BANTAM COMPUTER BOOKS

THE Bantam Apple IIGS Library THE Bantam Apple IIGS Library THE Banta

THE Apple® IIGS Book

Jeanne DuPrau and Molly Tyson
of Apple Computer, Inc.

THE Apple® IIGS Book

THE Apple® IIGS Book

Jeanne DuPrau
and Molly Tyson

of Apple Computer, Inc.

BANTAM BOOKS
TORONTO • NEW YORK • LONDON • SYDNEY • AUCKLAND

APPLE IIGS BOOK
A Bantam Book/October 1986

Cover design by J. Caroff Associates.
Interior design by Nancy Sugihara.

Throughout this book, the trade names and trademarks of many companies and products have been used and no such uses are intended to convey endorsement of or other affiliations with the book.

ISBN 0-553-34359-9

Published simultaneously in the United States and Canada

Bantam Books are published by Bantam Books, Inc. Its trademark, consisting of the words "Bantam Books" and the portrayal of a rooster, is registered in U.S. Patent and Trademark Office and in other countries. Marca Registrada. Bantam Books, Inc., 666 Fifth Avenue, New York, New York 10103.

PRINTED IN THE UNITED STATES OF AMERICA

B *0 9 8 7 6 5 4 3 2 1*

Acknowledgments

We couldn't have written this book without the support, encouragement, and indulgence of the people we work with. Special thanks to Neal Amsden, Gus Andrade, Mike Askins, Fern Bachman, Jim Batson, Eagle Berns, Kathie Blair, Peter Baum, Kendra Brotherton, Rain Burns, Rani Cochran, Lee Collings, Linda Curry, J.D. Eisenberg, Chris Espinosa, Sue Espinosa, David Eyes, Karyn Gray, Peter Green, Adam Grosser, Dan Hillman, Jim Jactzinski, Rich Jordan, Carol Kaehler, Ted Kaehler, Scot Kamins, Donna Keyes, Gary Knutson, Harvey Lehtman, Leslie Liedtka, Bob Loftis, Mike McGrath, Jane McKean, John Medica, Ray Montagne, Rob Moore, Jan Nelson, Don Porter, Todd Rudin, Curtis Sasaki, Mark Seymour, Nancy Stark, Martha Steffen, Larry Thompson, Bruce Tognazzini, Jeff Vasek, Jennifer Wallace, Joe Williams, Rich Williams, Jill Worsham, John Worthington.

And to the people at home who didn't know what the book was about but found ways to help us write it.

Some of the People Who Made the Machine

Table of Contents

CHAPTER FIVE

Peripherals

Peripherals you can attach to your Apple IIGS, ways you can attach them, and compatibility issues.

CHAPTER SIX

Control Panel Program

Using the Control Panel to customize your system.

CHAPTER SEVEN

Basic Computer Concepts

For computer beginners—explanations of basic terms and operations.

Introduction

The Apple II computer has always had a good reputation. It's powerful without being threatening, reliable without being boring, playful without being a plaything. It's the kind of machine you want your kids to meet; the kind of machine you could live with for a long time. And many people have.

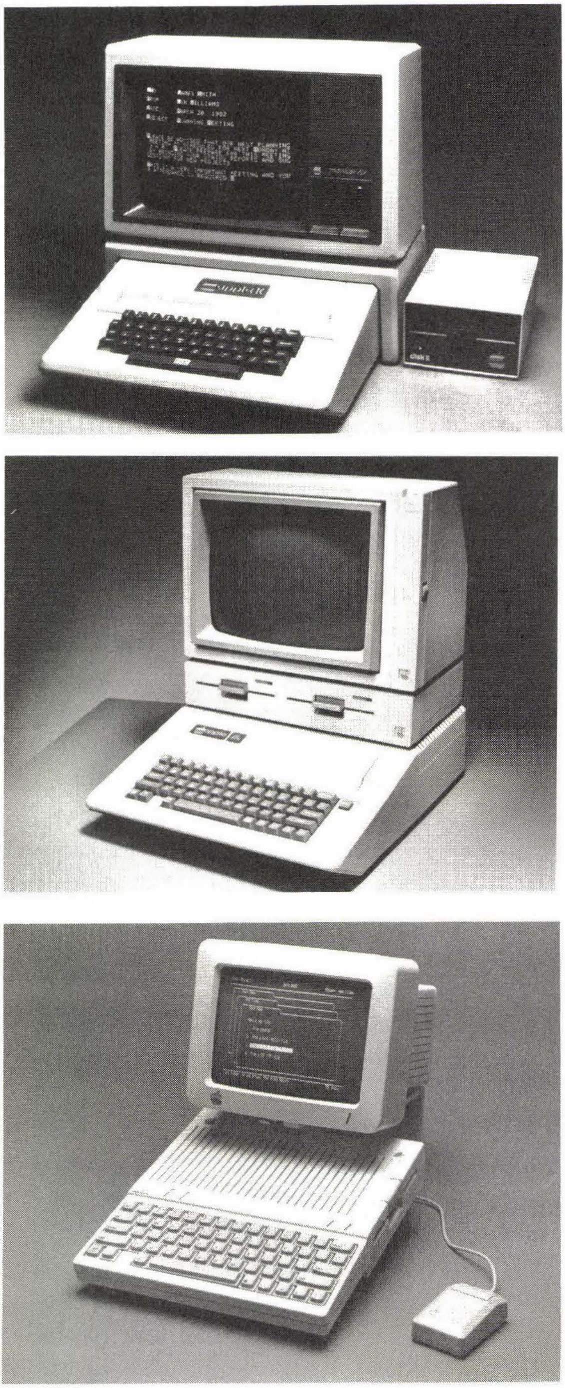
The Apple II has been around, in one form or another, since 1976. The original Apple II had just 4K of memory, but that was enough to write your own simple programs, and you could increase that 4K to 16K by plugging more RAM chips into empty sockets on the circuit board. And 16K was enough to play *StarTrek,* the hit program of that era. Programs came on cassette tapes then, not disks, and the standard display was a television set.

When the Apple II Plus came along in 1979 with 48K, it was the only computer with enough memory to run *Visicalc,* the spreadsheet program that gave personal computers sudden respectability among nonprogrammers as "productivity tools." With this new role, there was a demand for a more businesslike keyboard (previous models' keyboards produced only capital letters), for a display that looked more like a typed page, and for more memory.

The Apple IIe, introduced in 1983, delivered all that and more. And it's still selling strong today along with the Apple IIc, a compact version of the Apple IIe that has all the most popular features built in or easy to add on.

Now there's a new Apple II on the market—the Apple IIGS. In many ways it behaves like the other members of the Apple II family; if your kids are used to using an Apple IIe at school, they'll be able to use the Apple IIGS too, right away; if you have some favorites among the thousands of Apple II software programs, you can run them on the new machine just as you did on the old one. This family resemblance makes the Apple IIGS comfortable to use. There's another side to its personality, though. High speed, spectacular color and sound, and the new mouse interface give it a flash and power that the old Apple II can't approach. These new features alone are probably rea-

Figure I ▪ 1
Apple II family



son enough to buy an Apple IIGS. But it's the combination of the old world and the new—which has been accomplished not absolutely seamlessly but still with amazingly few compromises—that makes the Apple IIGS unique.

Because of its split personality, talking about the Apple IIGS is almost like talking about two different computers. We'll concentrate in this book on what's new about it; plenty of books have already been written about the older generation. Still, you need a sense of the Apple IIGS's heritage to understand fully all it offers. What has accounted for the Apple II family's enormous popular appeal?

Lots of Software

The Apple II was the first personal computer on the market—at least the first that you didn't have to solder together from a kit—and every improvement since then has been compatible with the model that came before. That means that all the software developed for all the variations of the Apple II over the years runs on the Apple II's being built today. That's a lot of software on a lot of subjects. It pretty much guarantees that you can find the exact program you need, whether it's for keeping school attendance records, laying out a vegetable garden, or planning investments for clients. And there are new programs coming out every day. The more people there are using Apple II's, the more incentive there is for software developers to keep writing new programs for the Apple II family.

Lots of People Use Them

There's a strong sense of community among Apple users. There's an incredible outpouring of support for new users—a desire to share the good experiences and protect new users from the bad experiences that go along with learning anything new. You don't have to wait for chance to throw you into contact with this community of supportive, experienced users. You can meet them by joining an Apple Users Group. Call 1-800-538-9696, extension 500, to find out the phone number of the computer club nearest you. It might be a good idea to go to a meeting *before* you buy a computer so you'll know what you're getting into before you invest in an Apple IIGS.

Sometimes the Apple sense of community is carried to outrageous extremes. Apple is competitive with the best-stocked college bookstore in terms of what you can buy to proclaim your affiliation. You can get T-shirts, baseball caps, beach towels, decals, mugs, wine glasses, windbreakers, even a wind-surfing sail emblazoned with the Apple logo. But when you need help getting your printer to work with your accounting program, you don't care if the person who solves your problem is wearing sunglasses with rainbow lenses shaped like Apples. It's nice to be part of a nurturing community.

The Kids Use Them at School

Your kids probably use Apple II's at school. This is good and bad. Kids will tease you a lot when you're learning because they will mistake your thoughtful deliberation for fear and caution, but children come in handy when you need help setting up your spreadsheet.

It Has Personality

Then there's Apple's colorful history. The company was started in a garage by a couple of college dropouts named Steve Wozniak and Steve Jobs. Jobs sold his Volkswagen bus and Wozniak sold his Hewlett-Packard calculator to buy the parts for their first build of 150 computers. Those first 150 machines didn't have cases or keyboards, but nobody seemed to mind.

The company made $200,000 the first year, $7 million the next year, and over $300 million the fifth year, to become the youngest company in the Fortune 500. Wozniak and Jobs are the Horatio Algers of the personal computer industry.

"Old Technology"—Proven Technology

In recent years the main argument against getting an Apple II has been that the technology is ten years old. Despite enhancements to the original Apple II that kept the memory size and the graphics competitive with those of newer computers, there was some truth to the argument. What was *revolutionary* in 1976 was more *evolutionary* in 1986. Apple still had credibility as a state-of-the-art company because of the Macintosh, with its distinctively crisp display and mouse-based interface. But the only thing the Apple II had in common with the Macintosh was its logo—the rainbow-colored apple with a bite taken out of one side.

With the introduction of the Apple IIGS, the "old technology" argument loses its bite. The Apple IIGS can run almost all the software designed for earlier models of the Apple II, yet it's a state-of-the-art

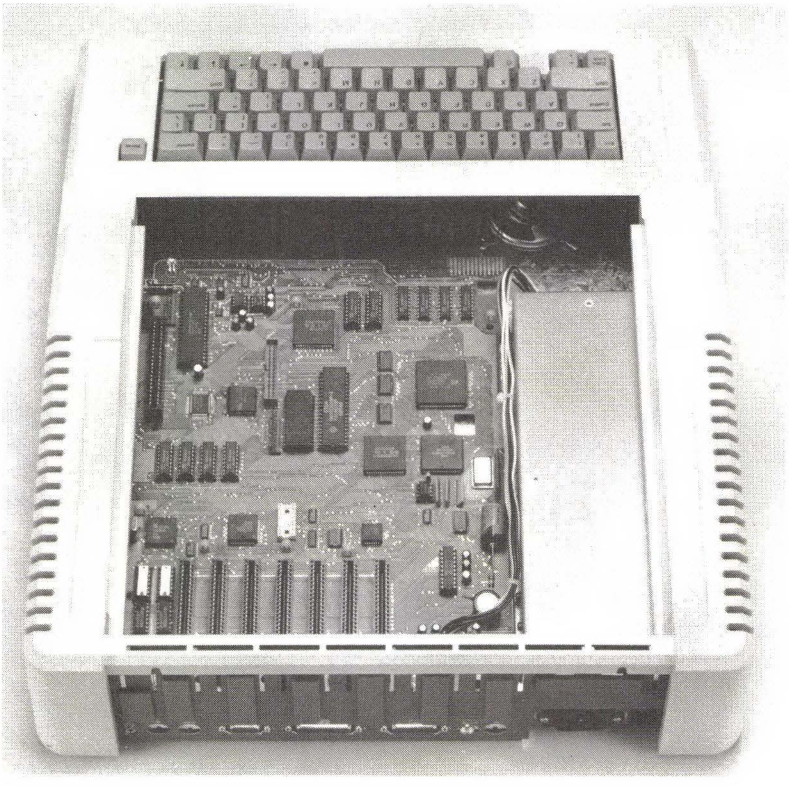
***Figure* I ▪ 2**
Apple IIGS under wraps

computer in terms of processing speed, memory size, screen resolution, color options, and especially sound. The GS stands for Graphics and Speed, but it could also stand for Great Sound. If you haven't seen and heard the Apple IIGS in action, go down to your local computer store and take a test drive. You won't believe you're using an Apple II.

Good News for Apple IIe Owners

If you already own an Apple IIe, you can bring your computer in for a "board-lift." For about $600 your Apple dealer will replace the main circuit board and back panel of your Apple IIe with an Apple IIGS board and back panel. (See Figure I.3.) It won't look like an Apple IIGS, but it

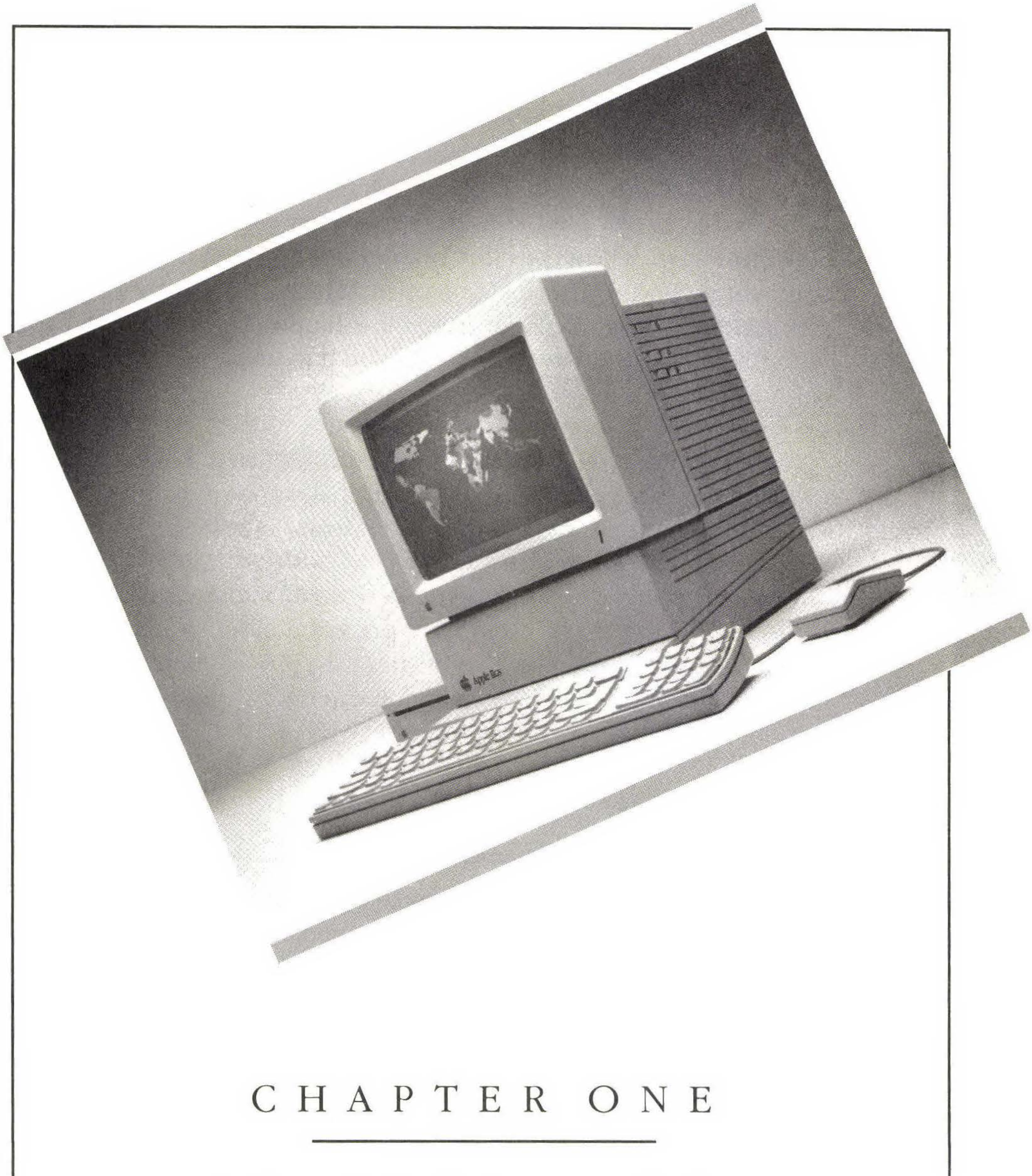
***Figure* I ▪ 3**
New board in the old "box"



will behave like one. Also the mouse, detached keyboard, and mouse-based utility disk that come as standard equipment with the Apple IIGS are available separately for upgrade customers.

About This Book

Most of this book is aimed at people who have had some exposure to computers. If you're brand new to these machines that Steve Jobs calls "bicycles for the mind," start with Chapter 7. It's the training wheels for the bicycle. The rest of the book will make a lot more sense once you understand the jargon and have a feel for what it's like to use a computer for writing, record-keeping, financial modeling, drawing, learning, and playing.



CHAPTER ONE

The Making of the Apple IIGS

Two of the early code names for the second generation Apple II were Brooklyn and Golden Gate. This new machine was going to be a bridge between the first generation of personal computers represented by the Apple II and the second generation represented by the Macintosh. The key to this bridge was a processor under development at Western Design Center in Arizona called the 65816 (816 for short). The 816 could emulate the 8-bit processor in the Apple II but also had a 16-bit mode that enabled it to use up to 16 megabytes of memory.

A computer's processor is sometimes described as its brain, but processors don't really think. They just carry out the instructions in a program very, very quickly. There are all sorts of technical differences between 8-bit and 16-bit processors, but the important distinction is that an 8-bit processor can address (use) only 64K of memory directly while a 16-bit processor can address up to 16 megabytes. The amount of memory a program can use is important, because it determines how sophisticated a program can be and how big the document you're working on can be.

The Development

The code name that leaked to the press wasn't Brooklyn or Golden Gate; it was Apple IIx. The "x" stood for expanded, extended, extraordinary, no one knows for sure, but that was the name Steve Wozniak used when he discussed the next generation Apple II with reporters and users groups. It's not ordinary Apple policy for employees to talk about products under development, but Wozniak was no ordinary employee. He was the co-founder of the company, the inventor of the original Apple II, and he subscribed to the hacker ethic that information is for sharing. (See Figure 1.1.)

That ethic had certainly paid off with the Apple II. When the original Apple was under development, Wozniak reported every new feature he came up with to his friends at monthly meetings of the Homebrew Computer Club, and in so doing he inspired the development of thousands of programs and hundreds of peripherals for the new machine. Before Apple computers were available in stores, Wozniak would even go to people's houses and help them build their own versions of his invention.

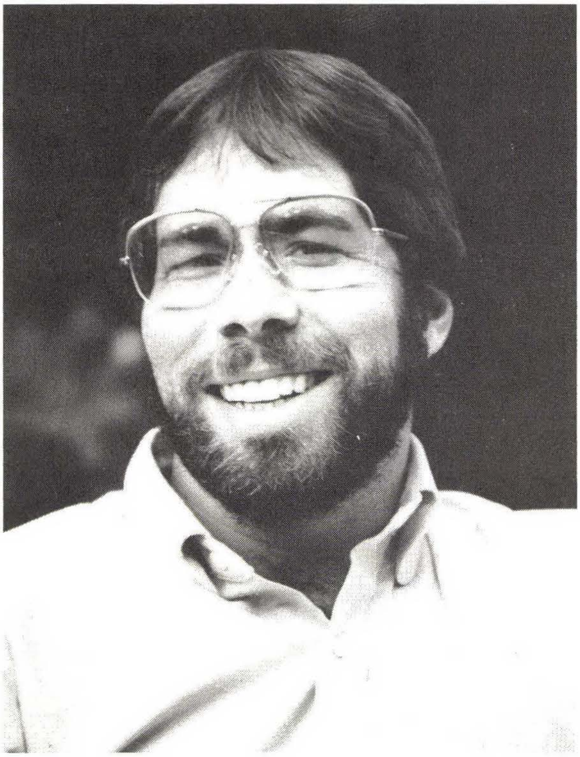


***Figure* 1 ▪ 1**
Wozniak

Wozniak took a break from Apple in 1981. He had stuck around long enough to help the Apple II become the Apple II Plus and to see the company grow from a startup to a Fortune 500 company, but he wanted to finish his bachelor's degree at Berkeley. He had planned to be gone from Apple for only a year—just long enough to get his diploma—but he got sidetracked sponsoring a couple of rock concerts. (He finally received his degree in 1986.) By the time he returned, it was 1983 and Apple was a different place. The company was no longer selling the Apple II or the Apple II Plus he had designed, but an enhancement called the Apple IIe, designed to appeal to the home and business users who were starting to outnumber the hobbyists and hackers who were the early Apple owners. The Apple IIe had uppercase and lowercase characters (the original was uppercase only), 64K of built-in memory that could easily be increased to 128K, and an easy way to change the standard 40-character-per-line display to 80 characters per line. So the Apple II was still going strong after seven years, even though most of the research and development money at Apple was going into other product lines.

