moves down to the first column of the next line.

Comma-Delimited Files

Comma-delimited data pose several problems. Commas are often used in large numbers (as in 1,000,000) and Europeans often use commas as decimal points. Because of this confusion, Multiplan on the Mac does not read comma-delimited data. If you need to use such a data file, you must convert the commas to tabs before moving the file.

If no commas are used in numbers or as decimals, then the switch is easy with a simple BASIC program, or it can be done on many word processors. Use the word processor's global search-and-replace function, but first check to make sure it can replace with a tab, as some programs can only search and replace ordinary characters. If your particular program does not accept a tab as a valid character for replacement, try Control-i, the ASCII code for tab. You may need to change valid commas within fields to another character temporarily and restore them afterwards.

Unless you are moving a SYLK-format file, all this effort still leaves you with incomplete information, for the formulas and relationships have been lost in the transfer. Nevertheless, for a spreadsheet model, the headings and number entries are correct and reconstructing the formulas is simpler than starting from scratch.

SOME SPECIFIC CASES

This section gives procedures for moving information from some specific programs running on the IBM PC to the Macintosh. The examples are mostly for Multiplan on the Macintosh; you can use the basic concepts to move information to other programs. The principles described apply to many different types of programs running on other computers.

You must be reasonably fluent with these programs to make these steps work; if necessary, find someone who can help you the first time through. The notation and instruction formats used here follow the style of each program's manual. In the following steps, all characters, including quotation marks, are literal. (Some of the specific programs mentioned here will appear in Macintosh versions that can read files generated by their counterparts on other computers. If so, simple file transfers will be possible.)

Multiplan

Moving Multiplan files between the Macintosh and another type of computer is easy once you have a communications link. Run Multiplan on the other computer, and then follow these steps:

First use

Transfer Load filename: then use the

Transfer Options command, and select the

Symbolic (SYLK) rather than normal mode. Finally,

Transfer Save the spreadsheet.

Use a communications program to move the new SYLK file. On the destination Macintosh, simply read it as a SYLK file (see Using a SYLK File later in this chapter). No changes will be necessary and all functions and formulas will work.

Since Multiplan can consolidate data from several spreadsheets, you can thus prepare part of the data on one computer and move the file to another one.

VisiCalc

Save the VisiCalc file on disk, using the normal /SS command (not the /S#S command for DIF). Use the MS-DOS or PC-DOS version of Multiplan and then follow these steps.

First use

Transfer Options Other to set Multiplan to read a VisiCalc file, then

Transfer Load filename: to load the VisiCalc file into Multiplan. You need to save it in SYLK format with

Transfer Options Symbolic and save the file with

Transfer Save filename:.

Now move the new file in SYLK to Mac; you can read it directly with Multiplan on the Mac, and all the formulas and relationships will be intact.

dBase II and dBase III

Since dBase II and the more recent dBase III create data bases and Multiplan is a spreadsheet, you won't be able to perform all the same functions, but Multiplan can use information from dBase II and dBase III files as well as pass it on to Microsoft's Chart program.

On the originating computer, load dBase II or III, and type

.USE <filename>

.COPY TO <filename>.TXT DELIMITED WITH "

Instead of double quotes you can use any other unambiguous character. Empty logic fields will appear as a space, empty numeric fields as 0.0.

Modify <filename>.TXT with a word processor. Each field in the file will be separated by "," (quotes-comma-quotes). Search for and replace these characters with a tab. Records are separated with a carriage return and therefore do not need modification. Then move <filename>.TXT to Mac and use it as a tab-delimited file.

dBase II and dBase III applications put any formulas and relationships among the contents of a data base into a program that cannot be transferred. If you need the formulas on Mac, you must enter each one manually.

SuperCalc

The SuperCalc file format is shared by all three versions (SuperCalc, SuperCalc2, SuperCalc3) as well as by PeachCalc.

Save the SuperCalc spreadsheet in the normal way, producing a file with the extension .CAL. Load the Sorcim/IUS Super Data Interchange program and select from the menu:

SuperCalc files to Comma Separated Value file.

The new file will have the extension .CSV. Using a word processor, convert it to a tab-delimited file by searching for the commas and replacing them with tabs. Since SuperCalc does not accept European-style decimals, there should be no ambiguous commas.

The formulas and relations in the SuperCalc model are lost during the transfer.

Lotus 1-2-3

Use the MacLink communications/file-conversion program or the utilities included in Jazz and Excel.

Using a SYLK File

Once you have the SYLK file on a Macintosh disk, you can read it easily. Start an application program, such as Multiplan, that can read a SYLK file. Once the program is loaded, choose Open from the File menu. The SYLK file will appear in the list of available files; select it and click the Open button.

Using a Tab-Delimited File

When you have the tab-delimited file on a Macintosh disk, start a word-processing program and choose Open from the File menu; select the file. When the information is on-screen, select it and choose Copy from the Edit menu, putting everything into the Clipboard. Then choose Quit from the File menu. Start Multiplan, and after it is loaded, choose Paste from the Edit menu. The Clipboard contents then go into the spreadsheet.

MOVING GRAPHICS FILES

If moving text and numbers is a problem, moving graphics is even more difficult. Graphics come in two main forms. For structured graphics of the kind MacDraw creates, there are no standardized formats in the microcomputer industry. For bit-mapped graphics of the kind MacPaint makes, there are no standards either, but many bit-mapped files use a relatively simple format, so you could write a program to convert them. For example, converting files from the Wang Professional Image Computer into a MacPaint form is fairly easy.

If you want to move screen graphics and the originating computer uses a standard RS-170 video-signal format, you can capture the screen image with a video digitizer (see Chapter 17) and get a MacPaint file. Or you can use a scanner on the printed output.

COMMUNICATIONS/FILE-CONVERSION PROGRAMS

Two products are designed specifically for communications between an IBM PC (and compatibles) and a Macintosh. They include both MS-DOS and Macintosh software and a connecting cable, although they will also work via modem.

MacLink (version 1.14, from DataViz, Norwalk, CT) is exceptionally easy to use. Once the IBM PC side of the program is started, you do everything from the Macintosh including changing and searching subdirectories. Besides the format conversion described earlier in this chapter, MacLink also has a security password provision.

PC to Mac and Back (version 1.0, from dilithium Press) does not do format conversion except for a simple eighth-bit stripping process for WordStar files, and it requires both machines to be attended during operation. Seaquest Software plans to release a revised version of this software as Message Mover.

The less ambitious MacTransfer (Southeastern Software, New Orleans) comes in two versions—IBM PC and Apple II—and consists of simple programs that are able to transfer text files only.

1st Base, whose file-conversion features were described earlier, supplies communications support on the Macintosh side only; you need a separate program on the other computer.

GETTING INFORMATION FROM A LARGE COMPUTER

All the problems of moving information from one microcomputer to another also apply to large computers. Mainframe computers are at their best handling large data bases and crunching huge arrays of numbers. Since large computers are always managed by data-processing specialists, the software they use appeals to the specialist rather than to the ordinary professional. Or, to put it another way, mainframe software is much harder to use than microcomputer software, especially Macintosh software.

Getting information out of a mainframe and into a micro isn't easy. In most cases, you must go through a normal session with the micro acting as a terminal to the mainframe data base and store all the results on disk in the micro. Then you must go through the information, and spend considerable time manipulating the data into a form suitable for a microcomputer program. The whole process is so tedious that most people don't bother to do it electronically; instead they print out the results and then type those results into the microcomputer, one by one. Finally, they can look at the results with a spreadsheet or graphing program.

A few microcomputer programs can now perform this transfer with less pain. These programs act as a buffer between you and the mainframe. You ask for information through a microcomputer interface; the program converts your requests into commands understood by the mainframe database manager. Information from the data base then passes through the microcomputer and appears in a spreadsheet, ready for your next step. In time, all important mainframe programs will have such microcomputer support; until then, there will be much wasted and repetitive work.

A similar problem applies to many data bases designed for access by microcomputers. All of the popular dial-up information services currently use awkward, obsolescent interfaces; they appeal only to determined computer hobbyists and specialists. Dow Jones, however, does have a special software product that hides some of the problems: Spreadsheet Link is specifically designed to extract financial information from the Dow Jones data base and put it into the SYLK format that can be read by many spreadsheet programs. Spreadsheet Link requires the Dow Jones Straight Talk communications program.

For specific information on the most popular dial-up electronic information services, see *MacTalk, Telecomputing on the Macintosh* by Sheldon Leemon and Arlan Levitan (Compute Publications, 1985), which deals specifically with the Macintosh and emphasizes services for computer hobbyists. *Online* by Steve Lambert (Microsoft Press, 1985) gives more information about business services from a general microcomputer perspective.

Reproducing the Macintosh Screen

Chapter 28

To share a Macintosh screen image with a small group of people, copies of a simple printed output will usually do. For presentations to large groups and for publication, advertising, or training, the choices are more complex. This chapter discusses the main methods of reproducing the screen.

Much of this chapter is highly technical; for each section, I assume the reader has the technical background appropriate to the specific topic at hand. Essential terms are defined in the glossary, but this chapter cannot be a substitute for a technical handbook on photography or video.

Many readers will want to review just the sections on still photography. For the most part, you will get satisfactory results if you just follow as much of the advice as your equipment can handle. If you are a cinema buff, you might read the section on motion-picture photography to learn how the professionals do it. Similarly, video enthusiasts might be interested to see where the Macintosh fits into the flood of present and forthcoming video equipment.

AN OVERVIEW

The alternatives for reproducing the screen image are:
Print a screen image on paper (low to medium quality).
Photograph the screen directly:

- Still photography (medium quality; low cost [$250]; quick).
- Motion-picture photography (very low quality without special equipment; medium quality with expensive [over $10,000] special equipment).

Move direct video signal electronically; requires a modified Mac.

- Photograph the image with a CRT/camera combination (medium or better quality; moderate-cost [$2000 to $3000] equipment).
- Put image on photographic film with electron beam or laser imaging device (high quality; very rare, extremely expensive [over $80,000] equipment).
- Distribute video signal to special monitors and projection televisions (medium to high quality; moderate-cost [monitors $1500 to $2000, projection television about $6000] equipment).

Convert Mac's video signal to a conventional video signal; distribute via closed-circuit television, broadcast television, videotape, or videodisc:

- Point a TV camera at the Mac screen (very low quality; simple, low-cost [$2000] TV camera).
- Electronic standards conversion (low to medium quality; complex, extremely expensive [over $40,000] equipment, not yet built).

Print screen image to disk; create reproduction-grade image with typesetter (high quality; expensive [$30,000 and up] equipment).