Apple was divided into product divisions then—one working on the Lisa (a very expensive, very innovative computer that pioneered the mouse-based interface for personal computers), one working on the Macintosh (the computer that would offer Lisa technology at a more affordable price), and one working on variations of the Apple II. Apple co-founder Steve Jobs and Wozniak's closest friends were all in the Macintosh group by this time, but Wozniak returned to the Apple II division, where he thought he could be the most useful.

Most of the people in the Apple II group were working feverishly to finish up the Apple IIc, a compact version of the Apple IIe featuring a built-in disk drive and ports for connecting peripheral devices instead of the slots that were part of the original Apple II design. It was a closed system—that is, you couldn't open the lid and attach things to the main circuit board as you could with the original Apple II and the Apple IIe.

The thinking was that ports would be less threatening to people who were buying the computer as an appliance for word and number processing, not as an electronic erector set. With slots you have to open the top of the computer and come face to face with the circuitry, which to the uninitiated seems like an activity that will, at best, void your warranty and at worst electrocute you. With ports you can remain ignorant of chips and resistors and limit your interaction with the computer to the words and numbers on the screen.

So the Apple IIc was designed as a plug-and-go system, in which everything needed was either built in or designed to be plugged into the back. So was the Macintosh. Everything you could possibly want was built in or easy to add on; you couldn't open the lid to add more. In other respects, the Macintosh had nothing in common with the Apple IIc.

Wozniak was leery of this kind of thinking. Not that he's against built-in features. He believes that the secret to the original Apple II's success was not the slots—everybody was doing that, he says—bùt the built-in features. "No other computer had built-in color," he says. "No other computer had graphics. No other computer had BASIC. There were a whole ton of things in our computer when you bought it. The other computers gave you a totally blank system with a ton of slots." To Wozniak, slots aren't what sell a machine. However, they are an important protection against obsolescence.

"The only argument Steve Jobs and I ever had about the design of the Apple II," says Wozniak, "was about slots. I said there should be eight and he said there should be two. So we compromised. I said I wouldn't start the company unless there were eight slots."

That was the state of affairs at Apple when Wozniak made his quiet reentry in July 1983. "I don't think Steve Jobs or John Sculley (Apple's president) even knew I was back for a few weeks," says Wozniak. "I went straight to Dave Paterson, who was head of Apple II engineering, and asked if he needed another engineer." Paterson referred the engineering job applicant to engineering manager Dan Hillman, who would soon head up the hardware effort on the Apple IIx project.

"He was absolutely clear that he wanted to be treated just like any other engineer," Hillman recalls. "He wanted to be paid like an engineer, he wanted an office just like an engineer's, everything like the other engineers. So I called up human resources and said, 'We have a real dilemma here. I have to hire the founder of our company. I'm not sure what salary to offer him and I'm not sure what position to put him in and does he get stock options?' "

So Wozniak helped with the Apple IIx and talked to users groups about it. This generated a lot of interest in the IIx outside Apple, but the project never caught on inside Apple. In hindsight, the trouble with the Apple IIx was that it was trying to be too many things to too many people, and no one was trying very hard to make it an Apple II. Almost all of the marketing interest and engineering effort focused on putting in a co-processor slot that would make it possible for the machine to run software designed for the Macintosh or the IBM PC. (A co-processor is a card containing the processor from another kind of computer. By using the power and memory of the host computer, a co-processor gives you two computers in one.) To the executive staff the IIx was an insurance policy. If the Macintosh didn't work out, Apple would have an alternative machine to position against the IBM PC.

As it turned out, the Macintosh didn't need a pinch hitter and the 816 fell behind schedule, so no one was surprised when the Brooklyn/Golden Gate project came falling down like one bridge it wasn't code-named after. In fact, the team leaders were the ones who suggested to the executive staff that the project be cancelled.

For six months after that there was no more talk about a 16-bit Apple II and Hillman got reassigned to a more mundane task—cost-reducing the Apple II. He and engineer Jay Rickard cost-reduced it all right. They figured out a way to put almost the whole Apple II on a single chip called the Mega II. (See Figure 1.2.)

"When we designed the Mega II, we didn't know what machine it

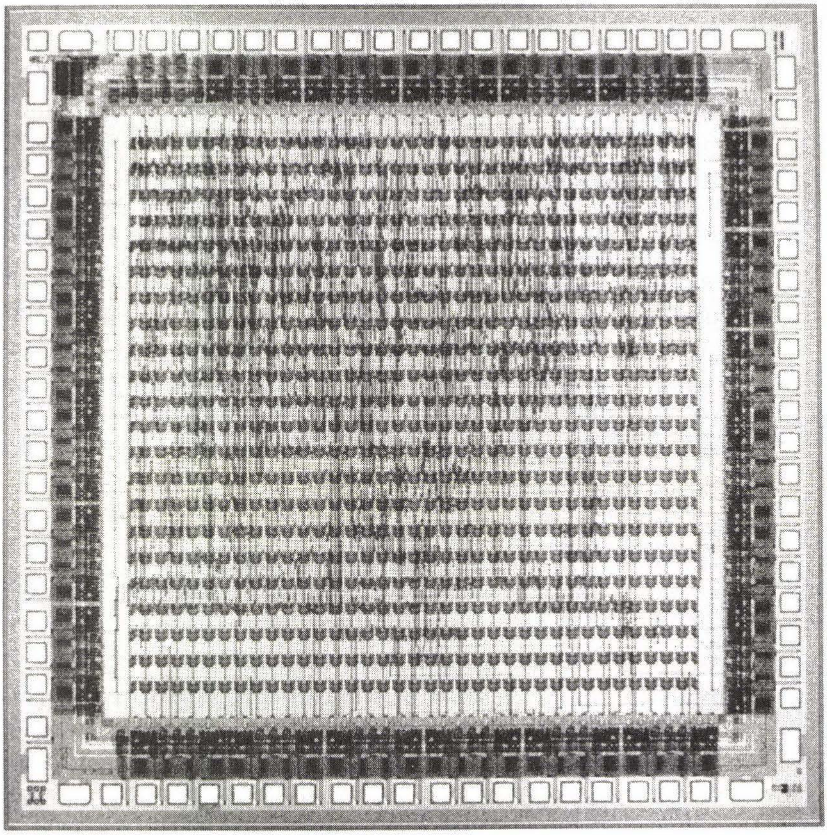


***Figure* 1 ▪ 2**
Mega II

was for," Hillman recalls, "but all the talk then was about the Apple IIc. At that time the Mega II just seemed like a way to make the Apple IIc smaller."

Return of the Apple IIx

What brought the IIx back to life? Partly it was the 816. It was finally ready. But mostly it was Apple II sales. The introduction of the Apple IIc in April 1984, with its marketing message "Apple II Forever," had revived interest in the whole Apple II product line, and Apple II sales were going through the roof. "They priced the IIe below the IIc, so it wasn't clear if people wanted an open Apple II or the cheaper Apple II," says Hillman, "but it was clear that the open system didn't scare them." Maybe there was a market for a high-end, expandable Apple II after all.

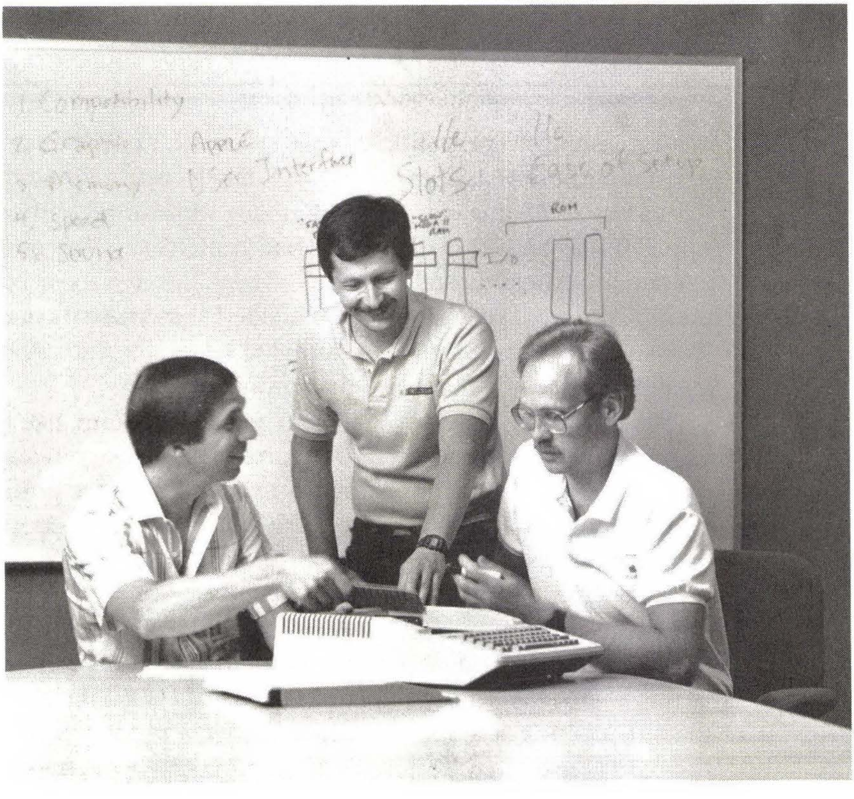
Hillman was at a meeting waiting to hear whether the Mega II would be used to cost-reduce the Apple IIe or the Apple IIc when Wozniak suggested reviving the 16-bit Apple II project. "My first reaction," says Hillman, "was 'Oh God, I think I've been through this before.' " But there was no arguing with Apple II sales. So the IIx rose from the ashes with the code name Phoenix.

Hillman had learned a lesson from the IIx project and from the Apple II sales. This time the emphasis was on the II-ness of the machine, not just the new-ness.

The first few planning meetings involved Wozniak, Hillman, Harvey Lehtman from the system software group, and Lee Collings from marketing. (See Figure 1.3.)

Collings is not your typical marketing type. "He's more of an honorary engineer," says Lehtman. Collings was managing a hi-fi shop

Figure 1 ▪ 3
Hillman, Lehtman, and Collings



when he saw his first Apple II in 1978. It was serial #5 and it came with a photocopied manual in a looseleaf binder. "Most of it was typed," Collings recalls, "except for some notes about hi-res graphics that Wozniak had written in by hand."

Collings joined Apple in 1979 to do dealer sales support and later marketing, but he was always a hacker at heart. "I still have all the old BASIC programs I wrote," he says. "They're on disks now, but they were on cassette tape when I first wrote them."

Collings drew on his intimate, long-term relationship with the Apple II in preparing the Apple IIGS's marketing requirement document (MRD)—the wish list that is marketing's contribution to the development of a new product. "The basis of that document," he says, "was what I wanted in an Apple II. Having lived with the II, the II Plus, the IIe, and the IIc, I knew what I wanted and I knew what a lot of my friends wanted so I just put it down on paper."

But Hillman wanted something more basic than an MRD. He wanted a mission. "I told them I,'d build anything they wanted no matter how crazy it was," says Hillman. "But I had to have a mission."

The Hardware Mission

"Our mission was very simple," says Hillman. "First, we wanted to preserve the Apple II as it exists today. It had to work with Apple IIe and IIc software. That was goal number one. But we recognized that the Apple II was an old computer. It had limitations. The new machine needed to address those limitations—break through those barriers, and the barriers were very obvious: We needed to increase the memory size. We had to make it run faster. We needed better graphics. And we had to have better sound. That was our mission."

"We also wanted the Apple IIGS to be easy to set up, like the IIc, and easy to expand, like the IIe," Lehtman recalls. That goal created some interesting challenges for Lehtman's group because the ports on the IIc emulate particular slots on a IIe. With both ports and slots, peripherals would face some serious identity crises without clever intervention from the system software.

Wozniak was heavily involved in designing the architecture of the Apple IIGS and adamant about keeping the design simple. "Woz likes to minimize headaches," explains Hillman. "He thinks that if you have to keep looking at a book to figure out what mode you're in, then it's too hard to use." In the interests of simplicity, he argued against putting a co-processor in the Apple IIGS. "He thought it wrecked the architecture of the machine," says Hillman. Also it was Wozniak's idea to keep the 8-bit world totally separate from the 16-bit world. Rather than perpetuating the strange graphics modes and memory-addressing schemes that made sense in 1976 but not in 1986, he suggested that they rope off 128K of the 256K built into the machine and designate it as "slow RAM." Everything in slow RAM would work the way it does on earlier Apple II's. The other 128K and the memory on the memory expansion card would be fast, clean, and efficient.

"Woz came in and drew it all up on the board in about two minutes," Hillman recalls. "It looked like a real mess but it was very clear in my mind." Laura Basso Roebuck, the group's secretary, came in, took one look at the board and said, "What's that?" "That," said Hillman, "is the future of Apple computer."

They hadn't solved the graphics problem yet, or the sound problem, but they had figured out a way for new programs to use a lot of memory (up to 16 megabytes) in a clean, efficient way without interfering with compatibility of old programs, and they had figured out a way for new programs *and old programs* to run three times faster than they could on an Apple II. The key to all of this was a custom chip called the FPI, the Fast Processor Interface, that would sit between the Mega II and the 816 and route tasks to the appropriate chip.

Just when all the hardware pieces were starting to fall into place, Wozniak left Apple again to start up a new company called CL9 (for Cloud 9). Wozniak concedes that his timing was strange, but he has become used to his life taking unexpected turns. Like his computer designs, Wozniak "likes to be flexible."

"I credit Woz a lot on the initial design of this product," says Hillman. "He helped design the architecture and then poof, he disappeared. He got interested in this little remote control device and that appealed to him more than slogging through the mess of getting a computer out."

"The Apple IIGS's design was beautiful," says Wozniak," but I had gotten turned on to a couple of projects that weren't computers. The big word at Apple then was 'productivity, productivity, productivity.' Everything was being sold as a productivity tool. Well, I don't think you have to be productive all the time. I wanted to work on nonproductivity products, including video editing systems and a little remote control thing. I wanted to do those things, and I had to leave Apple to do them, but I was very happy with the Apple IIGS and with the product definition and with the people working on it."

If oldtimer Wozniak was instrumental in solving the memory-addressing problem, it was newcomer Rob Moore who solved the graphics problem. The resolution they were shooting for was 640 by 200 in 4-color mode and 320 by 200 in 16-color mode. Those are the number of dots, or pixels, that make up a very high-resolution video display. The more pixels, the sharper the picture. The competition was doing 640 by 200 and 320 by 200, and that's what the 16-bit Apple II had to do, while maintaining compatibility with all the earlier graphics modes.

Actually, the Amiga from Commodore and the Atari ST were boasting a 640 by 400 display and that's what Collings was pushing for. "I got shot down on that one," says Collings, "but for a very good reason."

"Commodore gets a horizontal resolution of 400 lines using a technique called interlacing," Lehtman explains. "We chose not to do that because interlacing requires a special slow phosphor monitor, which is rather expensive, or else you get a lot of flicker. Going with 640 by 200 was a very deliberate choice."

So they aimed for a maximum resolution of 640 by 200 and it was Moore who figured out a way to get it. Moore was new to Apple the company, but he was an old hand with Apple computers. "A real hacker," says Hillman. "An Apple II guy all the way." Moore was assigned to the FPI chip, but he couldn't resist trying to solve the graphics problem.

"Our goal was that the graphics use linear addressing," says Hillman. With linear addressing there is a direct correlation between the location in memory that says what the pixel should do and the location of the pixel on the screen. It's a lot easier for programmers because it's more intuitive, but it takes up a lot more memory than what now seems like the old graphics implementation Wozniak designed for the original Apple II. While Wozniak concedes that linear addressing makes more sense in today's machines, he points out that his method wasn't odd for its time—there *was* no method at the time. "You have to remember that in 1975 there wasn't one single computer that had color in it. There was one color board, for S-100 computers, but you had to assemble those computers yourself, and the color board had more chips than the entire Apple II. I had to scramble to figure out a way to add color without adding cost. I feel sorry for the guys who have to keep it compatible. The trouble is that the environment changes. What made a whole new world possible back in 1976 is clumsy to use in 1986."

Hillman was determined that they come up with linear addressing; to do that they needed to find a contiguous chunk of memory—32K—and they couldn't find that much in one location. They couldn't just move things around to make room because certain parts of memory are reserved for particular functions. Moore kept proposing solutions and Hillman kept shooting them down.

Finally, Hillman recalls, they found 32K they could use but it was in two chunks, not one. That's when Moore came up with the idea of using a custom chip, the Video Graphics Controller (VGC), to do a logical-to-physical translation and memory-address rearrangement. From a programmer's point of view, the memory appears to be in one chunk even though physically it's spread out over two areas. "The new Apple IIGS graphics modes are very clean and easy to use because of it," says Hillman, "but it's one of the most complicated chips we've got."

"The other graphics modes are still bizarre," says Hillman. Which is to say that the Apple IIGS supports all of the earlier Apple II graphics modes as well as the two new clean ones. Therefore old programs will still run on the new machine and will look the way they always have.

In some ways sound was the easiest problem to solve, but it became one of the most controversial features of the new machine. The original sound chip proposed for the Apple IIGS gave it arcade quality sound. It was nothing special, but it was more or less what the competition was doing. Then Rob Moore proposed using the Ensoniq, a sound chip used in the Mirage music synthesizer. "Ensoniq sent the chip to Woz," Moore recalls. "He wasn't that interested in sound, so he passed it on to another engineer, who thought it was too complicated. But I went crazy over it. Computer music is a hobby of mine, and this chip did all the things I'd been trying to do with software all these years. I didn't have anything to do with designing the chip, but I was the official pain in the ass who made people appreciate what it could do."

No other personal computer had anything like the Ensoniq sound chip. The danger of putting a chip like that in a computer is that it can lead to typecasting as a game machine. (At that time, the only applications besides music composition programs that used sound to advantage were games.)

Hillman finally persuaded upper management to go with the Ensoniq sound chip by using a phrase borrowed from the Macintosh marketing phrase book. "I told them it was an enabling technology—that it would enable people to do things they never dreamed of."

According to product manager Curtis Sasaki, whose job it is to dream up new products and programs, there are already a lot of innovative sound applications in the works. "Imagine a program that comes with a children's storybook," he says. "You read the story, then you start up the disk and the characters you were just reading about are talking to you. I've already seen a demo of a program like that."

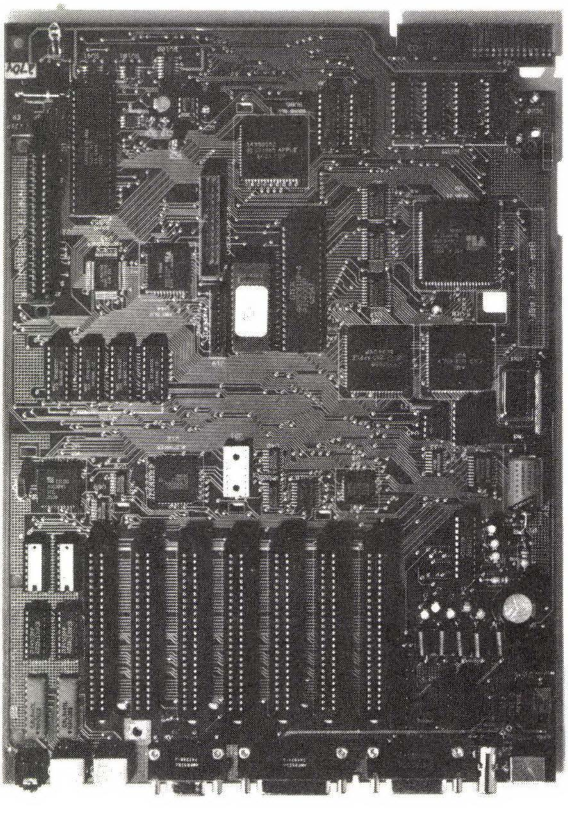
For older students, says Sasaki, there will be programs on CD-ROM's that let you hear information stored in a data base as well as read it on the screen. "Let's say you want to find out what Kennedy said in a certain speech. Instead of just seeing the speech displayed on the screen you could hear it spoken." And because of the sophistication of the sound chip, it will sound like Kennedy's own voice speaking, not like "a drunken Scandinavian trapped in an oildrum," as someone once described early attempts at speech synthesis.

Sasaki also anticipates interesting business applications for the sound chip—voice mail, for example. "Someone will be able to call you up and leave a message, and the message will be automatically digitized and stored on a disk with the time and date the person called."

Mission Accomplished

With the sound issue resolved, the hardware group had more than fulfilled their mission. They had maintained compatibility, boosted the memory beyond 4 megabytes, sped up the processor, improved

***Figure* 1 ▪ 4**
Circuit board



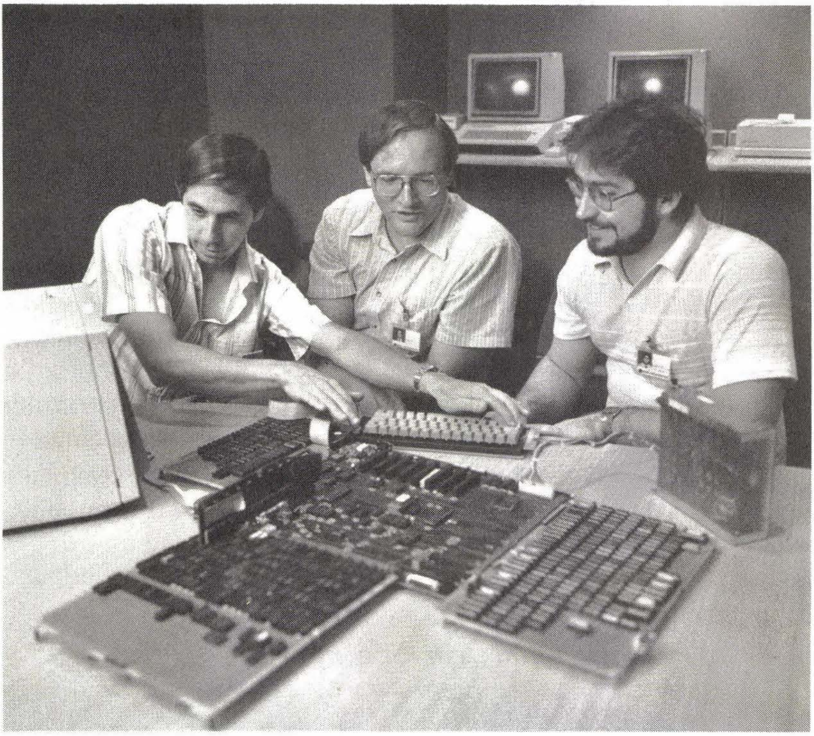


***Figure* 1 ▪ 5**
Engineers Dan Hillman, Rob Moore, and Jay Rickard with El Grando

the graphics, and leapfrogged the competition in sound. They had also added a built-in clock, made room on the circuit board for seven general-purpose expansion slots, and built in the equivalent of seven interface cards in such a way that you could tap into those cards simply by plugging the device into the appropriate port on the back of the computer. (See Figure 1.4.)

Once the hardware was defined, it was up to Harvey Lehtman's group to write the firmware, the programs in ROM that software developers would use to exploit the features of the new machine.

The first Apple IIGS systems that the system software group had to work with were what Hillman called "El Grando Boards," networks of interconnected circuit boards that sprawled across several desktops in the lab. The custom chips that would reduce the chip count from the hundreds on El Grando to a handful on the Apple IIGS were on order, but they had a very long lead time. In the meantime, it was El Grando or nothing. (See Figure 1.5.)

The code name for the project had gone from Phoenix, during its rebirth period, to Rambo, during the tense months when John Medica, Apple IIGS's product champion, was fighting for final approval from the executive staff. A product champion coordinates the efforts of the hardware, software, and other teams involved in getting a new product out. "If anyone tells you the product's canceled," he'd say, "kick 'em in the kneecaps. Or tell me and I'll kick 'em in the kneecaps." The name Rambo suited Medica, whose exuberant style helped focus attention on the project, but offended almost everyone else.

System software had their own code name for the new Apple II. They called it Gumby. The name was inspired by programmer John Worthington's unlikely impersonation of the green rubber cartoon character at Apple's annual Halloween parade. You could almost gauge the mood of the team and their progress by the contortions and costumes of the rubber Gumby and Pokey toys that hung from the ceiling and decorated their cubicle walls. (See Figure 1.6.) If Gumby was

***Figure* 1 ▪ 6**
Firmware team with their mascots

dressed for the disco, it was safe to enter the system software area. If he had a suicide note pinned to his chest, it was best to come back another time.

Firmware

Firmware is a layer of programs that sits between a computer's hardware and the application programs that use the hardware. Essentially, firmware saves application writers a lot of work. Instead of having to give detailed instructions to the hardware telling it what to do with each keystroke, how to interpret every mouse move, how to display information on the screen, where to put documents in memory, how to get documents on and off disks and out the serial port to a printer, the application can give general instructions—"make calls"—to the firmware. Because the application leaves the nitty-gritty details up to the firmware, the latter can change to cope with hardware changes—changes in memory size, changes in graphics, changes in microprocessor speed, changes in the way peripherals connect to the computer—without affecting the way the application works. As long as the same old call produces the same old result, it doesn't matter to the application, or to the user of the application, what contortions the firmware had to go through to make it happen.

An example of the way firmware can disguise changes in hardware from applications is the way a printer plugged into the printer port on the back of the Apple IIGS appears to be connected to an interface card in slot 1 on the main circuit board. On earlier models of the Apple II everything was connected to the computer by means of slots, so applications expect to find devices connected to slots. No problem. Apple IIGS firmware makes the ports look like slots containing interface cards. The information that was in the ROM on the interface card is now in the ROM on the main circuit board and applications are none the wiser.

The challenge for Lehtman's group was to give developers a way to use all the new features of the Apple IIGS while fooling old software into thinking it was still talking to the same old machine.

One of the members of his group, Rich Williams, has been performing this firmware sleight of hand since 1979. He came to Apple at the end of the first big hiring wave, when the company swelled from 100 to 250 employees. "We had disk drives connected to our Apples," says

Williams of those olden days, "but we were still shipping a lot of programs on cassette tape."

"If you've done a good job with the firmware, you can make a lot of changes to the machine without affecting compatibility," says Williams. "If programs use the firmware, and if we keep the firmware interface the same, the programs should work. Where you run into problems is where developers get in a bind and have to talk directly to the hardware. It might be that you didn't supply them with some tools they need, or you did supply the tools but they didn't like them, or you supplied the tools but you didn't document them well enough. So they work around the firmware, talk directly to the hardware, and when the hardware changes, their programs don't work anymore."

Fitting It All in ROM

Originally the system software group was allocated 64K of ROM. "We thought that was terrific," Lehtman recalls. "After all, there's only 16K of ROM in the Apple IIe. Then we started thinking about all the things we wanted to put in ROM and we had to go back and ask for 128K."

BASIC had to go in, not because it was the best programming language or even the easiest to use, but because it was in the ROM of the original Apple II and in every Apple II since. If it weren't in ROM none of the programs written in Applesoft BASIC would run on the Apple IIGS and that would rule out hundreds of educational programs. So BASIC had a confirmed reservation.

If putting BASIC in ROM was an idea borrowed from earlier generations of Apple II's, putting mouse tools in ROM was an idea borrowed from the Macintosh.

For years Apple had been urging Apple II developers to adopt a standard user interface (that is, the way a program communicates with the user), but urging is no substitute for handing them tools on a silicon platter.

"If you want people to use a consistent interface, there are two ways to go about it," says Williams. "One is to mandate what the interface is and tell developers that they have to use it. That's what we've done up until now, and that doesn't work. The other way is to provide built-in tools and say, 'You can make your own interface or you can use the stuff that's built in and have it easy.' People like to do what gets their programs done fastest, so this argument turns out to be very convincing."

So the system software group followed the Macintosh lead and put their mouse tools in ROM.

The Apple IIGS mouse tools are similar but not identical to the ones built into the Macintosh. The goal was not to be 100 percent compatible with the Macintosh tools, according to Williams, but to "steal" what was successful and change what needed to be changed. "The fundamental concepts are the same," says Williams, "but the Macintosh and Apple IIGS operating systems are radically different. You can't just spend 15 minutes fooling around with your Macintosh program and make it magically work on the Apple IIGS. And that's not such a bad thing.

"One of the things that people praise about MS-DOS (the operating system used by IBM and IBM clones) is that the same program can run on several different machines. It sounds wonderful, but the price you pay for generality is that the program is designed for the minimum machine. It doesn't take advantage of any of the special features of the machine. If you're writing something for the Apple IIGS, you want to change the application to take advantage of Apple IIGS features like color and this fancy sound chip we've got. It isn't as if we sat back and said, 'Let's make it difficult for them to port things over because it's good for them.' We just designed the tools that were best for our machine."

The Control Panel

The Control Panel is another idea borrowed from Macintosh. It's a program in ROM that lets you set the built-in clock and change the way information is displayed on the screen, the touch of the keyboard, the responsiveness of the mouse, the way information is sent to printers and modems, and more. It lets you customize the look and feel of the computer system to suit yourself. (See Figure 1.7.)

While the program that lets you change these settings is in ROM, your preferences are recorded in a special battery-powered RAM that doesn't get erased the way normal RAM does when the power goes off. You'll learn more about the Control Panel in Chapter 6.

Operating System

Operating systems and utilities are also part of the system software domain. A disk operating system is a set of programs on every application disk that manages how information is copied from disks into

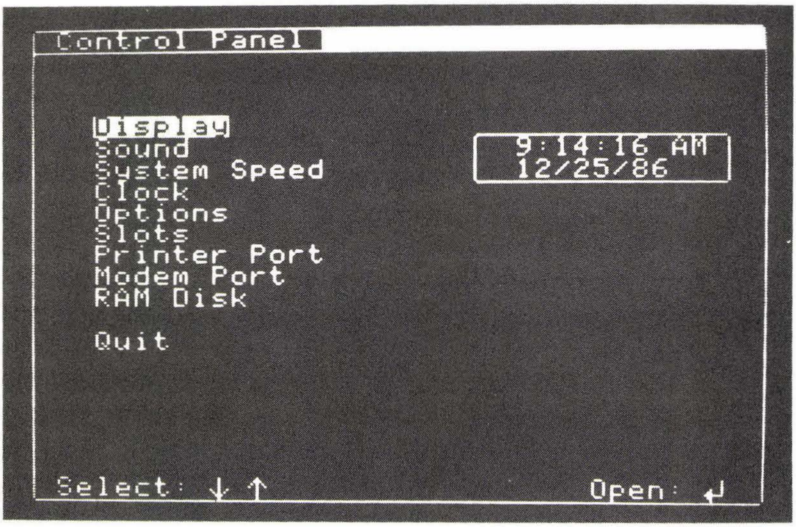


***Figure* 1 ▪ 7**
Control panel

memory and how it is copied from memory to disks. Lehtman wanted his group to update ProDOS, the current Apple II disk operating system, so it could take advantage of the Apple IIGS's additional memory and deal with the large storage devices that loomed on the horizon. He also wanted to create mouse-based utilities modeled after the utilities that are part of the Macintosh Finder.

ProDOS was developed at about the same time the Apple IIe was being developed—around 1983. It was designed with room to grow, but memory size and disk storage capacity have grown at a rate that was hard to imagine in 1983, so ProDOS is being updated.

"At the time ProDOS was developed," says Williams, "5-megabyte hard disks were a big deal and ProDOS could handle 32 megabytes, so it seemed like there was a lot of room to grow. Well, nobody makes 5-megabyte hard disks anymore. Very soon the standard will be 40 megabytes and then 80 megabytes." The new ProDOS doesn't put a limit on the size of the storage device.

It's hard to imagine one person filling 80 megabytes and it's unlikely that any one person will. The expectation is that several computers will use one large-capacity hard disk by forming a network. In its new version, ProDOS will be able to share a hard disk with another operating system (as it would have to do if Macintosh users and Apple IIGS users wanted to share the same hard disk).

Another reason for revising ProDOS is to make it possible for more documents to be open in memory at one time. The current version allows for eight documents to be open at a time. That was more than enough for 128K machines, but an Apple IIGS can have more than 4 megabytes of memory. The new version of ProDOS will make it possible for any number of documents to be open at one time. It will finally be possible for people to keep their computer "desktops" as cluttered as their real desktops.

Utilities

Utilities are programs that handle disk maintenance and housekeeping chores like copying disks and documents, deleting documents, renaming disks and documents, and so on. The System Utilities Disk available for the Apple IIe and IIc would have worked for the Apple IIGS, but Lehtman wanted mouse-based utilities that would be simpler to use and would promote the use of the mouse interface for new applications.

Mouse-based utilities are easier to use than menu-driven utilities because you can show the program what you want it to do by using sign language rather than typing commands. (See Figure 1.8.) For example, to indicate that you want to delete a document, you drag a picture of the document to a picture of a trash can. To indicate that you want to move a document to another disk, you drag a picture of the document to a picture of that other disk. You'll learn more about using Apple IIGS utilities in Chapter 3.

About the time that the El Grando boards reached the system software group, Apple decided to merge the Apple II and the Macintosh divisions. Steve Jobs, the head of the Macintosh division, opposed the idea and wound up leaving Apple in the wake of the sweeping reorganization.

It's hard to say what effect the reorganization had on the Apple IIGS project, but the Macintosh influence is unmistakable, from its detached keyboard to its mouse-based utility program modeled after the Macintosh Finder. Lehtman believes that the Apple II was moving in that direction anyway. He points out that there were mouse tools available to Apple II developers back in 1984. He does concede that having the Macintosh people in the same building, on the same floor,

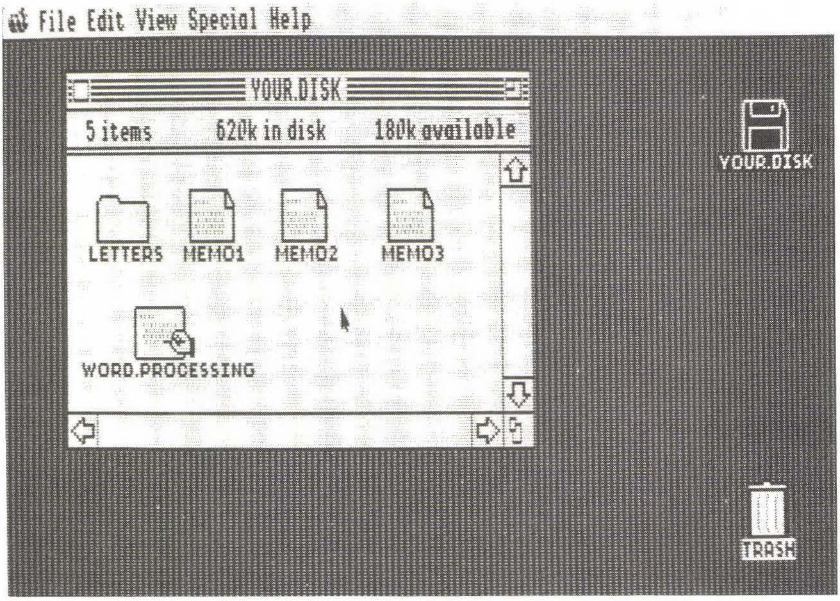

File Edit View Special Help
YOUR.DISK
5 items 620k in disk 180k available
LETTERS
MEMO1
MEMO2
MEMO3
WORD.PROCESSING
YOUR.DISK
TRASH

***Figure* 1 ▪ 8**
Finder screen

was "very, very helpful" when they were adapting the mouse tools for the Apple IIGS. Bill Atkinson, who wrote MacPaint and many of the mouse tools for the Macintosh, helped Steve Glass create a color version of his program QuickDraw, which is the basis for the tools that draw the pull-down menus, dialog boxes, and other graphic elements that are the foundation of mouse-based applications.

Lehtman also admits that it would have been hard to put a mouse in the box if it hadn't been for the reorganization. "Steve [Jobs] didn't want us to use the mouse interface on the Apple II," says Lehtman. "He wanted the identities of the machines very separate. We felt there was enough difference in the technology of the two machines. It was a waste not to use such a powerful interface."

Summary of Apple IIGS Features

Experienced users will tell you that the way to choose a computer is not by its hardware features, but by the software that runs on it. They'll advise you to choose the word processing program, spreadsheet program, art program, and other software you want, then find the computer—the hardware—that runs that software.

That's good advice, but you'll find that there's a direct correlation between the hardware features a computer has and the quality of software available for it. Great hardware doesn't guarantee great software; it just makes great software possible. If you were only buying software once, it would make sense to choose your software first, choose the computer that runs that software, and ignore the hardware entirely. But most people add to their software library over time. If you plan to do that, you want hardware that not only runs the best software available today, but that will be able to run innovative software developed in the future.

The Apple IIGS has a 16-bit processor, 256K RAM, 128K ROM, screen resolution up to 640 by 200 pixels, 15-voice sound, a built-in clock, support for all the standard peripheral devices (a printer, disk drives, a modem, a mouse, game controls, monochrome and color monitors), seven slots for adding interface cards, and a memory expansion slot for adding up to 8 megabytes of additional RAM or 1 megabyte of additional ROM.

Here's what all that means. (If you don't care what it all means, skip to Chapter 2 and let the hardware features speak for themselves when you start using software designed to take advantage of them.)

Processor

The Apple IIGS uses a 16-bit processor called the 816 (short for 65C816). The 816 was designed by Bill Mensch, one of the designers of the 6502, the 8-bit processor used in the original Apple II. The beauty of the 816 is that it can behave like an 8-bit processor when it's running programs designed for the 6502 and like a 16-bit processor when it's running programs that use its 16-bit features.

Memory

The amount of memory, or RAM, a computer has determines how sophisticated, powerful, attractive, entertaining, even how helpful applications for that computer can be. Integrated applications, for example, use more memory than stand-alone applications. Mouse-based applications, because of their heavy use of graphics, use more memory than keyboard-based applications. Animation, speech, music, help messages that are built into a program—all sorts of things that make programs innovative and interesting—eat up memory space. The amount of memory a computer has also determines how big a document can be and, indirectly, how many documents can be in memory at one time.

The original Apple II was the first home computer with enough memory to run spreadsheet programs—and at a time when spreadsheet programs required only 48K. Most spreadsheet programs today are part of integrated business applications that require at least 128K. These applications consist of programs like a word processor, data base, and spreadsheet designed to work together, so that, say, documents created with one application can be pasted into a document created with another application.

The Apple IIGS has 256K built in and you can add additional memory by plugging an Apple IIGS Memory Expansion Card into a special slot. As memory chips get cheaper, a 4-megabyte memory expansion card will be available. For now you can add up to 1 megabyte of RAM in increments of 256K.

Memory

K is short for *kilobyte*—the unit of measurement used to describe a computer's memory size and disk storage capacity. A kilobyte is 1024 bytes, and a byte is the amount of memory it takes to store one character—for example, the "H" in Hello Mom.

If it takes one byte to store one character, that means it takes about 2000 bytes, or 2K, to store one typed page of characters. And that means that if there weren't other things like programs taking up space in memory, you could fit a 32-page short story in 64K, a 128-page thesis in 256K, a 512-page book in one megabyte (which is a million bytes or 1000K) and an 8192-page encyclopedia in 16 megabytes.

1 character = 1 byte
1024 bytes = 1K
1 million* bytes = 1000K = 1 megabyte

*1,048,576 bytes to be exact—call it a baker's million

ROM

The Apple IIGS has 128K of ROM, compared with 16K in the Apple IIe. The Apple IIGS uses that extra ROM for storing the programs that tell the computer how to communicate with devices plugged into its ports—information that otherwise would be in ROM chips on interface cards if the devices were connected to slots as they are on the Apple IIe. The extra ROM is also used for programming tools that allow software developers to create mouse-based applications conforming to Apple's standard mouse interface.