Although the choices are complex, you probably have only a few of the techniques readily at hand, effectively limiting your choices. If you need commercial photographs or a motion picture of the screen, a small group of professionals specialize in CRT images. Other options, such as typesetting, are provided by service bureaus.

Many of the video techniques described here require that the video signal or some part of it be brought out of the Mac. Several companies perform the conversion; check the warranty situation carefully.

STILL PHOTOGRAPHY

You can either photograph the screen directly or photograph a printed screen image (a screen dump). Photographing the printed image is much easier, especially if you have a copy stand and macro lens. Imagewriter output, however, lacks the visual impact of the screen itself, and the tones are always uneven because printers are never perfectly aligned and ribbons are never free of irregularities or wear. A LaserWriter screen dump is much better, a typeset copy even better.

Photographing the Screen

The comments here apply to a 35-mm single-lens reflex camera, the most common high-quality camera type available.

Equipment

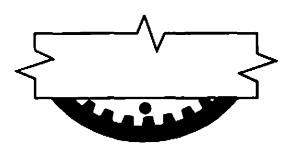
Mac's image size is 4¾ by 7 inches (12 by 18 cm), a 5:1 reproduction ratio for 35-mm film. Most normal lenses will not focus close enough for a full-screen shot; in most cases, extension tubes bring the camera in too close. Supplementary close-up lenses won't provide the sharpest results, but a +1-diopter lens with a 50-mm main lens will work.

A macro lens gives the best results. The Mac screen has a slight curvature, so straight lines may come out slightly curved with shorter-focal-length lenses. A 50- or 55-mm macro lens works satisfactorily; a 100-mm macro works even better. Macro zoom lens pictures won't be as sharp in the corners because of inherent design problems aggravated by the screen curvature.

Use a darkened room. If there is any stray light in the room, put a hood between the screen and camera; any closely woven, dark cloth serves well.

Photographic novices should note that the Mac screen creates its own light; if your camera reads inadequate light, a flash will not help, and you cannot take a usable picture.

Settings



To mark the screen brightness for photography, put a dab of nail polish on the brightness control.

Adjust the screen to a high, yet comfortable brightness level. In order to duplicate the screen brightness, you can put a tiny dab of nail polish on the brightness control.

Mac refreshes the screen 60.15 times a second. A shutter speed faster than 1/60 second will show only part of the image. A 1/60-second shutter speed won't show the entire image either, because of the time required to open and close the shutter curtains. Slightly longer exposures will be uneven, as part of the image may be refreshed one more time than the rest of the image; the unevenness will depend partly on which direction your shutter travels.

An uneven refresh will be less of a problem if the overall image is overexposed a little. You should achieve passable results at a ⅛-second shutter speed, and better results with longer exposure. With a shutter speed of one second, the density variation is at most 1/60 of the overall exposure and will be invisible.

Note that different CRTs vary in their light-decay characteristics, so experience gained with other computers or television sets may not apply to Macintosh.

With such slow shutter speeds, the camera must have a rigid support or tripod.

For exposure, take a through-the-lens reading on any mostly white image, then set two stops more exposure as a first trial; this will render the whites as white instead of a medium gray tone. Standard light meters may not be accurate because of a spectral sensitivity that is not matched to the screen's spectral balance. Light-meter readings may be off by as much as two f/stops.

For most macro lenses, the best f/stops will be f/8 or f/11. Smaller f/stops will be less sharp, while larger f/stops make focusing and depth of field more critical. Bracket exposures by half stops the first time, three stops total in each direction, and keep a record.

Film

If you are going to project the image, in most cases you will want a positive image (black letters on a white background as on the Mac screen), especially since some graphics and icons are hard to understand when shown in negative form. The exceptions may be cases where you must project in a partially lit room, where negative images can be easier to read.

Black-and-white positive films are rare; the only common high-contrast positive film, which is the best choice and also gives quick results, is Polagraph HC instant slide film for Polaroid's instant 35-mm Autoprocess system. The other two Polaroid instant slide films (the continuous-tone, black-and-white Polapan CT and the color Polachrome CS) don't work as well.

If these films are unavailable or not suitable for some reason, you can check with graphic-arts facilities to see which processes they can handle. Most graphic-arts labs use a two-step process, printing negative-to-negative to get a positive. Kodak High Contrast copy film works well as the negative material; use it when you need reproduction-quality images for printing or reproduction. As with all high-contrast materials, the exposure latitude is small. For black-and-white film, no filters are needed.

If you don't have a convenient graphic-arts facility nearby, you can use color film with commercial processing. The exact color balance of CRTs depends on the phosphor coating that creates light. Macintosh uses a P-4 two-component phosphor with two emission peaks, at 460 nm and 560 nm. You can see the two components with a magnifying glass; some regions are a little more bluish than others. Because of a peaked emission, the color balance depends on exposure and emulsion choice. The concept of color temperature only applies to smooth emission curves, not the P-4 spectrum profile, but the rough equivalent color temperature is high, about 11,000 degrees Kelvin. Uncorrected color photos will look bluish.

Color correction

Optimum correction of the Macintosh color balance will require some experiments with your exposure and film-emulsion preferences. As a starting point, use daylight color film and a color-balancing filter. An 85C filter should yield neutral white tones; the more common 85 filter also works, yielding a warmer tone. You may need a CC (Color Correction) filter as well. Kenko's TV-CC filter may work effectively, depending on the film emulsion. If you must use tungsten-light color film, try to find the rare 86 filter or combine an 85B with an 81 EF. Kodachrome emulsions have better contrast and sharpness than other color films, but fewer laboratories can process the film, so you may have to wait longer. If you want to experiment, you can create multiple-exposure images on Mac with color filters.

Some photographers and art directors may question the need for color correction, arguing that if the screen is bluish, then it should look that way in print. Perhaps they think the blue confers a cool, high-tech feel. However, when we look directly at the Macintosh screen, it appears white, not blue. Our visual systems adjusts the predominant illumination to white in the same way that we adjust to daylight or tungsten light.

Photographing the Screen as Part of a Larger Picture

If Mac's screen is only part of an overall picture, the bluish screen will mar the image. You can correct this by retouching or in the exposure. You have two strategies you can follow for the exposure:

- Single long exposure, if all elements are static and the overall light level can be balanced against the screen.
- Double exposure, once for the scene and once with a long exposure for just the screen. For the first exposure, black out the screen with photographic velvet; for the second, extinguish all other light sources and use a matte box if necessary to reduce spilled light from the screen.

To achieve color balance, you can:

- Filter the overall light to match the color balance of the Mac screen. If you have people in the scene, try an 80C or 80B gel over electronic flash units and a warming filter over the camera lens. Run tests if the results are critical; small errors can be corrected in the lab.
- Filter a double-exposed image separately, using the filters appropriate to the light source.
- Place a color gel over the screen. An 85 gel, available from motion-picture supply houses, may be roughly correct.
- The screen photograph can be taken at a different time and inserted as a photo composite. Achieving high-quality results with this method is difficult and expensive, except with the newest digital-image processors, such as the Scitex, which are merely expensive.

MOTION-PICTURE PHOTOGRAPHY

With ordinary cameras, motion-picture photography of CRTs will always show bars moving rapidly over the screen because the video scan rate is not synchronized with the motion-picture framing rate. If you have the resources, here's how to get the best possible results.

Get a motion-picture camera with mirror reflex viewing, 180-degree shutter, and an input for external speed control. For best results, the shutter should move vertically, from bottom to top (to match the direction the Macintosh writes the image—top to bottom, inverted for the camera lens).

To control the camera's speed, bring the 60.15-Hz vertical-sync signal out of the Macintosh (the signal is on a wire connecting the CPU board and the video/power-supply board) and divide it by two to 30.08 Hz; use the signal to control the camera's speed. The division won't be necessary with some units. With a good phased-locked system, you can set the controller so the camera shutter is open as Mac writes the image on the screen.

Alternatively, you can use a separate adjustable precision oscillator for the camera control signal. The oscillator must be accurate to 1/1000 of a frame to maintain sync; the longer the take, the higher the precision necessary. Make sure the shutter is set to 180 degrees, and run the camera without film to fine-tune the speed. For each take you must adjust the shutter phase on the speed control: Hold the phase button down until the thin sync bar rolls out of the CRT image and release it before the line returns. With a 180-degree shutter, if the bar is out of the eyepiece, it is out of the film; the film will record every other frame of the Macintosh screen.

For sync sound, simply operate normally; the camera control signal will drive the Pilottone track on a Nagra or other recorder at a frequency 25 percent higher than normal, a frequency that poses no problems for most resolving equipment. (A Nagra IV will accept a 5-volt square wave with a 50 percent duty cycle for the pilot.)

After you have the film and sound track, you can transfer them to videotape: one frame of film to one frame (two fields) of video instead of the ordinary 24-frames-per-second (fps) film transfer. The audio should be resolved to a magnetic film recorder running at 30 frames per second. The slight speed shift downward to the 59.94-Hz NTSC broadcast-standard field rate isn't important.

If the result will be used in a 24-fps motion picture, then you will have to convert speeds with a step printer, essentially dropping every fifth frame; transfer the sound at 24 fps. Step printing is expensive and may result in minor motion discontinuities.

The method just described generally works better than the classic technique for filming a television screen, which does have the virtue of producing a 24-fps film. This technique requires that you use a movie camera with a 144-degree shutter and external sync; slave the camera to Mac's vertical-sync signal.

The common practice of filming European television systems (50-Hz field rate) at 25 fps with a 180-degree shutter will not work with a Macintosh.

In some situations, you may get better results by replacing the CRT with another design that has a longer-persistence phosphor. If you have the technical expertise, you can also change the clock speed and, thus, the framing rate.

Step-Frame Motion-Picture Photography

For the best results, script and budget permitting, shoot the motion picture with double exposure. On the first pass, black out Mac's screen and shoot the live action or other surrounding material. Then shoot the screen on the second pass with a step-frame motor control, exposing each frame as a still image with an exposure of ⅛ second or longer and filtration. Use computerized stepping motors for camera movements if necessary. Matching the action may require a traveling matte shot. You could create the matte with a program that generates a blank, white screen. For a clean matte with adjustable brightness, take an empty Macintosh case and replace the screen with a translucent white screen. Cut a mask that matches Mac's screen area and illuminate the screen with a floodlight; use a filter for a blue screen matte.