One thing that's been in the ROM of every model of the Apple II is the BASIC programming language. Because it's in ROM, you can write programs in the Applesoft BASIC programming language without first loading the language from a disk.

Screen Resolution

Resolution refers to the crispness or clarity of what you see on the screen. The quality of resolution is determined by the number of dots (called *pixels*) used to generate the image on the screen. A picture with very high resolution looks like a photograph. A picture with very low resolution looks like a mosaic. The fewer dots used to form the image on the display, the grainier the image.

If you stick with programs designed especially for the Apple IIGS, you'll enjoy uniformly high resolution. If you dabble in older programs, you'll see a wide range of resolutions. In low resolution mode, one of two modes available for the original Apple II, programmers had only 48 by 40 blocks to work with. You had to use a lot of imagination with programs that used low-res graphics because a drawing of a spaceship might just as easily be a drawing of an apartment house. The Apple IIGS's resolution, by comparison, is 640 by 200 pixels using 4 colors and 320 by 200 pixels using 16 colors. Figure 1.9 shows a picture that uses the super high resolution graphics mode.

Color

Programs designed especially for the Apple IIGS can use 16 colors at a time chosen from 4096 possible colors. A programmer can specify a different set of 16 colors for different parts of the screen, however, so you can actually have one screen that uses 256 different colors at one time. What that means in an art program is that you can paint

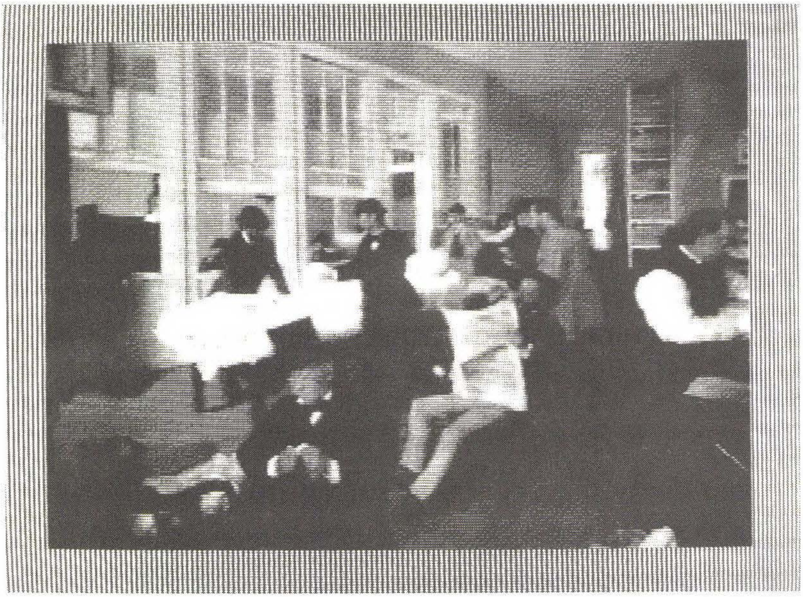


***Figure* 1 ▪ 9**
Super high resolution graphics

part of your picture with one palette of 16 colors, then change the palette to draw another part of the picture. Your sky can have 16 shades of blue and your wheat field can be 16 shades of gold and brown. By redefining the palette on the fly, programs can give the feeling of wheat fields waving, dust clouds rolling, and water cascading down a waterfall.

Because of the range of colors and the ability to redefine the palette, professional artists can use the Apple IIGS as a serious drawing tool, and applications can present very sophisticated graphics.

Sound

The Apple IIGS has the same sound chip that's in the Ensoniq Mirage, a music synthesizer used by professional musicians. Thus professional and amateur musicians can score music with a music composition program and hear it played back by a full orchestra or by any individual instruments the program lets you choose.

Software developers can select sounds from a library that includes a solo violin, a violin section, a variety of drums, horns, woodwinds, and much more. If a developer wants sounds that aren't in the library, he can sample his own sound. "Sampling sound," explains Geoff Brown, author of *Music Construction Set* for the Macintosh and the designer of some of the Apple IIGS sound tools, "means taking a microphone, playing the sound you want, and capturing it at a particular frequency. Let's say the sound was a bass playing one note. Once I have a real bass sound, I can play it at different speeds and get different pitches." The sound doesn't have to be an instrument, Brown points out. "You can sample yourself going 'whoop.'"

Software developers can take advantage of the Apple IIGS's sophisticated sound to add speech to their programs or for dramatic musical effects.

Connecting Peripherals

Peripherals are devices that you can attach to your computer to make it more useful, powerful, or playful. Some peripherals are absolutely essential. You need a monitor to see what you're doing. You need a disk drive to load programs from disks and to save documents on disks. You need a printer to get your documents down on paper. You need a mouse in order to use mouse-based applications. Things like modems, a color monitor, and a joystick aren't absolutely essential, but a lot of people wind up getting them. (See Figure 1.10.)

The Apple IIGS makes it easy to connect these essential and popular peripherals to the computer by building in "support." The instructions that tell the computer how to communicate with each of these standard devices are in ROM, so all you have to do is plug the device into the appropriate port on the back panel. There are even pictures (icons) above each port to show you what goes where. (See Figure 1.11.)

Furthermore, you aren't limited to the devices for which there are ports. If you want to connect something unusual to your Apple IIGS, you can do so by using the slots inside the case. (See Figure 1.12.) Slots are just connectors for interface cards. (See Figure 1.13.) Interface cards look like miniature versions of the main circuit board, that green sheet of plastic inside the computer case that holds the processor, the memory chips, and the network of wires that connect all the chips to each other. The chips on an interface card contain the special instructions that tell the computer how to communicate with a particular peripheral device. You plug the interface card into the

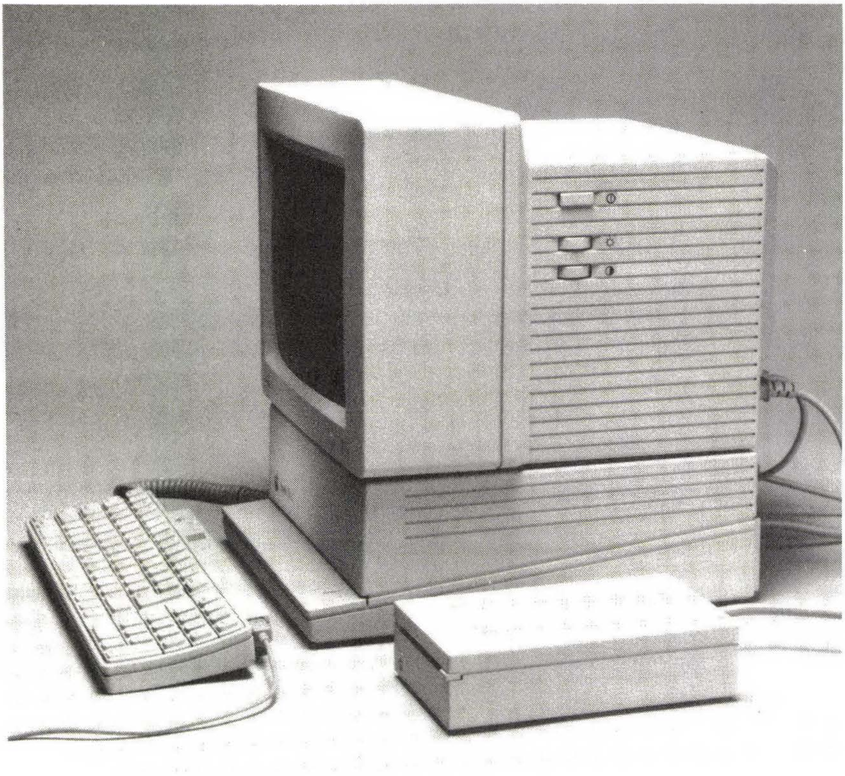


Figure 1 ▪ 10
Peripherals

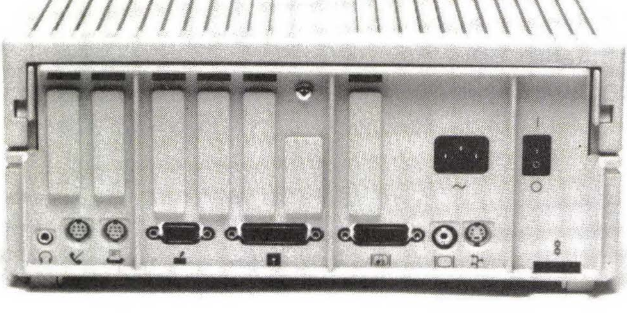


Figure 1 ▪ 11
Apple IIGS back panel

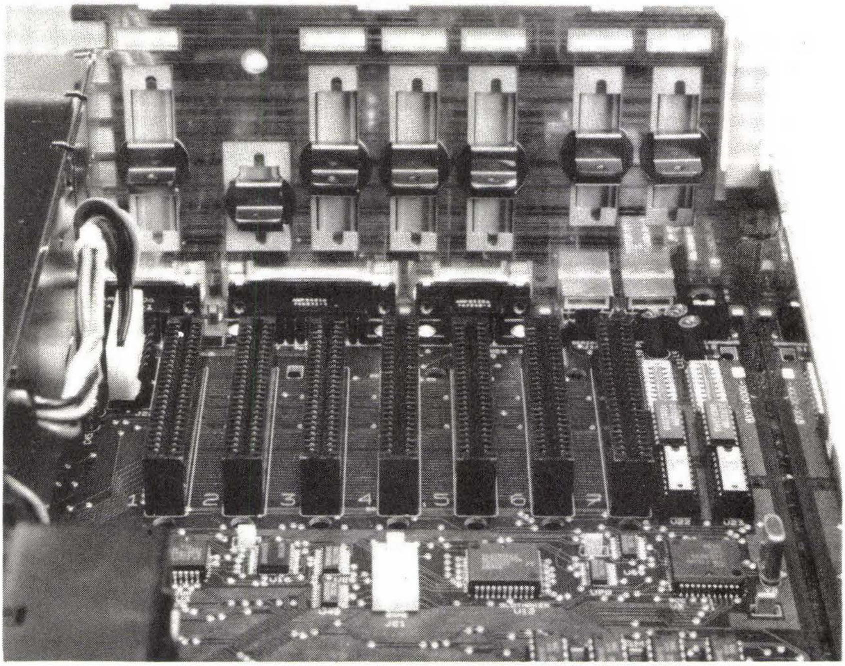


***Figure* 1 ▪ 12**
Slots

slot, plug the peripheral device into the card, and presto, the computer and the peripheral device know how to relate to each other.

Because the support (the instructions that tell the computer how to talk to the device) is on the cards instead of on the main circuit board, there's almost no limit to the kinds of things you can connect to the Apple IIGS by using the expansion slots.

Slots were a big key to the longevity of the original Apple II. They enabled early Apple II owners to take advantage of disk drives when they became available even though their machines were designed for cassette recorders. Similarly, because of the slots on the Apple IIGS, you can take advantage of hard disk drives and CD ROM players when they become more affordable, even though the built-in support is for 5.25-inch and 3.5-inch drives.

"You have to give people the ability to grow," says Steve Wozniak. "We know what people want today, but we can't define what they'll want tomorrow. The ability to grow is a strong selling point. People want to feel like they can add to their system, not that they're stuck with one system forever."

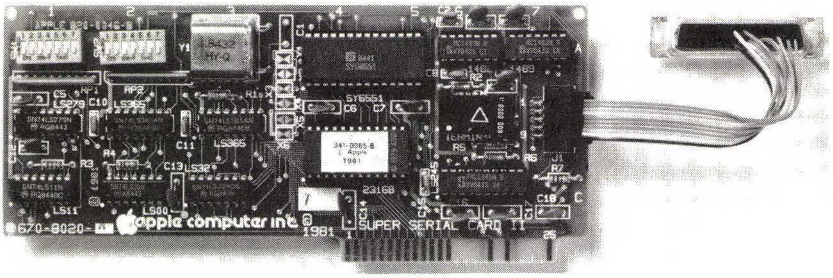


***Figure* 1 ▪ 13**
Interface card

The Mouse

The mouse isn't a new peripheral for the Apple II. It's been available as an option since 1984. There were even mouse tools available for developers to use in creating mouse-based applications. However, it was always an option; with the Apple IIGS the mouse is a standard feature that comes with the machine. See (Figure 1.14.)

When the mouse was just an option, there wasn't much incentive for software writers to design mouse-based applications. There were a few painting programs for the Apple II that used the mouse, but it wasn't the standard way of controlling an application program the way it is on the Macintosh. Now that the mouse is standard equipment, you're bound to see a lot more mouse-based applications.

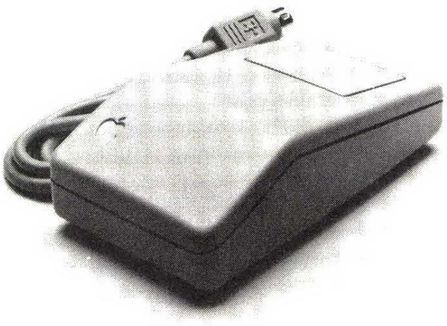


***Figure* 1 ▪ 14**
Mouse

CHAPTER TWO

The Mouse Interface

People and computers speak fundamentally different languages. We need something to mediate between us that we both understand. Therefore a computer has what's called a *user interface,* a means by which information is passed back and forth between the computer and the person who's using it.

In the best of all possible worlds, you could turn to your computer as you would to a secretary and say, "Get me last month's sales figures," and the sales figures would appear on the screen. Computers can't do that yet, but in the last few years there has been significant progress away from the crabby, confusing, unpredictable user interface usually associated with computers toward one that gives you the comforting feeling that you and the computer work the same way.

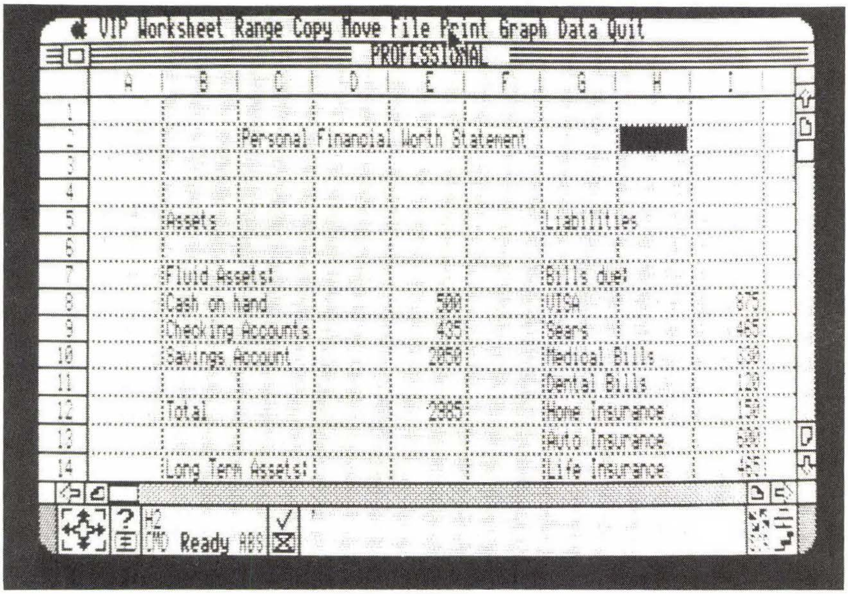
The device that is almost single-handedly paving the way toward more straightforward communication between you and the computer is the mouse, a little box that you slide around on your desk in order to move a pointer on the screen. The mouse lets you point instead of typing your command, and that one change makes a dramatic difference.

Using the Mouse Interface

On the screen, a typical mouse-based program looks like Figure 2.1.

The Pointer

When you move the mouse on your desktop, the pointer moves on the screen in a corresponding way and tells the Apple IIGS what you want to do. Moving it so it's aimed at something in particular is called *pointing.* Pressing the button on the mouse is called *clicking.* In general, pointing directs the Apple IIGS's attention to a specific location or object (icon), and clicking causes something to happen at that point or to that object. There's also something called *dragging*, which is holding the mouse button down while you move the pointer. You drag to move an object from one place on the screen to another, and also to open menus and change the size of windows (more about windows later in this chapter). Except for entering text and numbers, you can do just about everything in mouse-based programs by pointing, clicking, and dragging.

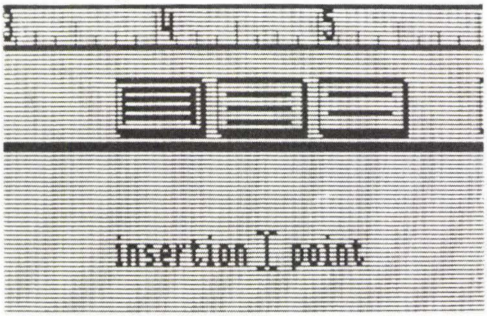


	A	B	C	D	E	F	G	H	I
1									
2			Personal Financial Worth Statement						
3									
4									
5		Assets					Liabilities		
6									
7		Fluid Assets:					Bills due:		
8		Cash on hand			500		VISA		375
9		Checking Accounts			435		Sears		465
10		Savings Account			2050		Medical Bills		330
11							Dental Bills		120
12		Total			2985		Home Insurance		150
13							Auto Insurance		500
14		Long Term Assets:					Life Insurance		465

***Figure* 2 ▪ 1**
Typical mouse-based program

Sometimes the pointer takes other shapes besides the arrow shape. When you're pointing at a spot on a page of text, for instance, the pointer looks like the one in Figure 2.2, sometimes called an *I-beam.* When the computer is busy doing something (saving a document to the disk, for example) and you can't continue working until it's finished, the pointer looks like a little watch (Figure 2.3). Within a graphics program you might find that the pointer sometimes looks like a paintbucket (Figure 2.4), which means you can "pour paint" with it. The pointer's shape depends on what its function is at the time.

***Figure* 2 ▪ 2**
I-beam pointer



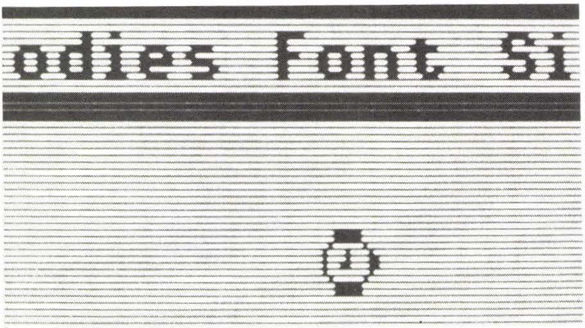


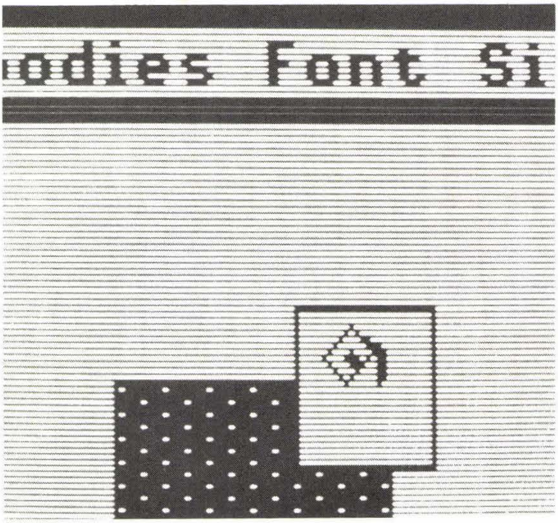
Figure 2 • 3
Watch pointer

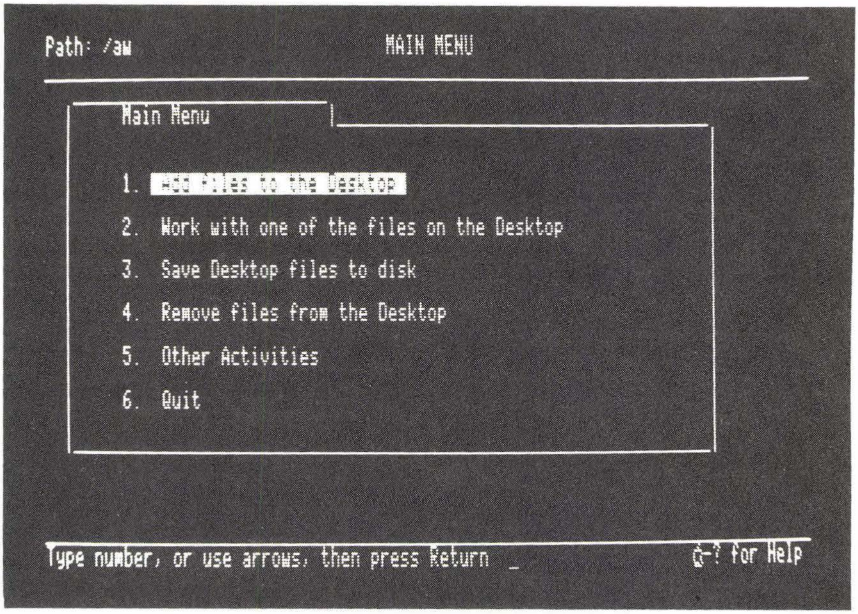
Menus

Programs had menus before the mouse interface came along. A menu is simply a list of choices on the computer screen. Instead of typing out the entire command, you type the letter of your choice or use arrow keys to move a marker to it (Figure 2.5).