ELECTRONIC USE OF THE MAC VIDEO SIGNAL

If you have a video-output adapter on your Mac, you can use its video signal in many different ways. In all cases, the equipment receiving the signal must be capable of handling a 22.255-kHz horizontal sweep frequency, much higher than the standard 15.738 kHz used in North America and Japan or the 15.625 kHz used in Europe. The equipment should also have convenient controls to adjust vertical and horizontal size as well as linearity. Some of the ways you can use the signal electronically include:

Using a camera/CRT combination system to photograph screen images in a specially designed, light-tight box. Most current units cannot handle Mac's sweep frequency, but the Lang model 1014 ($2800) and the MacSlideMaker from MicroGraphic Images can cope.

Driving an electron beam recorder (EBR) with the video signal for the ultimate in real-time photographic imaging. An EBR essentially replaces the CRT phosphor with photographic film; the scanning electron beam strikes the film directly without any optics. The entire mechanism works in a vacuum chamber. EBRs are extremely expensive and used mainly for high-quality video-to-film transfers. Laser-beam recorders work similarly but their color capability isn't needed for ordinary Mac images.

These elaborate techniques don't really make much sense for a still image; typesetters work just as well or better and are much cheaper unless you need a very large number of images.

Driving an external monitor or series of monitors with Mac's video signal. The monitor quality must be high; the bandwidth of ordinary televisions is limited to 4.5 MHz (more typically less than 3 MHz) and good closed-circuit monitors have 5- to 10-MHz bandwidth. The Mac video signal needs a 20-MHz bandwidth monitor for best results.

Several companies offer 25-inch monitors that work with Mac's sweep frequency at a cost of about $1500. The Electrohome model EDP 57 and other video projectors can show Mac images.

A distribution system using Mac's video signal will work with ordinary video signals as well, but the monitor settings for sync and image sizing will usually be different.

CONVERSION TO AN ORDINARY VIDEO SIGNAL

The simplest way to get an ordinary video signal out of Mac is to point a television camera at it. The results are poor, with a major loss of sharpness. Nevertheless, with a good camera, the results can be good enough for training and other less critical applications. For most people, this is the only way to get a videotape of the Mac screen.

Aside from the loss of sharpness, the difference in framing rates between Mac and a 60- or 59.94-Hz television produces a line or sync bar that runs slowly down the screen every few seconds. The line flutters more disturbingly on European 50-Hz television.

The only way to get rid of the bar is to slave the television system to Mac's vertical sync. In a studio, slave the studio sync line to the Macintosh and operate all equipment, including the videotape recorder, at 60.15 Hz.

As with other methods using two scanning processes, moire problems may emerge at some magnifications. (Since the Mac image is black and white, you can kill the color-burst signal to reduce the quite separate NTSC moire problem.)

The best way to convert the video signal is with a standards converter, similar to those used for converting European television to American and vice versa. The original image is stored in a large bank of RAM called a frame store or buffer. Unlike a CRT where the image begins to decay when the electron beam passes on, a frame store holds each part of the image until it is replaced by a new frame. Meanwhile the image is read out by a separate circuit operating at the rate of the converted signal, so the frame store acts as a buffer between the two video systems. No sync bars or other artifacts are visible except for minor discontinuities in movement.

Beck-Tech (Berkeley) has introduced the first relatively low-cost video-standards converter designed to handle Macintosh images. The company's $3000 Chromatron scan converter incorporates a one-megabyte frame buffer and produces a video signal that meets broadcast standards. The converted Mac screen is centered within an interlaced NTSC video frame. Repetitive black-and-white patterns in the original Mac image generate stable NTSC colors by an artifact; the spatial frequency components happen to produce the colors when accompanied by a color-burst signal. However, the color images display considerable dot crawl.

In the long run, higher-quality television distribution systems could preserve all of Mac's image quality. A high-definition television system can now, in some cases, meet or even exceed Mac's requirements. Japan Broadcasting Corporation's experimental HDTV (High Definition Television) uses 1125 interlaced lines (1041 lines visible) with a 20-MHz video bandwidth. To achieve such quality, HDTV has to replace every component of our present television systems, from cameras to recorders, transmitters to television sets.

If you need the best possible image, either photographic or video, contact consultant John Monsour, 2062 Stanley Hills Drive, Hollywood, CA 90046.

Technical Topics

Chapter 29

This chapter offers background information for people with some knowledge of computers; this information isn't essential to the average Macintosh user.

VIDEO SCREEN

Video screen: 9-inch diagonal, black-and-white

Resolution: 512 by 342 dots, bit-mapped, non-interlaced

Image area: 4¾ by 7 inches (12 by 17 cm)

Phosphor: P4, medium-fast, dual component

Spectrum peaks at 460 nanometers, 560 nanometers

1931 CIE color coordinates: $x = 0.27$ $y = 0.30$
1964 CIE color coordinates: $u = 0.18$ $v = 0.29$
Equivalent color temperature: 11,000 degrees Kelvin

Horizontal scanning frequency: 22.254545 kHz

Horizontal flyback time: 13 microseconds

Frame rate: 60.1474 Hz

Vertical blanking interval: 1.24 milliseconds, 7.5 percent of duty cycle

Dot rate (frequency of generating dots on screen): 15.6672 MHz

Scanning time per pixel: 65 nanoseconds

If connected externally, required video amplifier bandwidth: 20 MHz. (There is no internal video amplifier.)

Screen operating speed: 60.15 Hz instead of 60 Hz. (60.15 Hz is the speed that falls out from dividing the master clock frequency. The Macintosh uses only one clock, aside from the clock/calendar.)

MICROFLOPPY DRIVES AND DISKS

There are three different Sony-type microfloppy drives: the original Sony interface version, a 5¼-inch emulator version, and the Apple unit. All come in single- and double-sided models. The Apple version has a variable-speed disk controller and an auto-eject mechanism; the other two types are not compatible with the Macintosh.

On double-sided disk drives, the heads are offset; the top head is slightly closer to the center (59 mils or 1.5 mm, the equivalent of eight tracks). This follows the practice of earlier, larger floppy disk drives; originally this was supposed to improve head contact with the disk although new head geometries probably render the offset unnecessary.

The red LED lights visible through the disk slot are leftovers from the original non-Macintosh microfloppy drives, where they indicated disk activity. Sony found that leaving the lights inside was simpler than removing them with a circuit redesign.

Microfloppy disk specifications and performance (materials for Sony brand disks) are as follows:

Disk jacket size: 3.54 by 3.7 by 0.13 inches (90 by 94 by 3.3 mm)

Disk jacket construction: ABS plastic

Disk inner diameter: 25 mm

Disk outer diameter: 86 mm

Recorded area (bottom side for single-sided disks):
- bottom side: inner radius, 22.428 mm; outer radius, 41.961 mm
- top side: inner radius, 21.666 mm; outer radius, 40.970 mm
- Tracks: 80 tracks per side; track density, 135 tracks per inch
- Track width: 0.12 mm
- Substrate: polyethylene terephthalate
- Coating: cobalt adsorbed iron oxide, 0.0019 mm thick

Error rate:
- soft error rate: one in 10^9 bits (correctable on retry)
- hard error rate: one in 10^{12} bits with ten retries (not correctable)
- seek error rate: one in 10^6 seeks (finding the right track; correctable with retry)

DISK-DRIVE SPEED

Chapter 14 contains a brief discussion of disk-drive speed; here is a more complete explanation.

Raw transfer or peak rate. On the Macintosh, the floppies transfer data at a peak rate of 62.5 KB (500 kilobits) per second.

Hard disks transfer raw data at 625 KB per second (high-performance units at 1.25 MB per second). This is too fast for most microcomputers, so the disk-controller circuitry slows the data rate down to about 200 KB per second; instead of every sector on a track being read in one revolution, the disk must go around several times (this number is the interleave factor). On the Macintosh, even these slower speeds are only possible when the hard disk is connected to the CPU bus. Port-connected hard disks are slower.

Track-to-track access time. To move between adjacent tracks, typical floppy or hard disk drives take 3 to 5 milliseconds. Since many head movements travel over many tracks, the average access time runs from 30 to 100 milliseconds in hard disk drives. The slower hard disk drives use stepper motors to move the heads. Medium-performance disk drives use linear motors. Expensive, higher performance drives use voice coils, moving the heads like a loudspeaker cone and employing servo signals recorded in the track to help home in quickly on the narrow tracks. Typical access times on a floppy drive are well over 100 milliseconds.

Bits per inch. A floppy-drive head must perform more head moves to read the same amount of data because each track stores less information than a hard disk track.

Track geometry and cylinders. Double-sided floppies require half the head movement of the single-sided variety because the two heads move together and contiguous tracks are recorded alternately on the bottom and top surfaces. In the middle tracks, a single-sided drive can store 5.12 KB before it must step to the next track; a double-sided drive can store 10.24 KB. (It's because of this alternate bottom/top track usage that a single-sided drive cannot read useful information from a double-sided disk; by shutting off the upper read/write head, a double-sided drive can format, read, and write single-sided disks.)

Hard disk drives take the same idea further: They have an array of heads that move together and work on a stack of magnetic platters. All the platters have magnetic surfaces on both sides and share a common spindle. When recording a file, the drive steps the multiple-head assembly to a specific track location and then records the information on each platter surface in turn before the heads move on to the next track. Taken together, all the tracks associated with a particular head position are known as a cylinder. Floppies store 5.12 KB or 10.24 KB per cylinder as described, but a hard disk can typically store from 35 to 100 KB in a cylinder before stepping to the next track.

Block size and fragmentation. A disk operating system allocates the available storage into blocks, and the file allocation table in the disk directory keeps track of which files are in which blocks. A block is a logical entity whereas a sector, track, and cylinder are physical entities; the smallest practical block is usually a sector; the largest blocks can be bigger than a cylinder. A block is the minimum space that a file can occupy on a disk. Small files are padded with null characters to fill out a block; large files take up many blocks.

The boot blocks (blocks 0 and 1) are reserved on each floppy disk. The directory is in block 2. All other blocks are assigned independently. On a fresh disk a large file will take up contiguous blocks, but after many files have been created and deleted, files tend to fragment into non-contiguous blocks, as the operating system stuffs data into any available block. If the block size is small, the fragmentation gets worse, requiring more head moves to reach each block; if the block size is large, the disk cannot store many small files since the blocks are quickly used up and store more padding than data.

Occasionally a user can select the block size, depending on specific needs. More often the block size is fixed or, as in the case of some Mac hard disks, set according to the size of a disk partition or volume.

INPUT/OUTPUT PORTS

The Mac's RS-422 serial ports are specified for operation up to 230,400 bits per second. With external clocking, the ports can run up to 920,000 bits per second. The RS-422 interface uses balanced signal lines in pairs; the related RS-423 interface uses unbalanced lines.