Mouse menus work on the same principle, but take up less room on the screen. You don't have to leave what you're doing and go to a special menu screen to choose a command; the commands are always available on the screen where you're working. All you see are the menu titles, which are the words across the top of the screen. You can't see any of the choices until you "open" a menu by pointing at

Figure 2 • 4
Paintbucket pointer





```
Path: /aw                          MAIN MENU

    Main Menu

    1. Add files to the Desktop

    2. Work with one of the files on the Desktop

    3. Save Desktop files to disk

    4. Remove files from the Desktop

    5. Other Activities

    6. Quit

Type number, or use arrows, then press Return _        Ǘ-? for Help
```

***Figure* 2 ▪ 5**
Old-style menu

its title and holding down the mouse button. This type of menu is called a *pull-down* menu because it looks rather like a window shade being pulled down (Figure 2.6).

To choose a command from a menu, you hold down the mouse button and move the pointer down the list. Each command is highlighted when you point to it—that is, the dark type on a light background becomes light type on a dark background (Figure 2.7). When the one you want is highlighted, you let go of the mouse button and the Apple IIGS carries out the command.

What menu titles you see depends on what kind of application program you're using. One menu, however, is common to all applications: the File menu. This is where you'll find the commands to save your document, print it, close it, open another document, or start a new one from scratch.

Other menus you're likely to encounter include these:

- Edit: Lists text editing options such as Cut, Paste, and Copy.
- Font: Lists the names of the typefaces you can choose from.
- Style: Gives you choices of text style such as italic, bold, and underline. Sometimes also includes choices for type size, usually ranging from 9 to 24 point.

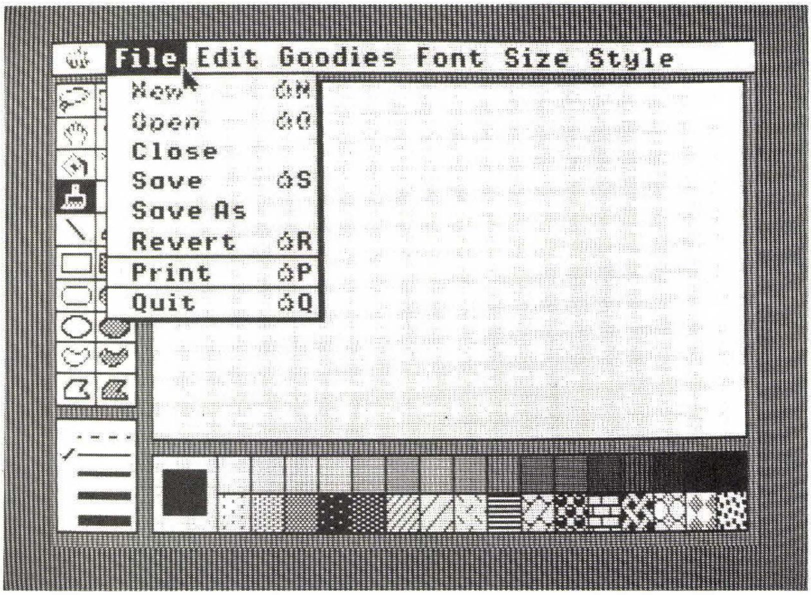


***Figure* 2 ▪ 6**
Pull-down menu

Desk Accessories

At the far left end of the menu bar is a small picture of an apnle, which is actually a menu title just as the words to its right are. It's called the *apple menu.* No matter which (mouse-based) application you're using, the *apple menu* will appear in that spot. What choices are on it depends not on the application but, for the most part, on you.

The items on the apple menu are called *Desk Accessories* because they are like the collection of handy tools you keep on your desk where you can reach them whenever you want. You might have a notepad, a calculator, a clock, and an appointment calendar on your desk; on your Apple IIGS, you can also have a notepad, a calculator, a clock, or an appointment calendar. As with real-life desk accessories, you can get at these any time you need them, no matter what program you're using.

There are two desk accessories built into the Apple IIGS. One is a calculator and one is a puzzle. If you want any others—and you probably will—you have to put them there yourself.

A calculator is among the most useful. If you need to do a quick

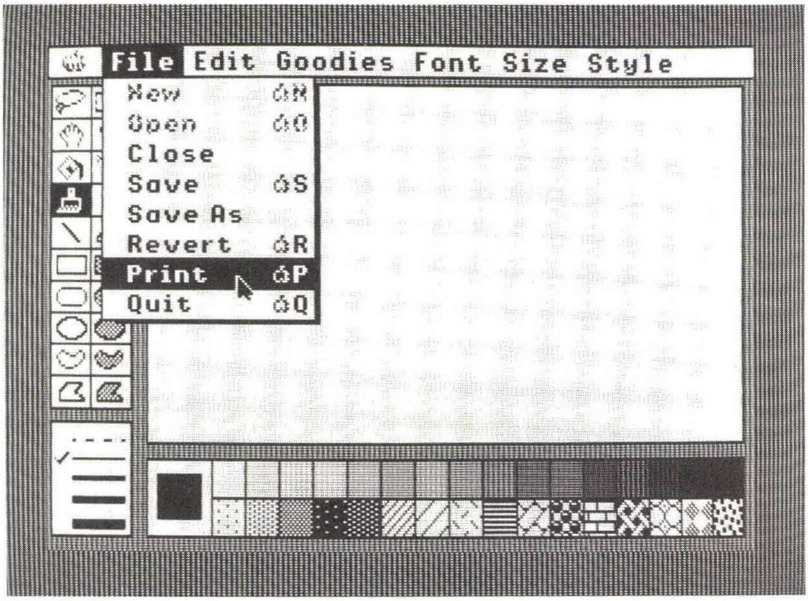


***Figure* 2 ▪ 7**
Menu with one option highlighted

computation while you're in the middle of writing a sales report, you choose Calculator from the apple menu, and a realistic-looking calculator appears on the screen, overlaid on your letter but having no effect on it. Using the mouse, you click on the buttons to enter your numbers and get your answer. When you put the calculator away, you can go right back to work.

A notepad is also convenient. Say you're working on Section 1 of a report, and you suddenly have a magnificent idea that you can use in Section 3. You haven't gotten to Section 3 yet, but you want to write down your idea so you won't forget it. You call up the notepad from the apple menu—it looks like a little pad of paper—and you type your idea there. When you get to Section 3, you pull out your notepad again and your idea is waiting for you.

The *Scrapbook* is very much like the notepad. It's a place to store either text or graphics that you're going to need later on or that you use frequently. The main difference between the scrapbook and the notepad is that you can copy things back and forth between the scrapbook and your document. For example, suppose you're using a small picture of a car to accompany the headings in a set of overheads. You create the picture using a graphics program and store it in the scrapbook. Then you start typing your overheads, and whenever you want to insert a car, you open the scrapbook, copy the car picture, close the scrapbook, and paste the picture into your document.

An alarm clock is another popular desk accessory. It serves the same function as one of those watches that beeps at you when it's time to go to your next meeting.

Windows

Just about everything that happens in a mouse-based application program happens inside a *window,* a rectangle that frames your document. It's called a window because, in a sense, you look through it at what you're working on. You can imagine how it works by picturing a pane of glass with a long piece of paper scrolling along behind it.

There's only so much of the document that will fit on the screen at once. This is the amount you see through the window. You can cause the "paper" to scroll up and down by using the *scroll bar,* the gray vertical bar you see at the right side of the window. Dragging the *scroll box* up or down the bar moves the document up or down on the screen. If the box is at the top of the scroll bar, you're looking at the beginning of your document, and if it's at the bottom, you're looking at the end. You can go through your document page by page, line by line, or a few paragraphs at a time. (Sometimes there's also a scroll bar at the bottom of a document. This lets you move sideways in documents that are *wider* than the screen.)

The top part of the window, a dark or colored band interrupted by the title of the document, is called the *title bar.* It functions as a kind of handle. When you grab this handle by pointing at it and holding down the mouse button, you can drag the whole window around on the computer screen. This is useful for the tasks you do on the Desktop (see Chapter 3) and also for programs that allow you to have more than one window on the screen at once. It's handy to be able to arrange them so you can see both.

Whichever window you're working in at the moment is called the *active* window. In that window you might have the speech you're writing about whales. In the window behind it you can have the outline you've developed for your speech. When you want to refer back to the outline, you make the outline window active simply by clicking on it. The outline window then comes to the front so you can see it, and the dark band appears across the top. (There is no band across the nonactive window.) Now you can study the outline, or make changes to it, until you're ready to write some more, at which point you just click on the other window.

Windows can also change size and shape. By pointing to the size box (in the lower right corner of the window), holding the mouse button down, and moving the mouse, you can adjust your windows to your purposes. You might resize your outline and speech windows, for example, so that they both occupied half the screen and you could refer to them both at the same time.

There's also a quick way to expand your window to full screen size, as well as to close it when you've finished working on it. In the upper right corner is the *zoom box*; click here and your window will immediately expand to fill the entire screen. In the upper left corner is the *close box;* clicking here is the equivalent of choosing Close from the File menu. The window showing your document disappears from the screen, but you're still within the application. You can start a new document by choosing New from the File menu, or work on one you've already started by choosing Open.

Every mouse-based program uses windows, pull-down menus, and icons. What you've just learned will serve you well in any of the new Apple IIGS programs you buy. There are some programs written for the IIe and IIc that *look* like mouse programs, but these work like mouse programs only up to a point. Many simply give you the ability to pull down menus. A few—*Mouse Desk* from International Solutions; *Person-to-Person* from TruTec; and *Catalyst 3.0* from Quark—do provide a full-fledged mouse interface, with windows, menus, dialog boxes, and icons. The others, however, are a blend of the mouse and the premouse interfaces. There's nothing wrong with these programs; but you can expect them to surprise you every now and then with an old-fashioned interaction.

Origins of the Mouse

The idea for a mouse-based interface originated not with Apple but with an engineer named Doug Engelbart at Stanford Research Institute in Menlo Park, California. In those days, the typical user interface was intelligible only to the program's designer and eight or nine of his most intimate friends—the other guys at the lab. To them, figuring out how a program worked was as engrossing as an adventure game. And, as in the best adventure games, breaking the interface code could take days. You had to need a program badly to bother to learn to use it.

Engelbart came up with a revolutionary idea: there might be a way to make computer programs learnable by ordinary people. The key was to use a pointing device instead of a large and intricate set of typed commands, and to put objects on the screen that the user could point at. Pointing at something was intuitive ("intuitive" was to become a byword later, at Apple). Everyone could point; not everyone could (or wanted to) learn a lot of commands. Doug Engelbart developed the mouse, and later on a group of engineers at Xerox PARC (Palo Alto Research Center) combined the mouse with an interface that made use of icons, and they got the whole thing running on a computer called the Star.

But Xerox was very slow about putting the Star into production, and many members of the team that had worked on it became impatient and left for other companies. One of the companies was Apple, where some of the Xerox engineers began working on a development team that ultimately produced the design of the Lisa and Macintosh interface.

Apple's Lisa computer was the first personal computer to use the mouse interface, but at $10,000 per machine it wasn't affordable enough to get mass exposure. Not until the introduction of the Macintosh did the interface start getting widespread attention and, later, the flattery of imitation in the form of products like Microsoft's *Windows* for the IBM and the GEM interface for the Atari. Now mice seem to be breeding like rabbits, and for good reason. The mouse has made some basic changes in computing that bring us a step closer to that ideal world in which we need only speak casually to our computers to get results.

You may have heard of something called a "transparent" user interface. "Transparent" has become a catchword in discussions of computer interface design; supposedly, a transparent user interface is one so unobtrusive that you barely know it's there. When you start thinking about it, however, you begin to wonder exactly what a transparent interface could be, unless it's the completely invisible interface of a computer that understands spoken English. Short of that, what you really want is not a transparent interface (which, taken literally, would let you look right into the dark, tangled innards of the machine) but an interface that makes things easy for you by letting you work in the most natural way possible. This, as it turns out, is not a transparent interface at all but a highly visible one, made up of recognizable, movable objects—an environment that imitates the concrete world you move around in every day.

Toward a Visible Interface

The Apple team took what Xerox had started and carried it forward. They added pull-down menus (an Apple invention) and other refinements, but their main contribution was philosophical. They took a close look at how the interface worked and compared it with how people worked, and they came up with some interesting observations.

"It wasn't icons that led to ease of use, it was the concept of direct manipulation," says Chris Espinosa, who was closely involved with the design of the interface. "Programs at that time isolated you from what you were doing. You stored your data in a data file and then you issued a command to invoke a file that had some code in it that performed a process on the data and typed the information somewhere else. . . . It was all very indirect. You didn't have the feeling of being there. With the mouse interface, you can be there all the time."

Espinosa, who is Apple employee #8, joined Apple in 1978, before it was a company. He was 16 years old at the time, a computer enthusiast who got to know Steve Wozniak because he hitched a ride with him to meetings of the Homebrew Computer Club. Espinosa got involved with interface design when he was named head of documentation for the newly formed Macintosh development team. "It turns out that the best way to find conceptual problems in a piece of software is to try to explain how it works to somebody else," says Espinosa. "So we [the documentation group] had a lot of influence on interface design. The guiding principle was that if it was hard to explain, it wasn't done right."

Considering that Espinosa was weaned on computers and is an expert at cracking obscure interface codes, he is surprisingly sensitive to the needs of new users. But you don't hear him advocating a "friendly" interface. What he wants is an interface that stays out of the way.

"Before, you had to use the arrow keys to move around the screen, which is still fairly indirect. With the mouse," he says, "you can point to where you want to go and go there. You can mark a block of text and say, 'Put this in bold,' and it's done, immediately. You don't have to set a marker at the beginning that says 'begin bold' and put a marker at the end that says 'end bold.' You just say, 'Put this in bold,' and it happens."

The mouse interface lets you work the way you do when you're not using a computer. Want to put a document in a folder? Pick it up (by pointing at it with the mouse and holding down the mouse button), and move it to the folder icon. Want to widen the margins of your

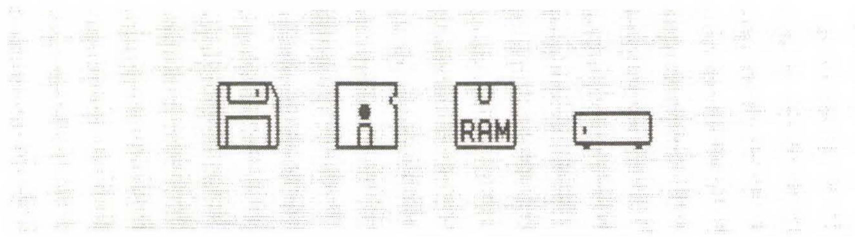


Figure 2 ▪ 8
Disk icon

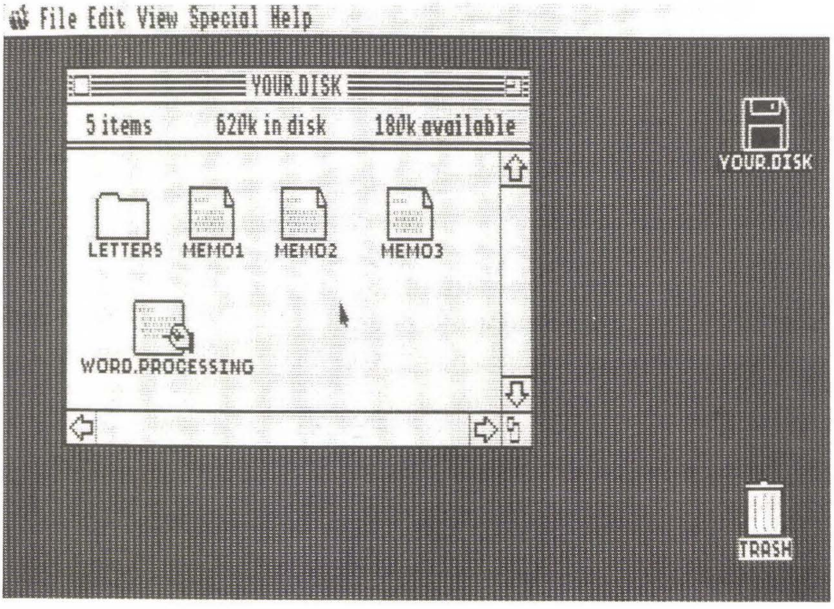
document? Push the margin marker farther out, just as you would on a typewriter. Want to draw a zigzag line? Move your hand in a zigzag, just as you would with a pencil. No computer-language command gets between you and the work.

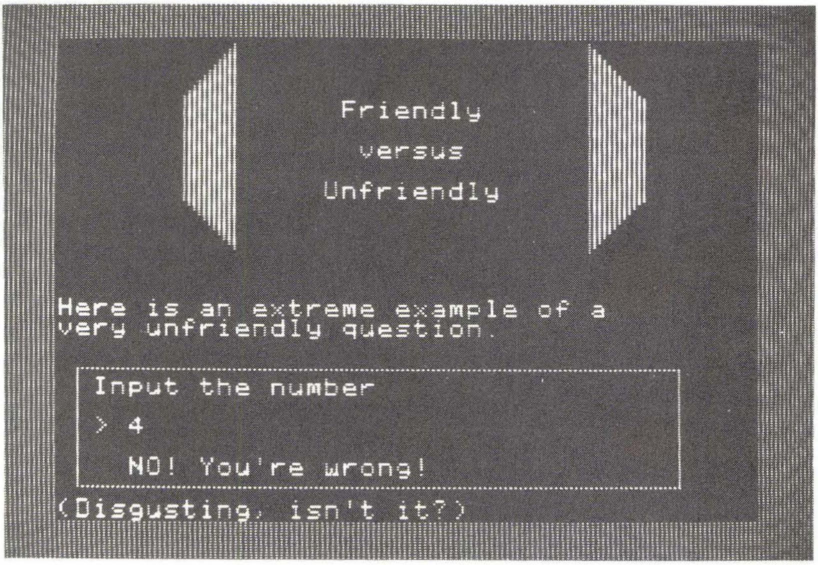
Icons contribute to this straightforward approach. The disk in your disk drive is represented by a picture of a disk (Figure 2.8), not by a string of characters.

You delete a document from your disk by moving a picture of a document to a picture of a trash can (Figure 2.9), not by typing DELETE .D2/REPORT.4/CHAP.1.

You still need words—words are often easier to understand than icons—but words don't dominate the screen in mouse-based applications. Even the words on menus stay out of sight until you need

Figure 2 ▪ 9
Trash icon and document icons





```
Friendly
versus
Unfriendly

Here is an extreme example of a
very unfriendly question.

Input the number
> 4
  NO! You're wrong!
(Disgusting, isn't it?)
```

Figure 2 ▪ 10
Screen from Apple Presents Apple

them. When you *do* want them, however, they're only a mouse move away. That means you can do what you want to do and go where you want to go in the program with a lot more freedom than you could before. The mouse interface puts the human being in charge.

Toward a More Human Interface

Apple's concern with a human interface goes back long before the introduction of the Macintosh. Bruce Tognazzini, working inside Apple, and Scot Kamins, working outside Apple, were two early advocates of human interface design. Tognazzini was the founder of Apple's Human Interface group and author of the original *Human Interface Guidelines*, first distributed in the fall of 1978, a document that tells software developers how to design programs that support users instead of inadvertently trapping or tripping them up. He and programmer J. D. Eisenberg authored the first microcomputer online interactive training program, Apple Presents Apple, in which they take a poke at "unfriendly" programs (Figure 2.10).