Apple does not follow the RS-449 connector standard in its 9-pin RS-422 plug. (The standards for RS-422 and RS-423 specify the RS-449 connector standard, with a 37-pin D connector for primary circuits and a 9-pin D connector for secondary circuits.)

Wiring for the Macintosh RS-422 Ports

1 Ground

2 +5 volts

3 Ground

4 TXD + (transmitted data)

5 TXD −

6 Filtered +12 volts

7 External clock in synchronous communication; also for handshake for printer or carrier detect (CTS)

8 RXD + (received data)

9 RXD −

Although power is available on lines 2 and 6, connecting devices should not depend on the power. The available current is small—200 milliamps at +5 volts and 100 milliamps at +12 volts from all ports (except the disk-drive port) combined. The +12-volt supply is for logic switching.

The balanced RS-422 interface is strapped to the unbalanced RS-232C form for the Imagewriter or for RS-423 devices. Radio Shack catalog #276-1978 is a low-cost extension cord for the serial ports and the mouse port; it lacks locking screws, sometimes a problem.

Cable Wiring

WIRING FOR THE NULL-MODEM CABLE,
RS-422 TO RS-422

Connector 1		*Connector 2*
1		1
2	no connection	
3*		3*
4		8
5*		9*
6	no connection	
7		7
8		4
9*		5*

For short runs without hardware handshake, you need only connect the three wires marked with an asterisk. Connect pin 7 for hardware handshake.

Other Cables

Mac	*Apple modem*
3	3
5	9
6	6
7	7
9	5

Mac	*Hayes modem*
3	7
5	2
7	5 (optional)
9	3

Mac	*IBM PC (DTE)*	
3	7	On the connector plugged into the IBM PC,
5	3	tie pins 4 and 5 together and
9	2	tie pins 6, 8, and 20 together

Mac	*Imagewriter or Radio Shack Model 100/200*
3 and 8	7
5	3
7	20
9	2

A Smart Cable (from IQ Technologies, Bellevue, WA) or an Easy-Cabler can solve many wiring problems. Once wired, you can tackle the software configuration problems. Good luck.

Wiring for the Mouse Connector

1 Ground

2 +5 volts

3 Ground

4 X-2

5 X-1

6 Not connected

7 Switch bar for mouse button; pressing the button connects this line to ground

8 Y-2

9 Y-1

The X and Y connectors are the quadrature signals for the horizontal and vertical axes. If, for some reason, you want to reverse the mouse action, you can swap X-1 and X-2 or Y-1 and Y-2.

Wiring for the Keyboard Connector

1	Black	Ground
2	Red	Clock
3	Green	Data (device ID and keyboard scan code)
4	Yellow	+5 volts

The keyboard port supports communications among eight daisy-chained devices and the Mac. A token-passing protocol controls which device can send information at a particular time. The information travels in 8-bit codes. The codes sent by the numeric keypad are distinct from those sent by the main keyboard; the 5 key on the numeric keypad sends a different code from 5 on the main keyboard. The Mac can distinguish among eight keyboard types as well as the eight different devices.

The keyboard cable uses RJ-11 plugs, the same as in a modular telephone handset (narrower than a telephone wall jack plug). However, the cables are not interchangeable; the keyboard cable uses heavier gauge wire and is wired straight through (pin 1 to pin 1, etc.) whereas the handset cable wiring is flopped (pin 1 to pin 4, pin 2 to pin 3, etc. to make inline coupling plugs simple).

Disk-Drive Port Connector

1	Ground (multiple grounds for interface signal integrity)
2	Ground
3	Ground
4	Ground
5	−12 volts
6	+5 volts
7	+12 volts
8	+12 volts
9	Not connected
10	Motor speed control
11	CA0 (status control line)
12	CA1 (status control line)
13	CA2 (status control line)
14	LSTRB (status control line)
15	Write request
16	SEL (select disk drive)
17	External drive enable
18	Read data
19	Write data

SOUND GENERATION

The sound generator operates during the horizontal flyback time. The 68000 CPU processes sound-generating information when it is not busy with the screen—while the electron beam is traveling back ("flyback") from the right edge of the screen to the left edge. The flyback occurs at the horizontal sweep rate, or every 44.93 microseconds. All frequencies are built up out of multiples of this time. The highest possible frequency is twice this period, 89.96 microseconds, or 11.116 kHz (Nyquist limit). From the programmer's point of view, there are three different sound synthesizers:

- The four-tone synthesizer for harmonic tones; 8-bit digital-to-analog conversion.
- The square-wave synthesizer for beeps and other effects.
- The free-form synthesizer for complex music and speech.

PARAMETER RAM

The very low-power CMOS clock/calendar operates off a battery (Eveready No. 523 or equivalent). The battery keeps the circuit alive at all times, even when the computer itself is turned off. In addition to the time and date, the custom chip contains the 20-byte parameter RAM, which stores the following information:

Validity status: a check for valid information; a failure restores default values

Modem-port configuration (port A): default is 9600 baud, 8 data bits, 2 stop bits, no parity

Printer-port configuration (port B): same as for modem port

Time setting: default is midnight, January 1, 1904

Default application font: font 2

Keyboard repeat threshold: default is 24 ticks, settings in four-tick steps (one tick is 1/60 second)

Keyboard repeat rate: default is 6 ticks, settings in two-tick steps

Printer port: whether printer information is sent to printer or modem port; default is printer port

Volume control: for speaker

Mouse double-click time: default is 32 ticks, three settings in four-tick steps

Mouse scaling

Pointer blink time: 32 ticks, settings in four-tick steps

Menu blink: how many times a menu selection will blink (0 to 3)

Start-up disk drive: internal or external

Unused: three bytes and a few bits

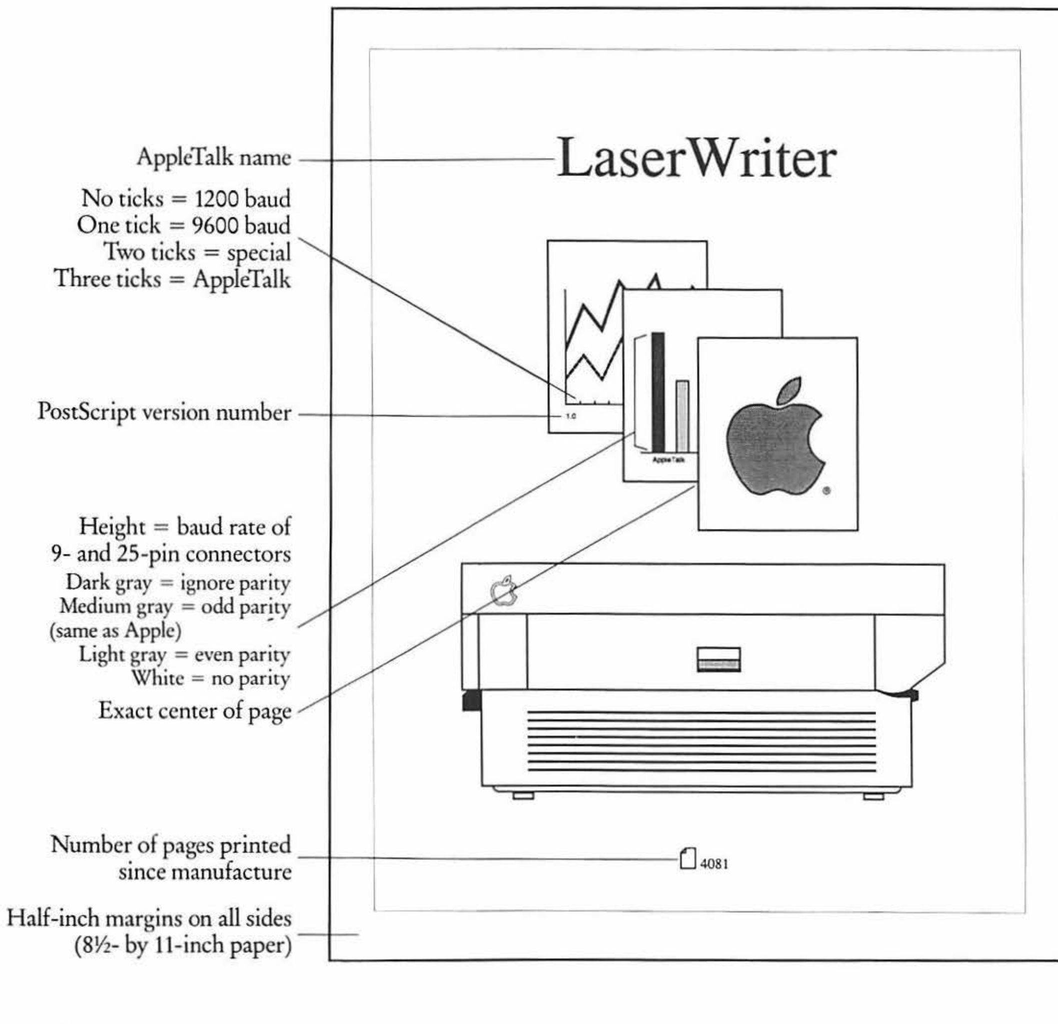
These values are read from the parameter RAM into the main RAM when you turn the power on. The Control Panel desk accessory can change most of these values. A few set-up programs (such as the Microsoft MacEnhancer) can change the port settings, if you use the Control Panel afterward; the Control Panel writes the main RAM values into the parameter RAM. To restore default settings, remove and then replace the battery (reset the clock if necessary).

LASERWRITER

The LaserWriter RS-422 port is designed only to be used with AppleTalk, not for direct connection. The direct-connect RS-232 port is wired DTE as follows:

2	Transmit data
3	Receive data
4	Request to send (optional, only if needed by host)
7	Signal ground
20	Data terminal ready (optional; only if needed by host)

When the LaserWriter starts up, it prints one sheet that gives its status (you can suppress this page with a short PostScript program).