"The thing that triggered the writing of *Apple Presents Apple,*" Tognazzini explains, "was a paper lesson that Sue Espinosa (now head of Apple User Education) developed. It told the reader, 'Press the Return key,' and then below that it listed the 27 things that would happen if you actually pressed some other key instead. She wanted us to review this and see if it was complete, and we reviewed it and said, 'This is silly.' We thought that since people were learning a computer, they should use the computer to learn." (Espinosa privately thought so too. Tognazzini's reaction to her lesson was the one she'd hoped for.)

Apple Presents Apple was designed for new users, but the message about friendly and unfriendly programs was clearly directed at Apple II software developers. Few of them were making strenuous efforts to humanize their interfaces in those days, and even among those who were, there was no consistency at all. Both Tognazzini and Scot Kamins were working to change that.

Kamins is the author of *The Apple Backpack,* one of the first books on human interface design for BASIC programmers. Kamins recalls what the state of user interface standards was when he and Tognazzini got their first Apple computers in January 1978. "In those days, you loaded a program into the computer and all you saw was a blinking question mark in the upper left corner of the screen. It was terrible. That's how we got interested in human interface design. That's why we became fanatics about it."

Getting software developers to incorporate features like error checking and confirmation questions ("Closing this file will lose some data, are you sure you want to do that?") was an uphill battle. "Programmers are an ornery, independent lot," Kamins observes. "They want to do it their own way and they don't mind being the pioneers—the guys with the arrows in their backs. The problem is that they were writing their programs for other programmers, who could deal with an interface that consisted of a blinking question mark."

So Tognazzini continued to expand the human interface guidelines and to promote their use by lecturing to and working with Apple developers all over the world. The guidelines stress consistency, user control, immediate feedback, and forgiveness, and this consideration for the humanity of the user makes Apple's mouse-based interface unique.

Human Interface Principles

Here are the cornerstones of the human interface gospel according to Kamins, Espinosa, and Tognazzini:

Consistency: "People are creatures of habit," says Kamins. "They like things to work the same way every time. The reason that auto-

***Figure* 2 ▪ 11**
Scot Kamins, Chris Espinosa, and Bruce Tognazzini

mobile rental places are successful is that every car is essentially the same. It doesn't matter if it's a Volkswagen or a Cadillac. A car is a car. If there's a clutch, it's always to the left of the brake. The gas pedal is on the right. The steering wheel is always in front of you. It's always round. You start the car with a key. Once you've learned to drive one car, you can drive any car. The interface is consistent. That's how computer programs should be."

Consistency hasn't been something you could count on in the Apple II world. Among the 10,000-plus programs designed for the Apple II Plus, Apple IIe, and Apple IIc, there are probably a thousand different interfaces. The way you start up the program, load a file, and give commands can vary dramatically from one program to another. You might get a screen listing commands at the beginning of a program and never see it again; you might have to make your way through layers of menus; you might accomplish things by typing odd combinations of the shift key, the Apple key, and various letter keys. You never know exactly what to expect from a new program—an interesting challenge for some users, but an annoying impediment for others.

With applications designed especially for the Apple IIGS, you won't have to deal with that level of inconsistency. The mouse interface is the standard Apple IIGS interface. As with Macintosh, all Apple IIGS software developers get tools for developing mouse-based applications and a set of guidelines that tell them how a mouse-based program should work. Disk icons, windows, and menus will work the same way from one program to another. Standard actions like double-clicking will yield predictable results. Icons will look like what they represent.

This alone is a giant step toward a more human interface. It protects you from unpleasant surprises (such as discovering that clicking on an icon gives you two radically different results in two different programs), and it saves you a lot of learning time by ensuring that once you've gotten familiar with one program, you've learned the basics of them all. "Users should be able to sit down with a piece of software and do something useful in twenty minutes," says Tognazzini. "If they can't, there are probably problems in the interface." He recommends that you subject any Apple IIGS software you're thinking of buying to this test. "If you feel stupid at the end of twenty minutes," he says, "try someone else's software. A piece of software can be powerful and complex, able to grow with you and your developing needs, without making you feel like you're walking into a brick wall the first time you use it."

Control: "In a well-designed program, you make the decisions," says Kamins. "A dialog box comes up and says, 'Listen, if you do that, you're going to blow yourself out of the water.' It doesn't say, 'I know you don't want to do this, so I'm going to clean it up for you.' It says, 'Listen, here's what's happening. Shall I clean it up for you?' You should be able to decide where you want to go and what you want to do. You're the human." (There's more about the dialog box in the next section.)

There is such a thing as a program that's too helpful. One of the early implementations of the mouse interface tried very hard to make things easy for the user and wound up taking control out of the user's hands. Menus would pop up on the screen with a choice already selected—the choice the program *guessed* was the one the user wanted. If he didn't want that choice, he had to do something explicit to cancel it before making the choice he wanted in the first place. The program, by trying to do the user's thinking for him, actually got in the way.

Interface features like pull-down menus and dialog boxes allow you more control over your work than, for example, the rigid, many-layered menu structures and mystifying error messages of older programs.

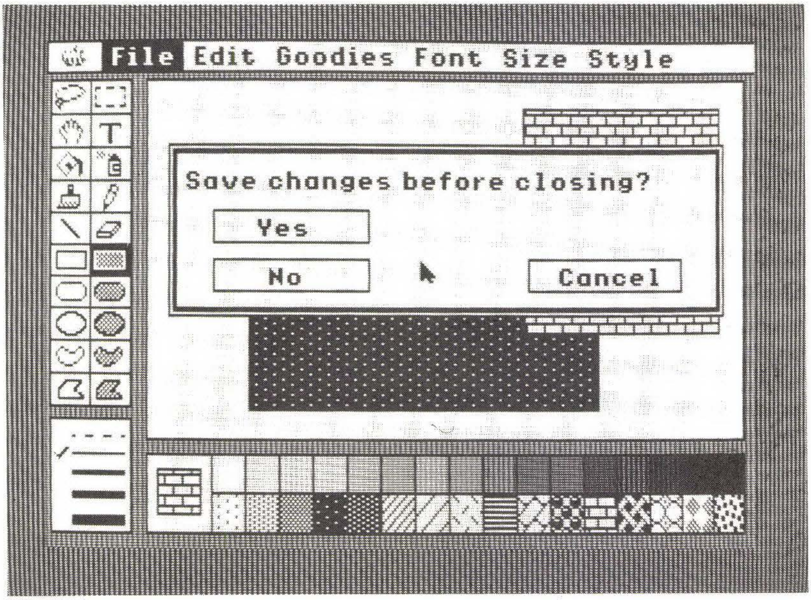
Immediate Feedback: "When you do something, you want feedback," says Kamins. "When the program highlights something or beeps, you know something is happening. There's a real dialog going on. Don't you hate it in a relationship when you can't tell if you've had an impact? You feel like saying, 'Hello, is anybody in there?' Well, it's the same with computers. You like to know you're having an impact."

There are two aspects to immediate feedback. One is what the computer "says" to you. The other is how the computer "feels" to you.

A tenet of the Apple interface guidelines is that the computer must let you know, promptly and clearly, what's going on. This happens by means of *dialog boxes*—messages that come up on the screen when you have to make a choice or when something significant is about to happen; they give you an opportunity to choose what you want to do or back out (Figure 2.12).

The messages are in English, not computer jargon, and it feels as if a dialog is taking place between you and the machine. The tone of this dialog establishes a kind of personality for the program. A program that says, "Disk not on line" feels less congenial than one that says, "Please insert the disk Office Memos"—partly because it sounds nicer, but also because it tells you not only what's wrong, but what

***Figure* 2 ▪ 12**
Dialog box



Save changes before closing?

to do about it. Who wouldn't rather work with a machine that's helpful, understanding, and easy to talk to?

The way a program responds "physically" is as important as how it responds verbally. "Direct manipulation works only if the program provides instantaneous feedback," Espinosa says. "When you move the mouse, the thing you're dragging should move with you. If you make a motion and the computer has to think about what you did, analyze it, and do the action later, there's a visual lag. Things move in a jerky way." As a result, that feeling of a close communication is diminished. Apple has put a lot of programming expertise into the mouse interface on the Macintosh and the Apple IIGS in order to eliminate the time lag between actions and results.

Forgiveness: "You're human," says Kamins. "You louse up. You want another chance. A lot of programs say, 'No way.' Good programs say, 'Okay.'"

Even with the most thoughtfully designed applications, there is potential for disaster. You can accidentally save a blank page over a document you've worked on for hours; you can erase the wıong disk; you can crash your system by entering the wrong command at the wrong time. An application that lets you plunge blindly into catastrophe is not the kind of partner you want. The Apple mouse interface is replete with warnings and ways out. Just about every dialog box has a Cancel button, so that if you find yourself confronted by two choices and neither of them is the one you want, you can back up and start over. In many applications there's a menu choice called "Undo," which undoes everything you did since you last clicked the mouse button. This can be very handy if you absentmindedly delete the wrong paragraph or paint your entire picture black.

In fact, forgiveness is built into the way the mouse works. "One of the most important parts of the user interface is what happens when you push the mouse button down and release it," Espinosa says. "In the Apple interface, nothing's supposed to happen until you press and release the mouse button. Most things don't happen until you release the mouse button on the same thing you pressed on. If you press, change your mind, drag away, and lift, nothing happens. You're safe. A lot of that is different in other user interfaces. Things zoom around because of unintentional clicks, and you do things you didn't mean to do."

Another safeguard is the dimmed command. A program's commands are always visible on pull-down menus, but whenever you're at a point where a certain command won't work or will cause trouble, that command is dimmed—it appears in gray type rather than black and is not selectable. This feature not only protects you from possibly unpleasant consequences, it also saves you time: you don't waste minutes typing in a command, getting an error message, and figuring out what you did wrong.

These characteristics of the Apple human interface—consistency, user control, immediate feedback, and forgiveness—make the Apple IIGS a congenial computer. More important, they allow you to maintain the closest possible connection with your work. You point and you're there. You move your hand, and something moves on the screen. You can almost forget that there's a computer standing between your actions and your final results.

The Mouse and the Apple II

Around 1983 Apple decided to make it possible to use a mouse with the Apple IIc, and Bruce Tognazzini was given the job of figuring out how the interface for it was going to work. Although he worked with the Apple II group, he knew a lot about the mouse interface that was then under development for the Macintosh. He had worked with Jef Raskin (who originated the idea for Macintosh), and he had good communication with the Macintosh group at Apple. "The first thing I figured out," he recalls, "was that a mouse interface couldn't be done on the Apple II because there wasn't enough speed. There was plenty of speed for text, but conventional wisdom dictated that a mouse interface had to be done in graphics, and there wasn't enough speed for that. Then, as I was lying in bed one morning at about three o'clock, I began to wonder whether we really needed graphics, and out of that I developed the *MouseText* concept. I analyzed everything that happens graphically in the average spreadsheet, word processor, data base, and so on, and I figured out that really there were just a few basic graphic elements that had to be there—a few different lines, squares, diamonds, things like that. So I developed the *MouseText* character set."

MouseText enabled software developers to write very fast if somewhat crude-looking mouse software for the Apple II family. Because neither the IIe nor the IIc came with a mouse, and early Apple II mouseware was a pain to write, not much mouse-based Apple II software was written. Tognazzini, however, was working toward the day when an Apple II could handle a full-fledged mouse interface. "For several years," he says, "I carried out a sort of underground operation to get our interface for the Apple II ready for the mouse." Part of this effort went into the design of the Apple IIGS: "The Apple IIGS team was very careful to get as much speed out of the graphics as possible, so we could do high-quality mouse-based software," Tognazzini says. But he was also setting the scene for the mouse within the existing Apple II family by designing an interface for *AppleWorks* that would be as close as possible to a keyboard equivalent of the way mouse software works. If you look at the *AppleWorks* interface, you see that it doesn't clash with the mouse interface. The cursor behavior is exactly the same, for example. From the time Lisa came out, Tognazzini was laying the groundwork for the mouse so it could fit in without conflict.

How the Mouse Affects Applications

The Apple IIGS, with its faster processor, larger memory, and better graphics capability, has finally made it all possible. Apple IIGS programs are going to be unlike anything seen in the Apple II world before. The mouse interface, after all, is more than a new way of doing things; it also adds enormously to the number of things you can do. It's as if Ford put a new gear on its cars called "Up," and when you went into Up, the car would rise off the ground and take you from home to the shopping center without going down a single traffic-clogged street. "A whole new dimension in driving!" the ads would say.

The mouse provides a whole new dimension in computing. When you point, click, and drag, you can do things that just weren't possible (or were so cumbersome that they might as well not be) when the keyboard was the only means of communication. At the bottom of it all is graphics—not just the technology that makes superior graphics possible, but also the idea that how something looks on the screen is just as important as what it says. This philosophy has naturally made a tremendous impact on programs whose purpose is drawing and designing, but it has spread out into every other kind of program as well.

Paint and Draw Programs

The mouse has revolutionized computer graphics. Putting a drawing on a computer screen used to be more a mathematical than an artistic undertaking—a complex matter of determining coordinates and plotting points. There was no particular resemblance between the process of drawing on paper and the process of drawing on a computer. Mouse-based graphics programs have shrunk the distance between those two processes. You work in an art environment instead of a mathematical one, using the equivalent of artists' tools and doing your drawing on the equivalent of a canvas or a drawing pad. Most important of all, the physical action of drawing a picture on a computer screen is no longer completely unrelated to the physical action of drawing on paper. You move the mouse around on the Desktop in more or less the same way as you move a brush or a pencil.

Not quite the same way, of course. The pencil and the mouse are still very different implements, with different advantages. With a mouse-based graphics program, you might not be able to achieve the sensitive lines or subtle emotional effects of a pencil drawing, but you can do other things that the pencil is quite incapable of:

- You can erase what you've done, hundreds of times, if necessary, without leaving a trace.
- You can draw a profile that faces right, flip it so it faces left, and then make five identical copies of it.
- You can draw something very small and stretch it to twice its size.
- You can pick up what you've drawn and move it to a different spot on the page.
- You can fill an entire area with color or pattern in an instant just by clicking the mouse button.
- You can easily try out lettering in various sizes and styles to see which looks best.

Graphics programs for the Apple IIGS incorporate all these capabilities and more. (See box.) They bring sophisticated graphics within the reach of people who aren't graphic artists but who care about the appearance of what they do.

DeluxePaint

A graphics program from Electronic Arts

Electronic Arts, publishers of such Apple II classics as *One on One, Pinball Construction Set,* and *Music Construction Set,* gets most of its software from outside developers, but

DeluxePaint was written by Electronic Arts insider Dan Silva.

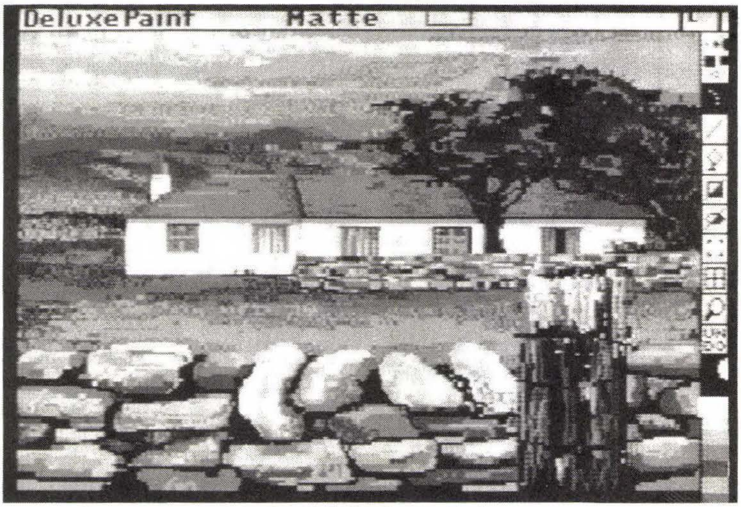
The program was originally conceived as an in-house art tool for programmers under contract to Electronic Arts (it was used, for example, to create the scenes and characters in Mail Order Monsters), but Dave Grady, the program's "producer" saw that DeluxePaint could appeal to a much wider audience. "It's a 747 in terms of features," says Grady, "but beginners can fly it." (Electronic Arts is modeled after the record and movie business in that programmers are called "artists" and product managers are called "producers.")

Some of the things people have thought up to do with the program come as a surprise even to Silva, who has lived with various versions of the paint program for four or five years.

"A guy called me from Houston," recalls Silva. "He was co-partner in some big architectural firm. He told me they had a $100,000 CAD CAM system but that they were using *D-Paint* instead. They like it because it lets them create a setting on the main canvas, then bring in different buildings from the spare screen to help clients visualize what they're proposing for a given site." The spare screen is a *DeluxePaint* feature that lets you create parts of your picture on a second canvas and copy them into the main picture when you're happy with them or, in the case of the architecture firm, when you want to wow a client.

The cut and paste feature in *D-Paint* works differently from the ones in other paint programs. Instead of cutting and pasting, you define the thing you want to move as your "brush." Once you've defined something as a brush, it moves around the screen in response to your mouse moves just as any other kind of cursor does. The brush can be a star, a leaf, a race car, a cartoon character, or anything else you can imagine. Animators have used this feature to create dynamic storyboards. They draw a background, define a character or object as the brush, and videotape the screen as they move the "brush" around the screen.

One of the things that makes *DeluxePaint* popular with



Farmhouse created with **DeluxePaint**

professional artists is the degree of color control. Choosing the 16 colors you want to work with (from the 4096 colors available) is similar to the way artists mix paint on a palette. For the first time, there is enough color control for artists to take the Apple II seriously as a drawing tool.

DeluxePaint comes with demo art that illustrates the program's possibilities. The samples range from what Grady calls "our classy images" (King Tut and Boticelli's Venus) to a space ship with aliens—the kind of scene you might find in a game program. There's also a landscape featuring what appears to be a cascading waterfall. The program's cycling feature creates the illusion of running water. When you turn on cycling, each dot in a given part of the picture can cycle through the 16 color palette. In the waterfall picture the dots are cycling through different shades of blue, thus making the water look as if it's moving.

One of the program's more interesting applications is as a portrait construction set. You start with a drawing of a blank face and an assortment of noses, ears, eyes, and mouths in different shapes and sizes. You construct portraits by dragging facial features to the face. It's as easy as making a face with Mr. Potato Head, but the results are as professional as police sketches.

It's not surprising that Electronic Arts thought up this application for Silva's art program. "We're the company that made construction sets famous," says Grady. *DeluxePaint* is one of four programs in the Deluxe series—programs that Electronic Arts promotes as "creativity productivity" tools—the artist's equivalent of business productivity tools like word processing, data base, and spreadsheet applications. The others in the series are *Deluxe Music Construction Set, DeluxePrint,* and *DeluxeVideo.*

Deluxe Music is a composition program, a word processor for musicians. *DeluxePrint* is a program for creating banners and greeting cards—like the classic *Print Shop* from Brøderbund, but with greater flexibility in layout and an opportunity to preview your creation before you print it. *DeluxeVideo* is used for creating special video effects: fades, wipes, dissolves, rotating images, and the like. "You can do almost anything you can do in a locally produced TV commercial," Grady claims. "If you're a video hobbyist, you'll have to get one just to create your titles."

The programs in the Deluxe series are integrated, which means the cards you make with *DeluxePrint* can use images you made with *DeluxePaint,* and your *DeluxeVideo* productions can include *Paint* images and a soundtrack created with *Deluxe Music.* (See box on page 74.)

Page Layout Programs

Page layout programs are a fairly recent phenomenon, but they have already sparked an explosion called "desktop publishing." Things like newsletters, advertising brochures, instruction booklets, catalogs, and annual reports are ordinarily laid out (at some expense) by graphic artists using scissors, glue, rulers, and rub-on letters, then sent to the typesetter and on to the printer. Now the entire process can be done by a nonexpert and a page layout program. You can pull your text and graphics from a data disk right into the page you've designed and move them around on the screen until you have them the way you want them. You can put your text in columns, edit it once it's placed, and resize or crop your drawings. The finished product, when printed on a LaserWriter, differs very little from typeset quality. (See box.)