SYSTEM ERROR CODES

You will sometimes see an error alert box containing a bomb icon. The alert box contains an error ID number that is generally useful only to the programmer, but if you report a problem, you should supply the ID number. For the curious, here are the error types:

1	bus error: doesn't happen on a normal Macintosh
2	address error: program referred to an odd address (usually trying to address a word on a byte boundary)
3	illegal instruction: the 68000 did not recognize an instruction
4	divide by zero
5	range check error: numerical range checking
6	overflow: integer number too large; such errors are normally trapped, but the trap failed
7	privilege violation: refers to 68000 modes not used in the Macintosh
8	trace mode error: for debugging programs
9	line 1010 trap: unlikely but serious system error
10	line 1111 trap: for debugging programs
11	miscellaneous exception error: all other 68000 exceptions
12	unimplemented core routine: undefined system code
13	uninstalled interrupt: Interrupts are signals that grab the CPU's attention when it is doing something else
14	I/O core: report of a very low-level input/output error
15	segment loader error: an attempt to read a program segment into memory from disk has failed
16	floating point error: a numeric computation problem
17–24	can't load package: missing segments of the system file, such as disk initializer, minifinder, and so on
25	out of memory
26	bad program launch: could not find a program, tried to run a nonexecutable file
27	file system map trashed: faulty information about the files on a disk
28	stack ran into heap: the stack, which passes parameters and allocates variables, has run into the heap, which holds the application program
30	disk insertion error: "please insert the disk"
31	no disk insertion
32–53	memory manager error
41	the file named Finder can't be found on the disk
100	can't mount system startup volume: couldn't read system resource file into memory
32767	general system error: "sorry, a system error has occurred"

DIAGNOSTIC TEST MESSAGES

The Macintosh has a built-in diagnostic test for hardware. To perform the test, install the INTERRUPT/RESET button. Hold down the INTERRUPT button and either turn the computer on or press the RESET button. If the computer passes the test, you will see a sad Mac icon with 0F 000D underneath. A list of possible results:

Class code	*Subcode*	*Explanation*
1		test failed
2	chip code	memory test: bus subtest
3	chip code	memory test: byte write
4	chip code	memory test: mod3test
5	chip code	memory test: address uniqueness
F		exception, after passing initial diagnostics; hardware traps in low memory
F	0001	bus error
F	0002	address error
F	0003	illegal instruction
F	0004	divide by zero
F	0005	check instruction
F	0006	trapV instruction; trap on overflow
F	0007	privilege violation
F	0008	trace: bit in trace bit; for debugging
F	0009	line 1010
F	000A	line 1111
F	000B	other exceptions
F	000D	non-maskable interrupt (normal result from diagnostic program)

Memory Chip Codes

The locations are stenciled like map coordinates on the edge of the printed circuit board.

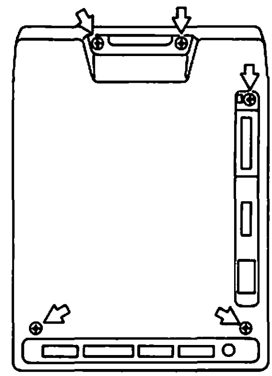
code	0001	0002	0004	0008	0010	0020	0040	0080	0100	0200	0400	0800	1000	2000	4000	8000
chip#	0	1	2	3	4	5	6	7	8	9	10	11	12	14	15	16
location	F5	F6	F7	F8	F9	F10	F11	F12	G5	G6	G7	G8	G9	G10	G11	G12

DISASSEMBLY

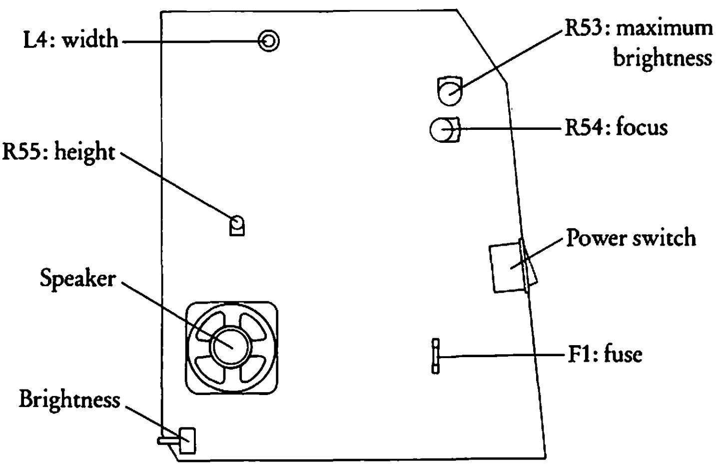
Don't attempt disassembly unless you have some technical experience and a good reason to open up the computer; you will void the warranty. The high-voltage power supply can be hazardous and you can get a dangerous shock even if the power cord is unplugged.

If you do disassemble your Mac, disconnect the power cord and remove the INTERRUPT/RESET button if you have installed it. Put the Mac face down on a soft cloth, and remove five screws (use a Torx T-15 screwdriver with a 6-inch-long shaft). One screw is underneath the battery cover. Removing the back cover requires some effort; two people make the job easier. Disconnect the floppy drive cable; watch out for sharp edges on a nearby bracket. To remove the CPU or logic board, unplug the power-supply cable; rock the board gently to disengage the locking mechanism. When you disconnect the power cable, the battery-powered clock stops; reset the clock after reassembly.

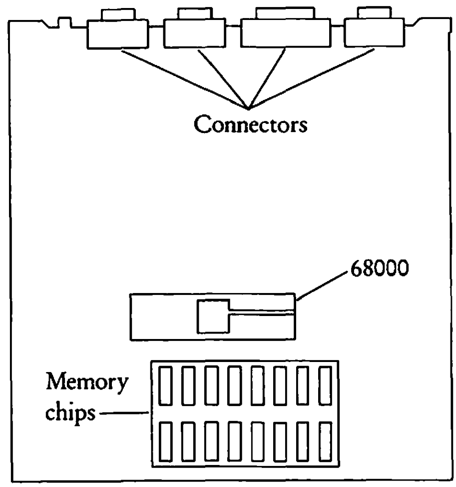
The analog circuit board, along the left side of the Mac, contains several controls and a fuse. Again, don't attempt any adjustments unless you know exactly what you are doing.



Cabinet screws



Analog circuit board with CRT adjustments



CPU board

If you have some electronics experience, *Macintosh Troubleshooting and Repair Guide* by Robert Brenner should be useful (Sams, planned for publication in fall 1985). I have not seen it yet, but Brenner's guides for the IBM PC and Apple II are valuable.

ENVIRONMENT

Operating temperature: 50 F to 104 F (10 C to 40 C)

Storage temperature: −40 F to 122 F (−40 C to 50 C)

Altitude: sea level to 15,000 feet (4500 m)

Temperature limits are set largely by the floppy disks and floppy disk drives. Altitude limits result from an electrical property of air: The breakdown voltage, or voltage at which a spark can jump across a gap, decreases with decreasing air pressure and density. Thus, at high altitudes, the high-voltage power supply driving the CRT will arc. As a practical matter, this is only likely to occur at a few weather and research stations or in unpressurized aircraft. (Above the altitude limit, you can disable the Macintosh CRT and high-voltage power supply and use a more suitable monitor with a video-output jack.) Similarly, the high-voltage coronas in the LaserWriter limit its operation to 8200 feet altitude (2500 m); above this altitude you will have to use a conventional printer. These are operational limits; you can ship these units in unpressurized aircraft without problems.

LISA/MACINTOSH XL

The Lisa was Apple's first visual-interface microcomputer. Like Mac, the Lisa used a 68000 CPU, but its hardware was organized differently from the Macintosh and required a special systems program—MacWorks—to configure it to run Macintosh software. Lisa used a 12-inch CRT with a 6¼- by 8¼-inch (16 by 21.5 cm) image area displaying 364 by 720 pixels. On the original Lisa, the pixels were oblong, 50 percent taller than wide (Macintosh pixels are square). Consequently, software that drew a perfect circle on the Macintosh drew an ellipse on the Lisa. Eventually the Lisa was renamed the Macintosh XL and a minor adjustment in the video circuitry made the pixels more square. The Macintosh XL was discontinued in 1985.

Glossary

Accessory card: Electronic circuitry printed on a hardware board (card) that adds features to a computer. The Macintosh does not take internal accessory cards.

Acoustic modem: In computer communications, a device that uses a small speaker and microphone to convert a computer's digital signals into sound and back again; allows transmission of information via telephone lines. Acoustic modems have special cups that fit snugly around the ear- and mouthpieces of a standard telephone handset.

Alert box: In Macintosh software, an urgent message on the screen, generally a warning. You must usually take some action before Macintosh allows you to proceed.

Analog-to-digital converter: An electronic circuit that changes continuous analog signals into discrete digital signals; abbreviated A/D. (See *digital-to-analog converter.*)

AppleTalk: A *local area network* developed by Apple Computer that can handle up to 32 *nodes* on a cable up to 1000 feet in length.

Application program: Software that does a particular task the user needs; word processors, spreadsheets, and graphics programs are applications. (See *utility program.*)

ASCII (text file): American Standard Code for Information Interchange; a standard scheme for coding characters as bits of data. Used by nearly all microcomputers and many large computers. (See *EBCDIC.*)

Asynchronous communications: A means of transmitting data that uses a special signal to indicate when each transmitted character starts and stops. The receiving computer therefore does not need precise timing information to read the data. (See *synchronous communications.*)

Audio-output port: In the Macintosh, the hardware and software connection that makes sound generation possible; connected either to the built-in speaker or to the ⅛-inch audio jack.

BASIC: Beginner's All-purpose Symbolic Instruction Code; the most common programming language for microcomputers. Although BASIC is relatively easy to learn, it is not especially flexible.

Baud, baud rate: In digital communications, one signal event per second. At low speeds (300 baud or less), one baud corresponds to one bit per second. At higher speeds, one baud codes two or more bits.

Bit: Binary digit; the most fundamental unit of digital information; represents either an electronic on (1) or off (0). One character is usually eight bits (one byte).

Bit-mapped image: A computer display technique where a special section of random access memory (RAM) is set aside for the screen; one bit in RAM corresponds to one dot (pixel) on the screen. (If the screen has color or grays, each pixel actually takes more than one bit.)

Boot: A computer's start-up process; begins with special start-up information stored in ROM. The term comes from "pulling yourself up by your own bootstraps."

bps: bits per second.

Brown-out: A period of low-voltage electrical power, usually caused by heavy demand or equipment problems at the electrical utility.

Buffer: A portion of memory that takes information from one device and feeds it to another; acts as a holding tank if the information from one device is coming in faster than it can be processed by the other.

Bug: A software problem. Named after a moth that caused the failure of an early (1945) digital computer at Harvard; evocative but inaccurate entomology: Moths (and butterflies) are in the order Lepidoptera, whereas true bugs are in the order Hemiptera.

Bulletin board, electronic: The computerized equivalent of a paper-based bulletin board, maintained on a computer and available via modem through the phone line.

Byte: A sequence of eight bits; usually equivalent to one character of information.

Cathode-ray tube (CRT): The screen used in nearly all televisions and most computers; an electron beam (the cathode ray) strikes a phosphor coating on the screen to produce light.

CCITT: Comité Consultatif Internationale de Télégraphie et Téléphonie (International Telegraph and Telephone Consultative Committee); an international standard-setting body, part of the International Telecommunications Union, which is part of the United Nations.

Central processing unit (CPU): A computer's main information-processing circuit. In a microcomputer, the CPU is a single silicon chip called the microprocessor or CPU chip; on larger computers, the CPU may consist of many chips.

Character width: In Apple usage for its communications programs, the number of bits, generally seven or eight, in a transmitted character; usually described as the number of data bits.

Chip: A tiny electronic circuit combining many components built on a base of silicon.

Clicking: For the Macintosh mouse, pointing at an object and pressing and releasing the mouse button once.

Clipboard: A region of random access memory that stores information you Copy or Cut while working with a Macintosh application. Clipboard information can be Pasted into files created with the same application program or with different programs. Clipboard contents remain intact until you replace them or turn off the computer.

Comma-delimited files: A data file in which commas separate data elements.

Command key (⌘): A special shift key to the left of the space bar, used to issue commands to Macintosh software; equivalent to the Control key on other computers.

Command-line interface: An old-fashioned way to get information into and out of a computer; displays commands a line at a time on the screen.

Communications protocol: The specific details governing how information is sent between computers.

Compiler: A program that translates a high-level programming language (source code) into machine code that the computer understands; an entire program is generally compiled as a unit. (See *interpreter* and *source code.*)

Control character: Normally non-printing ASCII characters that control operations or perform other functions. Control characters code for breaks between pages of text, tabs, line feeds, and so on. On the Macintosh, usually generated with the Command key.

Copy-protected disk: A disk that cannot be copied, usually because it is in non-standard format.

CPU: See *central processing unit.*

CRT: See *cathode-ray tube.*

CSMA/CD: Carrier Sense Multiple Access with Collison Detection; a network protocol that permits any node to send information when the network cable is free; if two nodes start at the same time, each node pauses for a random time period before trying again.

Cursor: The flashing marker indicating the current working location on a screen; called a pointer in Macintosh applications.

Cursor keys: A set of (usually) four keys found on many computers that can move the cursor in the four compass directions. The Macintosh keyboard has no cursor keys, although the accessory numeric keypad does.

Cylinder: On a disk, the aggregate of all the disk tracks available for a specific head position.

Daisy wheel: The printing element in one type of letter-quality printer; the element resembles a daisy with embossed letters at the tips of plastic "petals."

Data: Any form of information; the raw material that you process with a program.

Data base: A structured file of information, such as an address book, organized for storage, retrieval, and updating.

Data bits: Bits that code a transmitted character in computer communications. Seven or eight data bits usually make up a character; Apple refers to this number as *character width.*

Data disk: A disk that contains only data, no programs or start-up information.

Data fork: In a Macintosh disk file, generally the user data portion as distinct from the *resource fork* or *header.*

Data-phone Digital Service (DDS): A telephone service that handles digital signals directly without modems.

DCA: Document Content Architecture, IBM's format for transmitting text documents from one system to another; comes in two forms, the revisable *RFTDCA* and the final-form *FFTDCA.*

Desk accessory: One of several small programs in the Macintosh system file that can operate at the same time as an application program, including an alarm clock, note pad, calculator, and so on.

Desktop: The image on the Macintosh screen showing disk icons and disk windows; produced by the Finder.

Desktop file: A hidden data file created on all Macintosh disks by the Finder; contains housekeeping information the Finder uses to keep track of file folders, icons, etc.

Dialog box: In the Macintosh, a box on the screen asking for information that you must supply before the program can proceed.

Digital circuit: An electronic circuit that works with information coded in binary digits.

Digital-to-analog converter: An electronic circuit that converts discrete digital signals (bits) into continuous analog signals; abbreviated D/A. (See *analog-to-digital converter.*)

Digitizer pad: A computer accessory that registers the motion of a special stylus; used most often for graphics.

Digitizer, video: A hardware accessory that converts standard analog video signals into a digital form for computers.

Direct-connect modem: In computer communications, a device that plugs directly into a telephone line and converts a computer's digital signals into sound frequencies and back again; allows transmission of information via telephone.

Directory: A file stored on disk that indexes the location of information on the disk.

Disk buffer: A special portion of random access memory that temporarily holds often-used information (for example, the directory) from the disk; since the information does not have to be constantly read to and from the disk, operations are speeded up.

Disk controller: An electronic circuit that converts information on the microcomputer bus to a signal used by the heads in a disk drive.

Disk drive: An electro-mechanical device that records information on and plays it back from a magnetic disk.

Disk-drive port: In the Macintosh, a parallel port designed for connection to an external microfloppy disk drive.

Disk emulation: Using part of random access memory as an electronic disk drive; once set aside, this part of RAM emulates a normal disk drive but runs much faster. Not possible on the 128-KB Macintosh.

Disk server: A disk drive on a network, available to any user; only one user at a time can open any particular file (or, sometimes, logical volume).

DISOSS: Distributed Office Support System; developed by IBM as the standard information-exchange format for its office products; now used by other vendors; incorporates *DCA.*

Dot-matrix printer: A printer that creates characters and graphics out of small dots.

Double-clicking: For the Macintosh mouse, pointing at an object and quickly pressing the mouse button twice.

dpi: Dots per inch.

Dragging: For the Macintosh mouse, pointing at an object, then moving the mouse (and object) to another screen location while pressing and holding down the mouse button.

Duplex: In communications, two-way transmissions. Full duplex means two ways simultaneously, as with a conventional voice telephone; half duplex means one way at a time, as with citizens-band radio.

Dvorak keyboard: A keyboard layout devised by August Dvorak to improve typing efficiency.

EBCDIC: Extended Binary Coded Decimal Interchange Code; a scheme for coding characters as bits of data. Mainly used by large computers. (See *ASCII.*)

Echo: In computer communications, characters returned to the sender by the receiving computer. The echo mirrors the original transmission.

Electronic mail: A form of communication; the sender transmits a message to a central computer that stores the message until the recipient can retrieve it electronically.

Emulation: The technique of getting one product to mimic another.

Encryption: Converting information into an unintelligible form for security, most often with a password that acts as a key for later decryption.

Enter key: A special key on the Macintosh keyboard, often used to complete a keyboard entry.

Ergonomics: Considering the human element in engineering design. A truly ergonomic computer—both hardware and software—would work smoothly with people, but the standards are ill-defined; whether a device is "ergonomic" is usually decided by the advertising department.

Escape key: A special key on a computer keyboard that sends an "escape" code (ASCII 1B hex). A Macintosh needs an Escape key only when communicating with another computer; virtually all communications programs provide some way to send the escape code.

Expansion slot: A place inside the computer for accessory circuit cards; absent in the Macintosh.

Exponentiation: Raising a number (a) to a power (b); the number a is multiplied by itself b times. Two cubed is two times two times two, or two raised to the power of three.

Facsimile or fax: A method for sending pictures over ordinary dial-up telephone lines. The earlier analog Group II fax system uses 80 dots per inch resolution; the digital Group III fax uses 200 dots per inch. The latest Group IV has 400 dots per inch.

FFTDCA: Final-Form-Text Document Content Architecture, a text-file format that contains the final page image in electronic form. (See *DCA*.)

Field: In a database file, a unit of information; a collection of related fields constitutes a record. A name or zip code would be a field in a complete address record.

File server: A network *node* that contains a disk drive and processor that can handle multiple users working with the same data file or document; the processor and software control access so only one user at a time can change a specific record.

Finder: Macintosh software that manages files and disk directories.

Firmware: Programs embedded in a computer's circuitry; such programs cannot be changed as easily as a program on disk (software), but are not as fixed as the other electronic circuits (hardware). Read-only memory programs are often called firmware.

Formant synthesis: A common method of synthesizing speech, based on a small number of resonant or formant frequencies.

Frame store, frame buffer: In television technology, a memory buffer for video images.

Gigabyte: 1024 megabytes; sometimes 1000 megabytes.

Gray scale: Levels of shading used by display equipment or data files capable of displaying or recording the gray information associated with a pixel, instead of displaying or recording a simple on/off or black/white.

Handshake: In computer communications, an electrical signal used by the receiving device to stop transmission from the sending device until the data can be processed. The handshake between printers and computers allows the printer to catch up with the characters coming from the computer.

Hardware: The physical components of a computer—electronic parts, wires, cases, and so on.

Hayes-compatible modem: A modem that uses a standard set of commands developed originally by Hayes Microcomputer Products to set modes and features. The Apple modems are Hayes-compatible for most commands.

Header: In a Macintosh disk file, the directory entry and related information; gives the name, type, and source of the file.

Hierarchical data base: A data base whose information is organized in a tree-like structure.

High-level language: A programming language such as BASIC or Pascal that incorporates elements of English into its syntax.

I-beam pointer: The standard Macintosh text-editing cursor.

Icon: In the Macintosh, small graphic symbols that represent files or functions.

Impact printer: A printer that forms characters by striking an inked ribbon against paper.

Initializing (a disk): Placing address markers on new disks so the drive can locate information.

Ink-jet printer: A printer that forms characters or images by squirting tiny drops of ink onto paper.

Integrated software: Loose definition: software that can perform more than one task. Strict definition: software that can perform word-processing, spreadsheet, and database tasks and transfer information among tasks.

Interface: The common boundary between two entities, such as user and computer, printer and computer.

Internal modem: A modem built into a computer; not possible with the Macintosh.

Interpreter: A program that translates a high-level programming language into machine-readable code; the translation is done line by line. (See *compiler.*)

Kerning: In typography, the practice of printing specific letter pairs closer together than standard spacing for improved appearance.

Key disk: In a copy-protected program, the floppy disk you must use to start the program.

Keyboard macros: A software feature in which multiple keystrokes and sometimes mouse actions are assigned to a single command key.

Kilobyte (KB): 1024 bytes; the most common measure of computer file length or memory capacity. A typical double-spaced typewritten page is 1.5 KB.

Laser printer: A printer that forms images by scanning a laser beam across a photoconductive drum; after electrostatic development, the printer puts the image on paper with xerography.

Light pen: A computer pointing device shaped like a pen in which a light receptor senses the scanning beam across a CRT face. Not practical with the Macintosh.

Linked files, live links: Interdependent documents organized so that, when a change is made in one document, the dependent documents are automatically changed as well.

LISP: A high-level programming language used principally in artificial-intelligence research. LISP is the progenitor of the Logo programming language.

Local area network (LAN): A communications pathway that ties many *nodes* together so that any node can send a message to any other node.

Logical volumes: Results of a software-controlled technique that partitions a single physical disk drive into what appears to the user to be several disk drives or volumes (drawers in the HyperDrive).

Logo: A high-level programming language derived from LISP. Though simple functions are easy to learn, complex functions are fairly difficult.