A page layout program from Quark, Incorporated

Quark (the company that worked closely with Apple on mouse-based applications for previous Apple II's, and the developer of *Word Juggler,* one of the most popular Apple IIe word processing programs) is planning a program that combines the features of a page layout program and a word processor. "In this program, there's no distinction between text and graphics," says Tim Gill, director of R&D at Quark. "You can put in pictures and move them around just as easily as you can move text." Current page layout programs (most of them designed for Macintosh) require you to write your documents with a regular word processing program and then move them into the page layout program with a kind of cut-and-paste process. Once placed, the text can be edited, but not very conveniently. These page layout programs aren't designed as full-fledged word processors.

Quark's program *is* designed that way. It's a full-featured page layout program with all the capabilities you'd expect in a high-end word processor. You can type your articles, edit them, change their fonts, add pictures to them, and move the elements around—all within the same program. Up to eight documents can be open on the screen at once, and you can copy and paste between them—which means that if you're cutting pictures from a page of clip art and taking bits and pieces from several different articles, you don't have to keep opening and closing documents to do it.

The process gets even speedier when you add macros to it. The program lets you take any phrase or key sequence and make it equal to a single keystroke. If you get tired of typing "International Conglomerated Applications" all the time, for example, you can define a macro for it and cause the whole phrase to appear at the tap of a key.

You can resize your text and graphics once you've got them on the page, and for decorative effect you can add rules and frames. Once the whole thing is arranged the way you like it, you can proof it with the built-in spelling checker.

Although Quark's program provides all the power you need to create your newsletter, ad, or brochure, you can also import text and graphics easily from other sources. Quark's product is flexible enough to do everything current page layout programs can do—but its new features make it a next-generation product.

Word Processing Programs

Suddenly, you can decide how your text will *look* as well as what it will say. Do you want a small, businesslike typeface or a large, flowing one? Do you want headings in 24-point boldface and the rest in 12-point plain? Do you want certain words italicized, or underlined, or even printed in outline type? All these things are possible with a graphics-based word processor.

This may seem like a frill. After all, the content of a page of text is a lot more important than its appearance, isn't it? The trouble is, a page that looks dull and difficult is often ignored in favor of a page that looks attractive and accessible. Graphics-based word processors let you make your text more attractive than you ever could before, short of taking it to a typesetter. This capability has particular value for people who make presentations, or write memos that compete with other memos for attention, or design newsletters or advertising. Even if you do none of these, though, you're likely to find that having a choice of sizes and styles for your text inspires you to make what you write look as appealing as possible.

The mouse has brought about practical changes in word processing as well as esthetic ones. With an old-style word processing program you pressed the arrow keys to move the cursor to where you wanted to type, delete, or change something. Even though there were special key combinations that let you jump to the end or beginning of a document, or jump several words or paragraphs at a time, you might still have to do a lot of key presses to get from a spot in the third line of the second paragraph to a spot in the fifth line of the twenty-eighth paragraph. When you have a pointer on the screen instead of a cursor, you can simply point to the spot you have in mind. The pointer doesn't have to follow a track through the text. It jumps over it.

Some people argue that the mouse doesn't make sense in a text-based application like word processing, because it requires you to take your hand off the keyboard. These are usually people who have gotten very good at keyboard cursor movements and feel as though they're wasting time when the keys aren't clicking away. Certainly there's a place for arrow keys, however: if you're moving the cursor one letter to the left, it's easier to hit the left arrow key than to maneuver with the mouse. Fortunately, the Apple IIGS interface gives you a choice; if you'd rather use the keyboard than the mouse, you can.

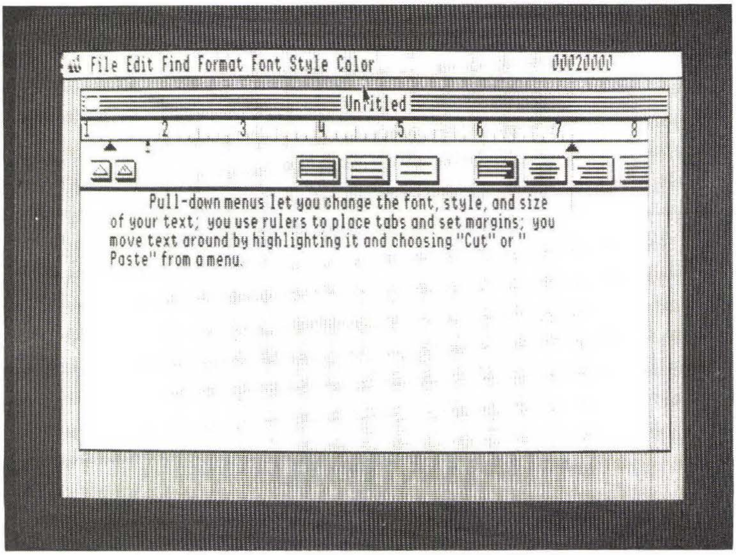
An Apple IIGS word processor will be coming from Version Soft, one of the companies that designed mouse software for the Apple IIe and IIc and therefore has a head start over other developers of Apple II mouse applications. (See box.)

A word processing program from Version Soft

Word processing is going to look different on the Apple IIGS. Version Soft's new program takes advantage of the Apple IIGS features to give you word processing capabilities that haven't been seen before in the Apple II world.

If you're familiar with the Macintosh program called *MacWrite,* however, you'll recognize this program. It works almost exactly the same way. Pull-down menus let you change the font, style, and size of your text; you use rulers to place tabs and set margins; you move text around by highlighting it and choosing "Cut" or "Paste" from a menu. What makes this program different is its use of color, not just to decorate the page but to add functions to the program.

Suppose you're writing a report. You get the first draft finished, and then you start revising. A sentence needs to be changed. You select it and rewrite, and as you do a light blue highlight appears across the words you've replaced. When you've finished editing, there will be blue highlights everywhere you've made changes, and if you decide in any of those places that you liked the first version better than the second, you can go back to it. The program saves the original text behind every change you make.



Pull-down menus let you change the font, style, and size of your text; you use rulers to place tabs and set margins; you move text around by highlighting it and choosing "Cut" or " Paste" from a menu.

Version Soft program

There are other color features, too. By drawing a red line through a word, you can cross it out and yet leave it visible in case you later change your mind. You can make your text appear in color and, with a color ribbon on the ImageWriter II, you can print in color. You can also paste in pictures from another of Version Soft's Apple IIGS products, a paint program that makes spectacular use of the Apple IIGS's 4096 colors. When you add these color features to all the benefits of MacWrite, you get a word processor that's fast and easy to use, provides you with a lot of power, and also looks lively and cheerful on the screen.

The Version Soft program will print to either the ImageWriter II (for color) or the LaserWriter (for typeset-quality text and graphics). With a program like this, you can produce—without undue expense and bother—written material that will grab people's attention and hold it.

Spreadsheet Programs

No one who has used a spreadsheet program will have much trouble seeing the advantages of a mouse interface over a keyboard interface for doing financial modeling. To do useful things with a spreadsheet, you have to refer to its cells. You have to tell the program to add the numbers in cells A16 through A38, divide that total by the difference between cells C23 and D23, and then mulitply the whole thing by what's in cell RR47. In a keyboard-based spreadsheet, that takes either a lot of tedious typing or a lot of cursor movements. With a mouse, you can avoid both. You simply point at the cell you want to refer to—or, if you're referring to a whole group of cells, you drag across them or surround them with a selection rectangle. It's a direct indication, accomplished in one movement ("I mean *these* cells"), rather than a verbal description ("I mean cell A42 through E28").

A natural for a mouse-based spreadsheet program is the graphic connection. You can expect the new spreadsheets to translate numbers into graphs with speed and polish and to provide graphic features not seen on old spreadsheet programs, such as the ability to ask what-if questions not only by changing your numbers and observing the effect on the graph but by actually changing the graph itself: grabbing one of the bars of a bar graph, for instance, and pulling it up from 30 on the scale to 40. Color will add to a graph's appeal and clarity, and so will graph titles in various fonts and styles.

VIP Software is developing a spreadsheet program for the Apple IIGS, as are a number of other companies. (See box.) From all indications, these spreadsheets will be powerful, easy on the eyes, and startlingly simple to use.

VIP Professional

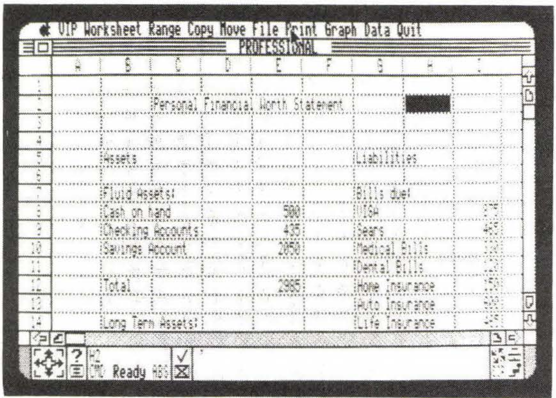
A spreadsheet program from VIP Technologies Corporation

Among microcomputer spreadsheet programs, *Lotus 1-2-3* has become something of a standard. "Millions of people use it," says Tom Nelson, vice-president of VIP Technologies Corporation. "There are books about it, templates for it, training courses on it. That's not to denigrate other programs, but the business world seems to have chosen this one." That's why VIP has designed its spreadsheet program, *VIP Professional,* to be completely compatible with *Lotus 1-2-3.* The commands and functions are the same; you can use all the files that already exist; you can transfer files between computers. Everything you can do with the Lotus program, you can do with *VIP Professional.*

And, according to Tom Nelson, you can do it better. "In my opinion," he says, "a spreadsheet that has a mouse is about five times easier to use than one that doesn't. The ability to access portions of your spreadsheet is essential, and the mouse interface, where you have scroll bars, paging icons, things like that, lets you do it much faster."

There are other advantages, too. You get to cells by pointing at them rather than pressing the arrow keys over and over; you specify ranges of cells to be used in calculation by dragging across them; and, instead of having to type and memorize commands, you choose them from pull-down menus.

VIP Professional includes fifty math functions, data base commands for sorting and arranging your information, and a built-in graph program. It also makes use of the macros created for Lotus by other companies. These are mini-programs that you can activate with a command, which enable you to create spreadsheets for very specific purposes: an accounts payable package, a curriculum guide, an automated payroll program. You can use the macros already created by Lotus and other companies, or you can create your own.



	A	B	C	D	E	F	G	H	I
1									
2			Personal Financial Worth Statement						
3									
4									
5		Assets					Liabilities		
6									
7		Fluid Assets:					Bills due:		
8		Cash on hand			500		VISA		[illegible]
9		Checking Accounts			435		Sears		[illegible]
10		Savings Account			2050		Medical Bills		[illegible]
11							Dental Bills		[illegible]
12		Total			2985		Home Insurance		[illegible]
13							Auto Insurance		[illegible]
14		Long Term Assets:					Life Insurance		[illegible]

***Screen from* VIP Professional**

Except for the impressive color that the Apple IIGS contributes to the graph program, you can find all these features in 1-2-3. But VIP goes beyond 1-2-3 in several ways (in addition to the use of the mouse interface). It's a larger spreadsheet, and it can address more of the computer's memory than Lotus 1-2-3 on an IBM PC. "With our program," says Nelson, "even a 256K machine will have a sizable worksheet—and of course you can slam in a megabyte board, and it'll address all that memory quite easily. The IBM PC has an artificial limit of 640K, and you have to buy an extra board to go over the limit."

Because windowing capabilities are built into the Apple IIGS, VIP Professional can let you see more than one window on the screen at once. You can, for example, look at your spreadsheet and your graph at the same time. You can't do this with Lotus on the IBM.

VIP Professional is faster than Lotus on the IBM, too. "The program is straight machine language all the way," Nelson says. "It just zings on the Apple IIGS."

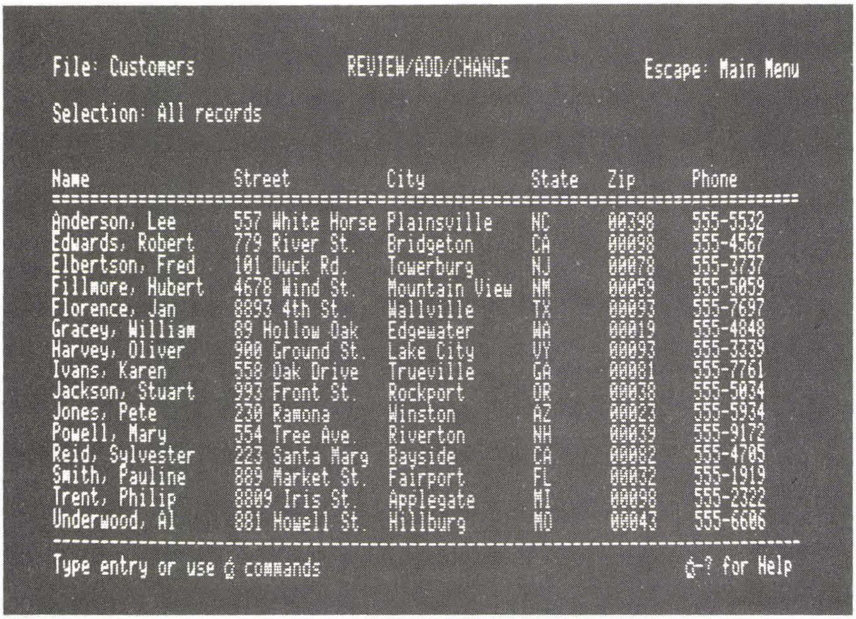
The company seems determined to maximize that zing, so you can probably expect additions to the list of VIP features not offered by 1-2-3. "We recognize the fact that the state of technology is moving ahead, and we're not going to be stopped by an artificial standard," Nelson says. "First release will be totally compatible with Lotus 1-2-3; what we do beyond that is another question."

Data Base Programs

A mouse-based program can allow you to change the structure of your information just by dragging it around on the screen. This means you can change the width of a column, for example, by grabbing the column divider and moving it instead of pressing a combination of keys, or you can select information to copy or cut by dragging across it rather than moving cursor keys to highlight it.

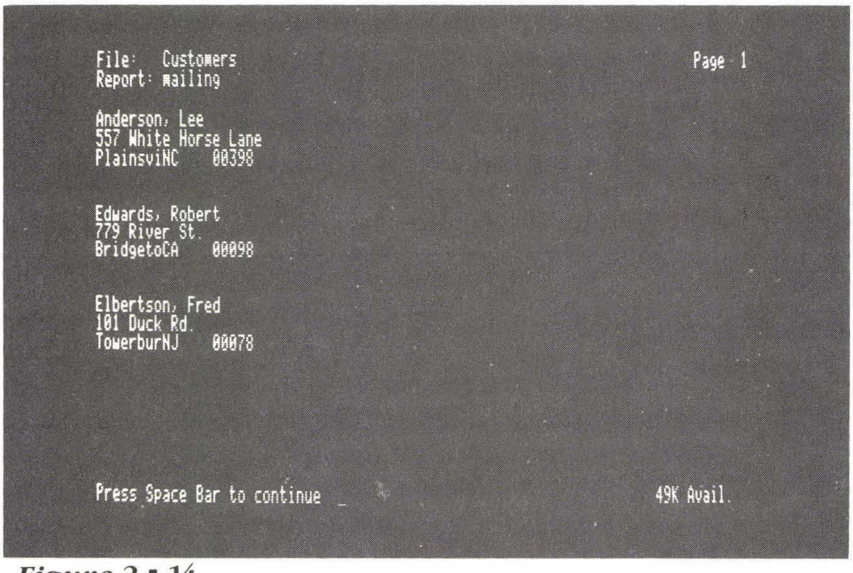
You can also set up your data base records much more flexibly than with old-style programs, which usually supply a standard multiple-record format and a standard report format (Figures 2.13 and 2.14).

A mouse-based program lets you lay out your information in whatever way is most helpful to you. You can drag your information



```
File: Customers              REVIEW/ADD/CHANGE              Escape: Main Menu

Selection: All records

Name             Street           City           State  Zip      Phone
===============================================================================
Anderson, Lee    557 White Horse Plainsville     NC     00398    555-5532
Edwards, Robert  779 River St.   Bridgeton       CA     00098    555-4567
Elbertson, Fred  101 Duck Rd.    Towerburg       NJ     00078    555-3737
Fillmore, Hubert 4678 Wind St.   Mountain View   NM     00059    555-5059
Florence, Jan    8893 4th St.    Wallville       TX     00093    555-7697
Gracey, William  89 Hollow Oak   Edgewater       WA     00019    555-4848
Harvey, Oliver   900 Ground St.  Lake City       VY     00093    555-3339
Ivans, Karen     558 Oak Drive   Trueville       GA     00081    555-7761
Jackson, Stuart  993 Front St.   Rockport        OR     00038    555-5034
Jones, Pete      230 Ramona      Winston         AZ     00023    555-5934
Powell, Mary     554 Tree Ave.   Riverton        NH     00039    555-9172
Reid, Sylvester  223 Santa Marg  Bayside         CA     00082    555-4705
Smith, Pauline   889 Market St.  Fairport        FL     00032    555-1919
Trent, Philip    8809 Iris St.   Applegate       MI     00098    555-2322
Underwood, Al    881 Howell St.  Hillburg        MO     00043    555-6606
-------------------------------------------------------------------------------
Type entry or use  commands                                        -? for Help
```

Figure 2 ▪ 13
Standard multiple-record format



```
File:   Customers                                        Page  1
Report: mailing

Anderson, Lee
557 White Horse Lane
PlainsviNC    00398


Edwards, Robert
779 River St.
BridgetoCA    00098


Elbertson, Fred
101 Duck Rd.
TowerburNJ    00078







Press Space Bar to continue _                          49K Avail.
```

Figure 2 ▪ 14
Standard report format

fields (say, a client's name, address, phone number, and social security number) anywhere you want on your electronic "index card": you might put the phone number in the upper right corner, for instance, or center the name at the top.

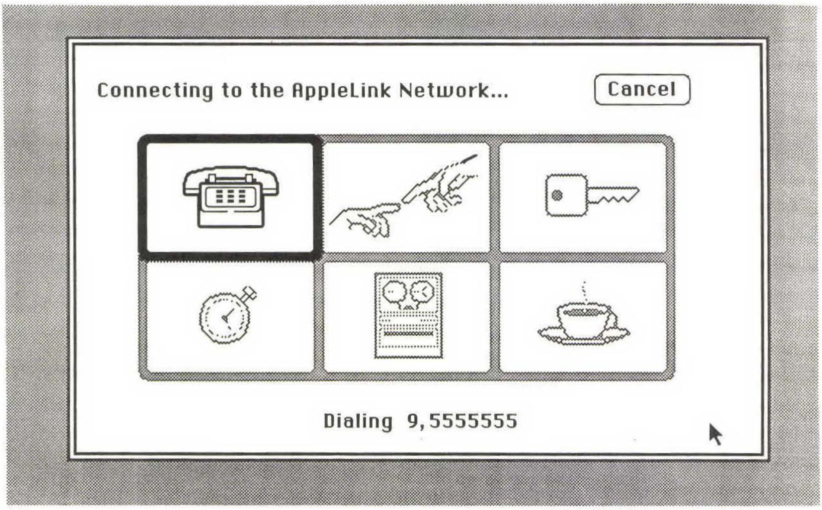
Changing from one layout to another is just a matter of dragging rectangles around on the screen. This kind of flexibility makes a mouse data base program a tool you can customize to a degree not possible before.

Communications Programs

Communications programs usually look like a mess on the screen. First there are all the parameters you have to set to make sure that your computer is going to send information the right way to the other computer. Then there are the procedures you have to go through to log on to whatever service you're using. To deal with all this, you get lists of technical questions to answer and often mysterious prompts on the screen. You need the program manual by your side, at least until you've memorized the commands.

A more visual approach helps a lot. It can't eliminate the need for setting parameters, but it can provide less cryptic log-on procedures. Figure 2.15, for example, shows the main screen of an Apple intracompany communications program.

***Figure* 2 ▪ 15**
AppleLink *screen*



Connecting to the AppleLink Network...

Dialing 9,5555555

MouseTalk

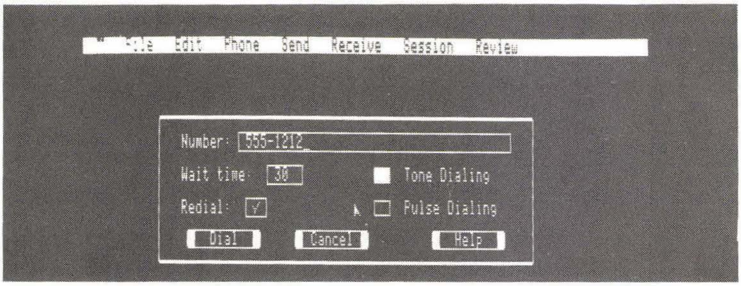
A communications program from United Software Industries

The whole field of data communications is about to explode. At least that's what the pundits tell us, and have been telling us for some time now. Still, only a fraction of personal computer users are hooked up to other computers. If this is an explosion, it has a long fuse.