Low-level language: A programming language (assembler) that is closely related to the intrinsic operation of a computer. Programs in assembler are fast and flexible, but also hard to read and write.

Macintosh, Charles: Scottish inventor (1766–1843) of waterproof rubberized fabric, used in mackintosh (with a "k") rain jackets.

Magnetic printer: A printer that uses an array of recording heads to create an image on a magnetic belt. Toner containing magnetic particles develops the image, which is then transferred to paper electrostatically.

Mainframe: A large traditional computer shared by many users.

McIntosh, John: Discoverer and cultivator of the McIntosh apple in Ontario, 1796. His name was misspelled by Apple when the Macintosh project began.

Megabyte: 1024 kilobytes; sometimes 1000 kilobytes.

Menu bar: The line with available menus at the top of the Macintosh screen.

Menu-driven interface: A user-computer interface in which menu selections usually take up the entire screen.

Menu-initial interface: A user-computer interface in which single letters indicate available menu choices.

Menu-word interface: A user-computer interface in which single words arranged in a line or two on the screen indicate available choices.

Microcomputer: A small computer designed in size and price to serve (mainly) a single user.

Microfloppy: A 3½-inch flexible disk within a semi-rigid plastic envelope; designed by Sony and used in the Macintosh.

Microprocessor: A single silicon chip containing thousands of electronic components, capable of manipulating information when operated in conjunction with accessory devices.

Mil: One-thousandth of an inch.

Millisecond: One-thousandth of a second.

Minifloppy: A 5¼-inch flexible computer disk, currently the most common in microcomputers.

Modem: An electronic circuit that converts digital signals into sound frequencies and back again for transmission by telephone lines.

Moire pattern: Images created by the juxtaposition of two repetitive structures, for example, the pattern you see looking through two railings on a distant bridge.

Mouse: A palm-operated pointing device used with the Macintosh and other computers that registers movement; contains a signaling switch, the mouse button.

Nanosecond: One-billionth of a second, or 0.000000001 second.

Network: An electronic communications pathway linking multiple computers and accessories such as printers and large disk-storage units. Any device can send a message to any other device on the network.

Node: In a network, any device capable of sending and/or receiving information.

Null-modem: A simple wire connection for communicating between two computers.

Object code: In programming, the executable code or machine-language program produced by the *compiler.*

Operating system: Essential software that acts as a traffic cop within a computer, directing information to and from different components.

Optical character reader (OCR): A machine that scans a typed or printed document optically, turning the text into computer code.

Optical disc (with a "c"): A recording and playback medium. To record, a laser beam makes tiny deformations on the surface of a reflective material; to play back, a laser beam plays across the surface, and a photodetector measures the reflectance interrupted by the deformations. (Some optical disc systems use other techniques.)

Optical fiber: A long thin strand of glass that carries information as a modulated light beam; can handle far higher communication rates than ordinary wire connections.

Option key: A special shift key on the Macintosh; similar to the Alternate key on some other microcomputers. Pressing Option at the same time as a character key produces characters for foreign languages or symbols.

Overlays: A program fragment stored on disk until needed by the main program core. With some large programs, only the main portion fits into random access memory; the overlays are brought in as needed for specific tasks, each overlay replacing ones no longer in use.

Pantograph: A mechanical device with a series of rods and joints that can reproduce drawings at different scales.

Parity: An error-detection technique that adds up the number of bits in a character or other unit; the result is usually noted as an even or odd number. Both sending and receiving devices compute the parity independently; a mismatch signals an error.

Pascal: A high-level programming language favored by many educators.

PBX: Private Branch Exchange; the private telephone switchboard within an organization.

Phoneme: A phonetic unit of language representing a single sound and used in speech synthesis.

Phosphor: Any material that emits visible light when struck by an electron beam; used in *CRTs*.

Pixel: A picture element, or single dot in an image. Sometimes elaborated into two forms: a pel consists of on/off or black/white information only; a pixel contains further attributes, such as gray level or color.

Plotter: A mechanical drawing device in which a pen and/or paper move in vertical and horizontal directions to create charts or other graphics.

PostScript: A computer language developed by Adobe Systems, Inc. to describe an image for printing; provides for text and graphics.

Power-line conditioner: An electrical filter designed to remove potentially damaging surges and brief high-voltage transients; installs between the AC power supply and an electronic device.

Printer buffer: A memory buffer between a computer and a printer. The computer writes characters into the buffer at high speed and is then free for other tasks; the mechanical printer reads information out of the buffer more slowly.

Printer port: The hardware and software that puts information destined for a printer on the wires physically going to the printer.

Printer server: A printer available to all users on a network.

Printer sound hood: A printer cabinet made with sound-deadening material.

Programming language: The words, symbols, numbers, and grammar used to give instructions to a computer.

Proportional spacing: Printing in which wider letters (such as M or W) take up more space than narrow ones (i or l).

Pull-down menu: A set of computer command choices that appears only when requested; until then, the menu titles alone appear on the screen.

RAM disk: A block of random access memory set aside to behave like a disk drive to the operating system; considerably faster than a physical disk drive.

Random access memory (RAM): Electronic memory that can be written to and read from.

Raw document: An application program's native data file; usually cannot be read directly by another application program.

Raw transfer rate: In a disk drive or on a network, the peak speed that information is transferred to and from the computer; usually much higher than the average transfer rate.

Read-only memory (ROM): Electronic memory that can normally be written to only once; a user cannot change its contents.

Relational data base: A data base in which any field or record can be associated with any other field or record.

Resource fork: A portion of Macintosh disk files that contains the program code, font information, and other data not normally generated directly by the user.

RFTDCA: Revisable-Form-Text Document Content Architecture; a text-file format with both text and formatting information; used for converting documents from one word processor to another. (See *DCA*.)

RS-170: Specification for a composite video signal compatible with broadcast standards in North America and Japan (straight video, not radio-frequency modulated).

RS-422, 423, 232C: Recommended standards (RS) for computer interfaces, from the Electronic Industries Association.

Scanner: A device that converts graphic images into a computer-readable form.

Scrapbook: In the Macintosh, a way to transfer information (text, picture, or other data) between files created with different programs. The Scrapbook operates like the Clipboard except that it is saved on disk. (See *Clipboard*.)

Screen dump: A pixel-for-pixel screen image printed on paper or stored in a disk file.

Scroll arrow, bar, box: On the Macintosh, the symbols along the borders of a window; show the window's position with respect to its contents and allow the user to change the position.

Sector: On a disk, the smallest contiguous unit of recorded information; several sectors make up a *track*.

Sector-reading program: Software that can read and change disk sectors directly; for experienced users.

Serial port: An electronic interface for computer devices that sends information in a sequential stream.

Server: In a network, any device that can be shared by all users.

Sheet feeder: A mechanical printer accessory that feeds paper, one sheet at a time, into the printing mechanism.

Signal-to-noise-ratio (S/N): The ratio (in voltage or power) of a received electric signal to the interfering noise; usually given in decibels.

Simplex: A communications channel, such as an ordinary radio or television broadcast, that always works one way only.

Socket: In AppleTalk, a specification within a single node that identifies the use of incoming data; used to distinguish electronic mail from program code, for example.

Software: The instructions that specify the operation of a central processor and other computer hardware.

Source code, source program: The original instructions (usually in a high-level programming language) that an *interpreter* or *compiler* turns into machine code for execution on a computer.

Spreadsheet program: A program that manipulates values laid out in a rectangular grid; the user specifies interrelationships among the values.

Start-up disk, boot disk: A disk with information necessary to start computer operations.

Stop bit: In asynchronous communications, a bit or bits added to mark the end of each character.

Structured data program: Any application program that stores information in a regular, defined way. A spreadsheet is a structured data program; a word processor or free-form graphics program is not.

Switcher: Apple's utility software that permits up to four programs to be resident in RAM simultaneously.

SYLK file: A data file using the SYmbolic LinK format developed by Microsoft; designed for business applications such as spreadsheets and data bases, and used for transferring data between otherwise incompatible programs.

Synchronous communications: A method of sending computer data in units of generally fixed size, with a timing (synchronizing) signal at the beginning of each unit. (See *asynchronous communications.*)

System disk: In the Macintosh, a disk containing the start-up and other utility information, including the Finder.

Tab-delimited file: A data file in which tabs separate data elements.

Thermal-transfer printer: A dot-matrix printer design that uses small heated pins to melt small dots of pigment onto paper.

Toner: The black powder that functions as ink in photocopiers and laser printers.

Touch pad: A computer pointing device that is operated by moving a finger over a flat surface.

Touch screen: A computer screen that allows the user to point at objects by touching the screen.

Track: A path on magnetic media for information storage. On a disk, tracks are concentric circles on the surface, made up of *sectors;* one or more tracks make up a *cylinder.*

Trackball: A computer pointing device with a large roller that the user turns.

Tractor feed or form feed: A device that transports form-feed paper through a printer.

Uninterruptible power supply (UPS): A power system that protects against power failures. A true UPS operates continuously from a battery, which is regularly recharged. A switcher, or standby, UPS has a battery and a fast switch; if the power fails, the system switches to the battery before the attached equipment can fail.

Utility program: Software needed to support a computer's operation rather than a user application. (See *application program.*)

V.22 bis: Modem protocol (data-link layer) for 2400 bits per second transmission over telephone lines; set by the *CCITT.* V.22 is a 1200 bps standard; the word *bis* is French for secondary.

Video-controller circuit: An electronic circuit that takes digital information and creates the signals necessary for displaying that information on a *CRT.*

Video RAM: A portion of random access memory set aside for buffering screen information. In the Macintosh, video RAM stores a bit-map of the screen display.

Visual interface: A modern computer interface using icons and other visually symbolic information instead of pure text.

Warm boot: The process of resetting a computer to its start-up state without shutting off the power.

Windows: In the Macintosh, a technique that partitions the screen display into several independently controlled regions.

Word wrap: Text entry display on a computer in which the software automatically advances to the next line at the end of a line; words are preserved as units.

Write-protect tab: A small part built into a disk jacket that is set to prevent accidentally erasing the disk contents.

Xerography: The most widely used photocopying technology, employing a photoconductive drum and electrostatic image development.

X-axis: The horizontal axis in a two-dimensional graph.

Y-axis: The vertical axis in a two-dimensional graph.

Index

Glossary entries are in **bold.**

G

H

I

CARY LU

Born in Qingdao, China, Cary Lu grew up in California and received an A.B. in Physics from the University of California–Berkeley and a Ph.D. in Biology from the California Institute of Technology. In addition to making a number of short films for *Sesame Street*, Cary has been an associate producer for *NOVA* and the science and technology advisor for The Children's Television Workshop. He is currently the microcomputer editor for *High Technology* magazine and a nationally recognized authority on office automation.

The first edition of The Apple Macintosh Book *was written on an IBM PC with SuperWriter software from Sorcim/IUS. (Macintosh software was not stable enough at the time.)*

The second edition was written on a Macintosh with a Corvus hard disk drive and Microsoft Word software.

Cover design and calligraphy by Tim Girvin; drawings by Rick van Genderen. Interior text design by Karen-Lynne de Robinson; adapted for second edition by John D. Berry and Greg Hickman.

Text composition in Stempel Garamond, with display in Helvetica Italic, by Microsoft Press, using the CCI-400 system and the Mergenthaler Linotron 202 digital phototypesetter.

Cover art separated by Color Masters, Phoenix, Arizona.
Cover printed by Philips Offset Company, Mamaroneck, New York.
Text stock, 60 lb. Glatfelter Offset,
supplied by Carpenter/Offutt; cover 12 pt. Carolina.
Printed and bound by R.R. Donnelley and Sons,
Harrisonburg, Virginia.

MacFortran™ is a trademark of Absoft Corporation. PostScript™ is a trademark of Adobe Systems, Incorporated. PageMaker™ is a trademark of Aldus Corporation. Linotron® is a registered trademark of Allied Corporation. FONTastic™ is a trademark of Altsys Corporation. Vu-Tek™ is a trademark of American Hoechst Corporation. TeX™ is a trademark of the American Mathematical Society. Apple® is a registered trademark, and AppleLine™, AppleTalk™, Finder™, Imagewriter™, LaserWriter™, Lisa™, MacDraw™, Macintosh™, Macintosh XL™, MacPaint™, MacProject™, and MacWrite™ are trademarks of Apple Computer, Incorporated. AlKaatib™ is a trademark of Arabic Software Associates, Incorporated. dBASE II® is a registered trademark of Ashton-Tate. Lock It™, Mac Memory Disk™, Mac Spell Right™, Mac Tracks™, and Mac Turbo Touch™ are trademarks of Assimilation Process. MAGIC™ and UNIX™ are trademarks of AT & T. MacPublisher™ is a trademark of Boston Software Publishers. CompuServe® is a registered trademark of CompuServe Information Service. MacLion™ is a trademark of Computer Software Design, Incorporated. ConcertWare™ is a trademark of ConcertWare. Corvus Concept™ and OMNINET™ are trademarks of Corvus Systems, Incorporated. MacForth™ is a trademark of Creative Solutions, Incorporated. Dasher® is a registered trademark of Data General Corporation. MacLink™ is a trademark of DataViz, Incorporated. MacCharlie™ is a trademark of Dayna Communications. MacTEP™ is a trademark of Dennis Brothers. 1st Base™, 1st Merge™, and 1st Port™ are trademarks of DeskTop Software Corporation. VAX®, VT52®, and VT100® are registered trademarks of Digital Equipment Corporation. CP/M-80® is a registered trademark and GEM™ is a trademark of Digital Research, Incorporated. Dow Jones Spreadsheet Link™ and Straight Talk™ are trademarks of Dow Jones & Company, Incorporated. ExperLisp™ and ExperLogo™ are trademarks of ExperTelligence. Factfinder™, FileMaker™, and Macware™ are trademarks of Forethought, Incorporated. HyperDrive™ is a trademark of General Computer Company. Quartet™ is a trademark of Haba Systems. Ensemble™, Hayden: Speller™, and I Know It's Here Somewhere™ are trademarks of Hayden Software Company, Incorporated. Smartcom II™ is a trademark of Hayes Microcomputer Products, Incorporated. MacLabeler™ is a trademark of Ideaform, Incorporated. MacDraft™ is a trademark of Innovative Data Designs. Intel® 8021, Intel® 8080 and Multibus® are registered trademarks of Intel Corporation. McBall™ is a trademark of Interfirm Systems Corporation. MacPhone™ is a trademark of Intermatrix. IBM® and Selectric® are registered trademarks, and PC-AT™, PC-DOS™, PC-XT™, and TopView™ are trademarks of International Business Machines Corporation. Mac488™ is a trademark of IO tech, Incorporated. Mac Private Eye™ is a trademark of I/O Video, Incorporated. Bernoulli Box™ is a trademark of IOMEGA Corporation. MacNosy™ is a trademark of Jasik Designs. Juki® 6100 is a registered trademark of Juki Office Machines Corporation. MacNifties™ is a trademark of KGI. LaserTools™ is a trademark of Knowledge Engineering. KAT...™ and MacVision™ are trademarks of Koala Technologies Corporation. NEON™ and Typing Tutor III™ are trademarks of Kriya Systems, Incorporated. Front Desk™ is a trademark of Layered, Incorporated. ThinkTank™ is a trademark of Living Videotext, Incorporated. Lotus® is a registered trademark, and Jazz™, 1-2-3™, and Symphony™ are trademarks of Lotus Development Corporation. MusicWorks™ and VideoWorks™ are trademarks of MacroMind. P/C Privacy™ is a trademark of MCTel, Incorporated. MegaFiler™, MegaForm™, and MegaMerge™ are trademarks of Megahaus Corporation. Typing Intrigue™ is a trademark of Micro Courseware Corporation. MacCOBOL™ is a trademark of Micro Focus. Microcom Networking Protocol® is a registered trademark and MacModem™ is a trademark of Microcom, Incorporated. Micron Eye™ is a trademark of Micron Technology, Incorporated. WordStar® is a registered trademark of MicroPro International Corporation. Microsoft®, MS-DOS®, Multiplan®, and XENIX® are registered trademarks, and MacEnhancer™ is a trademark of Microsoft Corporation. Crosstalk™ is a trademark of Microstuff, Incorporated. Motorola® 6800 is a registered trademark of Motorola, Incorporated. A+ Mouse™ and MouseSystems™ are trademarks of Mouse Systems Corporation. Viziflex Seels® is a registered trademark of Multiflex Seals, Incorporated. MultiMate™ is a trademark of Multimate, International. MITE® is a registered trademark of Mycroft Labs, Incorporated. Kaypro® is a registered trademark of Non-Linear Systems, Incorporated. Omni-Reader™ is a trademark of Oberon International. Helix™ is a trademark of Odesta Corporation. PeachCalc® is a registered trademark of Peachtree Software, Incorporated. Macbottom™ is a trademark of Personal Computer Peripherals Corporation. Polachrome CS™, Polagraph HC™, and Polapan CT™ are trademarks of Polaroid Corporation. PortaAPL™ is a trademark of Portable Software. ProModem™ is a trademark of Prometheus Products, Incorporated. OverVUE™ is a trademark of ProVUE Development Corporation. Sprint® is a registered trademark of Qume Corporation. TRSDOS® is a registered trademark of Radio Shack, a division of Tandy Corporation. UCSD p-System® is a registered trademark of the Regents of the University of California/Softech Microsystems, Incorporated. Macontrol™ and Macquisition™ are trademarks of Small Business Computers of New England, Incorporated. p-System™ and The MacAdvantage: USCD Pascal™ are trademarks of SofTech Microsystems, Incorporated. Decision Map™ is a trademark of SoftStyle, Incorporated. TK! Solver® and VisiCalc® are registered trademarks, and DIF™ is a trademark of Software Arts, Incorporated. SuperCalc® is a registered trademark of Sorcim Corporation. The Nod™ is a trademark of Stride Micro. MacTablet™ is a trademark of Summagraphics Corporation. Symbolics 3600™ is a trademark of Symbolics, Incorporated. FastFinder™ is a trademark of Tardis Software. Tektronix® is a registered trademark of Tektronix Corporation. Filevision™ is a trademark of Telos Software Products. EtherMac™, 3Com™, and 3Server™ are trademarks of 3Com Corporation. ThunderScan™ is a trademark of Thunderware, Incorporated. ClickArt™ is a trademark of T/Maker Graphics Company. True BASIC™ is a trademark of True BASIC. Tymnet® is a registered trademark of Tymshare, Incorporated. Mail Center™ is a trademark of Videx, Incorporated. Wang Professional Image Computer™ and WangNet™ are trademarks of Wang. Diablo® 630 and Xerox® are registered trademarks, and Ethernet™ and Smalltalk™ are trademarks of Xerox Corporation. Z-80® is a registered trademark of Zilog, Incorporated.

"The best book I have seen so far [on the Apple Macintosh] is THE APPLE MACINTOSH BOOK by Cary Lu."

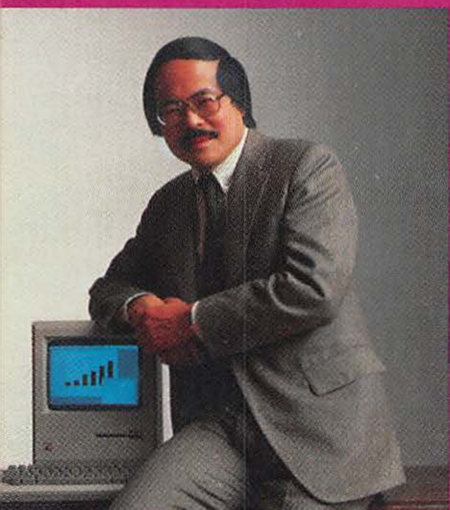
Erik Sandberg-Diment,
The New York Times

"Lu's is certainly the best reference I've seen, and I recommend it to anyone who either has a Mac or is thinking of getting one."

Jerry Pournelle,
BYTE

"It is a veritable fountain of information about the Mac and its software."

Jeffrey Bairstow,
Personal Computing



The Experience Continues!

Here is a timely new edition of the book that quickly became the most indispensable book on the Apple Macintosh. Cary Lu, microcomputer expert, takes you behind the scenes and gives you a comprehensive look at the Mac's pace-setting technology, its philosophy, and the hardware that is now available. And he sorts through today's new Mac software to help you evaluate what's right for you.

If you already own a Mac or are thinking of upgrading to a 512K, this book will enhance your pleasure and productivity. If you are deciding whether to buy a Mac, this book provides all the information you'll need for comparison shopping.

Cary Lu observed the development of the Mac firsthand, sharing the excitement of its evolution with the designers at Apple and the programmers at Microsoft. In THE APPLE MACINTOSH BOOK, 2nd edition, he invites you to continue the Mac experience.

The Apple Macintosh Book has been translated into French, Italian, German, Japanese, Swedish, Norwegian, and Dutch.

U.S.A.	**$19.95**
U.K.	£16.95
AUST.	$29.95 (recommended)
CAN.	$29.95

MICROSOFT PRESS

0-914845-66-7