What's to blame, among other things, is the complexity inherent in making one computer talk to another. Different computers operate in different ways: they send data at different speeds and in different formats, and they have different codes to announce the beginning and end of a piece of information. A lot of questions have to be answered, and anyone who uses a communications program has to answer them.

But it needn't necessarily be difficult to do. United Software Industries is coming out with a new program called *MouseTalk* whose intent, according to Executive Vice-President Mark Robbins, is to be "so simple to use that it's a real pleasure." An easy-to-use communications program is almost a contradiction in terms, but United Software probably has a better chance than most companies to create such a thing. They are the developers of ASCII Express Professional, a best-selling communications program for the Apple IIe, known for its speed and its ability to do just about any kind of communications. "It's fairly difficult to use," Robbins acknowledges, "because of its complicated nature." But the company has accumulated a lot of information about what parts of it are difficult and why; a call-in technical support division has kept careful records of users' questions and complaints. "We're taking all we've learned from ASCII Express and putting it into *MouseTalk,*" Robbins says.

"Ninety percent of our questions about ASCII Express were installation questions," he says. "So we made *MouseTalk* incredibly simple to install." The first time you run the program, you get a menu that lets you immediately



Number: 555-1212_

Wait time: 30

Redial: ☑

■ Tone Dialing

□ Pulse Dialing

MouseTalk ***screen***

format a blank disk and copy the master program disk onto it; you don't have to do this with a utilities program before you can start *MouseTalk*. Once you've made your copy, you put the master away in a safe place, and the program goes on to ask you a series of straightforward questions: What kind of computer do you have? What kind of modem? What kind of printer interface? Once you've answered these, you can get to work.

MouseTalk takes full advantage of the intelligence of today's modems. "When a smart modem connects with a computer, it reports back what speed it connected at," Robbins says. "We make use of that feature by setting the speed according to what the modem has told us." This is one less setting the user has to worry about, although if he does for some reason need to set a particular baud rate, he can open the appropriate menu and do it.

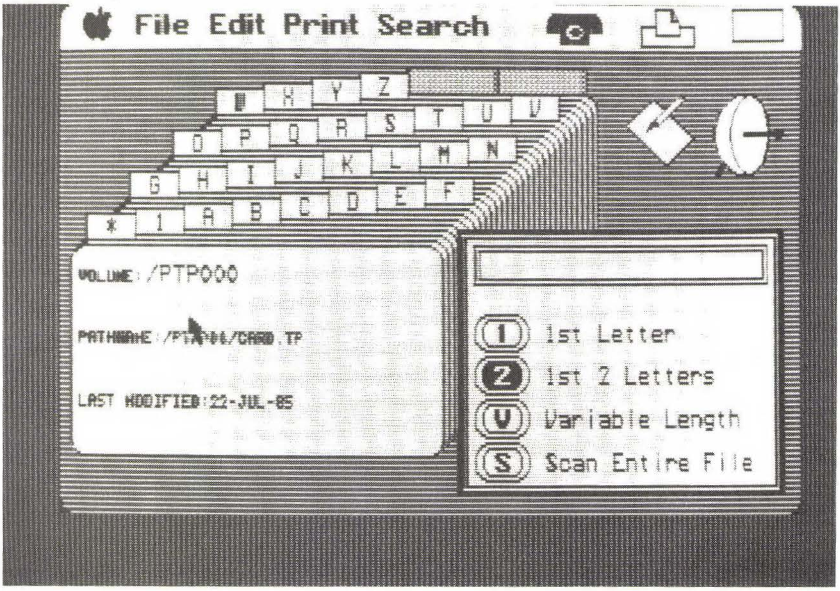
Whenever possible, the user is protected from the "ugliness" of the run-of-the-mill communications program. *MouseTalk* automatically decides on settings that have become fairly standard and hides such settings in a menu so you don't have to deal with them every time you run the program. There are three parameters, called *data bits, stop bits,* and *parity,* which have to do with the way data are sent and the type of error checking used by the sending and receiving devices. Most devices work just fine if you set data bits to eight, stop bits to one, and parity to none, so that's how *MouseTalk* sets them. If you need to change them, you can; otherwise, you never have to see them.

Another thing that can make a communications program less "ugly" is graphics (see the AppleLink illustration, page 69). Here, however, there's a trade-off involved: communication, even more than other kinds of programs, requires speed. Graphics slows things down. "We love the Apple IIGS's speed," Robbins says. "We can receive data with greater than 19,200 bits per second throughput. When you're hooked up to a mainframe, high speed is important, and we don't want to sacrifice that for a graphics screen." Users will no doubt agree—especially since *MouseTalk* is just about as attractive as if it did use graphics to draw its screens.

When you have speed, ease of use, and power, good looks may not be high on your list—but it's a valuable extra all the same.

Commercial programs are going in the same direction. *Person-to-Person,* a program from TruTec, provides a kind of electronic Rolodex file (Figure 2.16). The file cards hold the computer addresses of everyone to whom you send electronic mail. To post a message, you click on the card you want and give the command to send. The visual

Figure **2 • 16**
Screen from **Person-to-Person**



VOLUME: /PTP000

PATHNAME: /PTP000/CARD.TP

LAST MODIFIED: 22-JUL-85

(1) 1st Letter
(2) 1st 2 Letters
(V) Variable Length
(S) Scan Entire File

metaphor makes it all seem familiar, and therefore easy. The way you work with a communications program begins to feel like the way you work without one—another step toward the comfortably human interface.

Music Programs

Personal computers have come a long way since the days when beeps were their most expressive noises. The Apple IIGS has extraordinary sound capabilities, and software developers are busy finding new ways to take advantage of it. The most remarkable is the kind of music program that lets you compose and notate your music more or less simultaneously. (See box.) This kind of program is to music as a word processing program is to words: you can put notes on a staff, move them around, and erase them; you can change the key of whole pieces at once; you can listen to· the music as you go along; and when you have your piece the way you want it, you can print it out. In a way, programs like this are even more revolutionary than word processors: before word processors there were typewriters, after all, which let you put your handwritten words into more legible form. But until now there hasn't been anything for musicians between handwritten manuscript and final printed version.

Music programs aren't just for professional musicians. Students of music theory will find them useful; people who like music but have never learned to play an instrument can learn to play the computer in no time; and even those who are simply music appreciators can listen to the already-programmed selections that usually come on the disk simply by choosing a "play" command from a menu.

Cutting and Pasting Between Applications

An integrated application is a set of application programs designed to be used together. Among old-world programs, AppleWorks is the premier example: a word processor, a data base, and a spreadsheet all under one roof. Integrated applications let you take a file created with the spreadsheet, for example, and paste it electronically into the word processor. The files are made to be compatible. Several Apple IIGS developers are coming out with integrated applications.

The Apple IIGS, however, takes integration a step further than older Apple II's by establishing what is called a *standard file format.* This means that even applications that come from different companies can usually share files. You can, for example, put a picture drawn with

Deluxe Music Construction Set

A music program from Electronic Arts

Music scoring programs don't write the song for you, but they make it very easy to get the notes out of your head and into a form that you can play back and edit.

Deluxe Music Construction Set lets you build songs in one of two ways: by dragging notes to the electronic music paper at the top of the screen or by playing notes on the piano keyboard displayed at the bottom of the screen. When you point to a key and press the mouse button, you hear the note and it appears on the score at the insertion point.

Geoff Brown, the program's author, has a strong background in computer graphics, and it shows in his visually appealing program. One particularly nice touch, if you have a color monitor, is that the keys are ivory-colored just as they are on a real piano.

Brown worked in computer graphics at Xerox, then at Apple until 1984, when he left the corporate environment to write *Music Construction* Set for the Macintosh—a program that won the 1986 Software Publishers Award for best sound.

Music had been a hobby with Brown for a long time—he was the drummer in a band—but he had never mixed computers and music, and the idea appealed to him. "As a drummer I have a good feel for rhythm and I can hear and think about tunes and melodies," says Brown, "but I can't play an instrument that creates tunes. I thought that by working with a program that could do that, I might be able to express some of these musical thoughts I had."

Now he's working on a version of his award-winning program for the Apple IIGS, and he expects the sound to be even better than on the Macintosh because of the Apple IIGS's sound chip—the same chip that's at the heart of a professional music synthesizer called the Mirage.

There are three parts to a music composition program: composition, playback, and printing. Most programs focus on one or two. Brown gave equal emphasis to all three.

Composition: *Deluxe Music Construction Set* gives you all of the tools you need to build sophisticated musical scores, whether you're writing for one guitar or a 16-piece orchestra. The program's musical notation includes triplets, quintuplets, beams, slurs, and ties. "Notes processing" is easy because you can use the mouse to position notes and symbols and to set measure lengths and beam angles. You can also enter music by using a Musical Instrument Digital Interface (MIDI) keyboard.

Playback: You can play back your compositions by using either the Apple IIGS-generated sound or MIDI synthesizer.

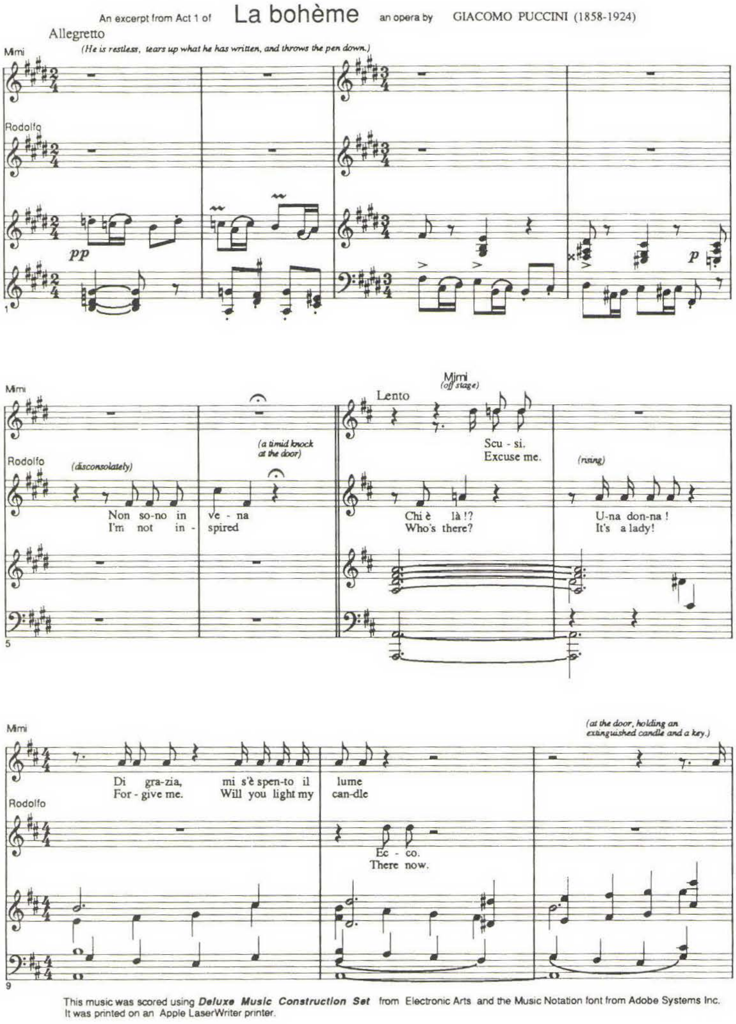
Printout: See for yourself. The program will be able to produce professional quality sheet music complete with lyrics as shown in the LaserWriter printout.

Deluxe Music Construction Set is designed for both professional and novice composers, says Brown, as long as novices don't expect the program to do the composing for them. He compares it to using a paint program or a word processor. "I can have a lot of fun and do some interesting things with a paint program, but I can't do real art because I'm not a real artist. It's the same with a word processor. Anybody can type, but it won't sound like Tolstoy unless you're Tolstoy."

While it is relatively difficult for new composers to write songs from scratch, almost anyone can take a song written by someone else and edit it—reinstrument it, add a different drumbeat, change a few notes. "That's the neat thing about a music composition program," says Brown. "It puts music in a plastic form that you can edit."

So Brown's program is accessible to serious beginners and powerful enough for most serious composers. Brown got a call from a singer in a barbershop quartet who told him that barbershop quartet singers have their own convention for how clefs are used. "They needed to be able to retune the program to conform to their convention, and I hadn't made that possible," Brown explains. "It will be possible on the Apple IIGS version of the program."

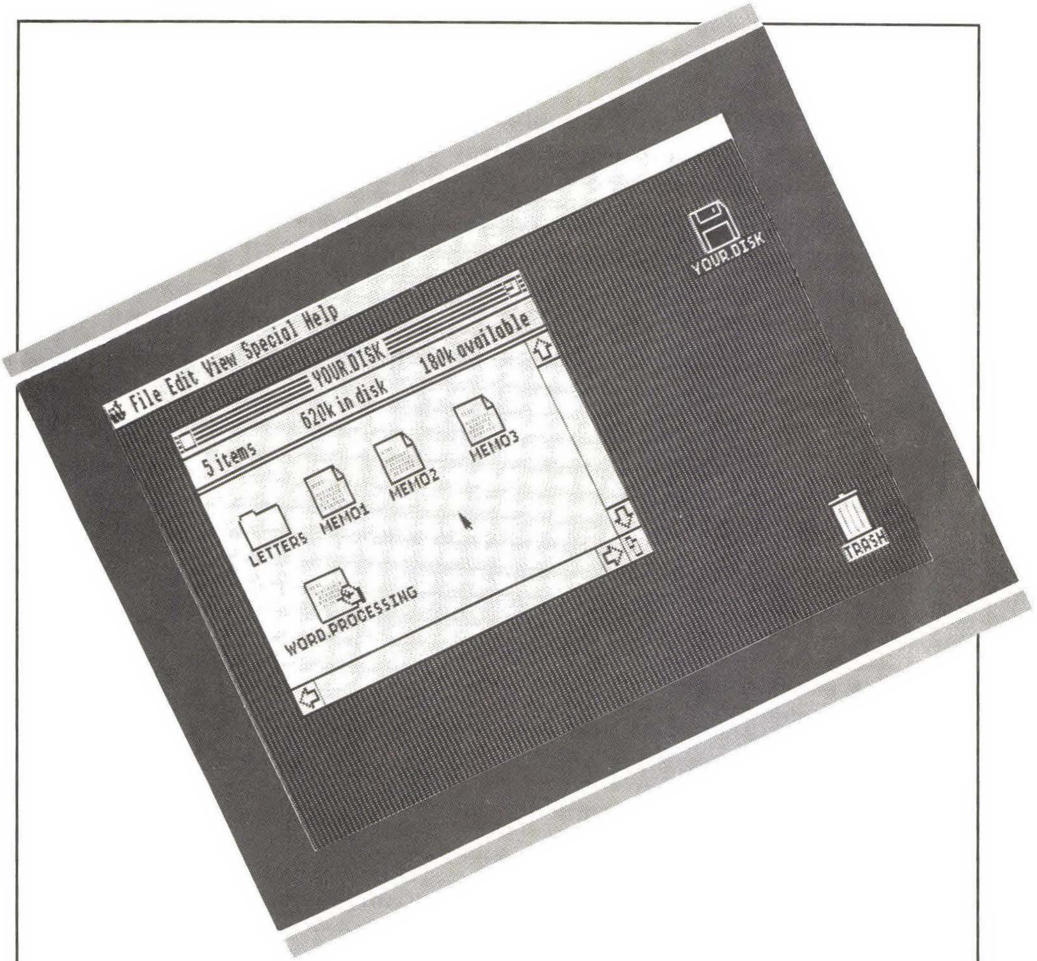
It's not that Brown has a soft spot in his heart for barbershop quartets, but he wants his program to be as flexible


An excerpt from Act 1 of La bohème an opera by GIACOMO PUCCINI (1858-1924)
Allegretto
Mimi
(He is restless, tears up what he has written, and throws the pen down.)
Rodolfo
pp
p
Mimi
Rodolfo
(disconsolately)
Non so-no in ve - na
I'm not in - spired
(a timid knock at the door)
Mimi (off stage)
Lento
Scu - si.
Excuse me.
Chi è là!?
Who's there?
(rising)
U-na don-na!
It's a lady!
Mimi
Di gra-zia, mi s'è spen-to il lume
For - give me. Will you light my can-dle
Rodolfo
Ec - co.
There now.
(at the door, holding an extinguished candle and a key.)
This music was scored using Deluxe Music Construction Set from Electronic Arts and the Music Notation font from Adobe Systems Inc. It was printed on an Apple LaserWriter printer.

La Boheme

as possible. "That's one thing you don't find in other scoring programs," says Brown. "They have a style and you work in that style or you don't work. I want people to get whatever they want out of this program."

one company's paint program into a letter written with another company's word processor, or use a column of figures from one company's spreadsheet to make a pie chart with another company's business graphics program. It's a convenient feature, and it allows you to choose the applications you like best, whether they come from the same place or not.



CHAPTER THREE

The Finder—
The Mouse Utilities

Let's say you've done your research and picked out the Apple IIGS software programs that suit your needs. Fired by enthusiasm, you forge ahead, writing poetry and annual reports, keeping track of your clients or your stamp collection, making graphs, playing music. Pretty soon you've created dozens of documents of all different kinds, spread out among twenty or thirty disks. You begin to forget where things are. Is your most recent version of "Tax.Deducts" on the disk called PR.1 or PR.2? *Is* "Tax.Deducts" the most recent version? Or is it the one you called "Tx.Dcts"? Where is your backup copy of "Letter.Sam"? Why are the three chapters of your novel all on different disks? You need to consolidate them. What in the world is on this disk called "Pat.NRJ.3"? It's high time, you realize, to get organized.

What it takes to get organized is a utilities program. With a utilities program, you can do all the housekeeping tasks that make for a manageable work space. You can, for example:

- List what's on your disks
- Move documents from one disk to another
- Delete documents
- Make copies of documents
- Group certain documents together
- Copy all the contents of one disk to another disk
- Format a new disk
- Erase and reformat a disk whose contents you don't need any more

All these utilities are part of a program called the Apple II Desktop—sometimes referred to as the Finder.

What Is the Finder?

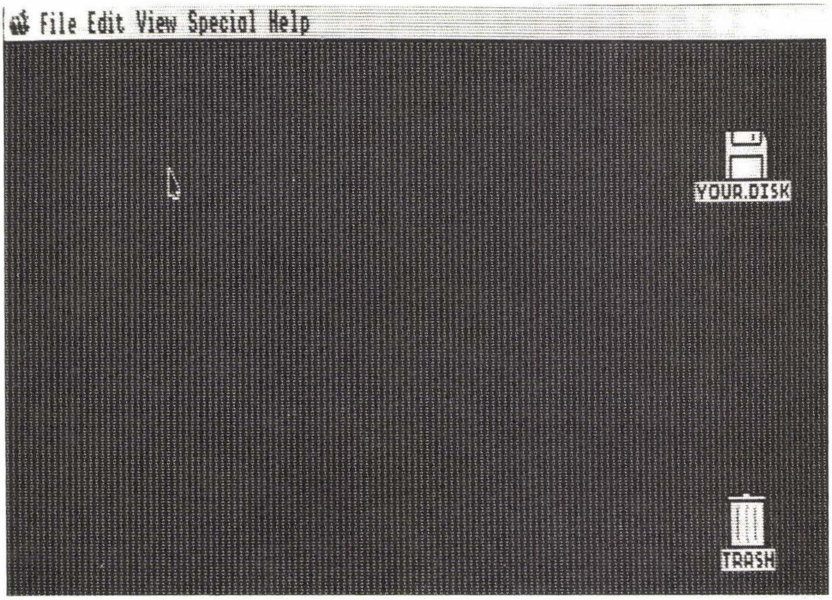
The Finder is a program that "finds" your disks in the disk drives, shows you what's on them in the form of icons on the screen, and lets you start up applications and do utilities-type work with disks and documents. You can use it with any ProDOS application. (You *can't* use it, however, with applications that use older operating systems like DOS 3.3 or Pascal. These require a program called System Utilities, which you'll find on your Apple IIGS system disk. You'll learn more about working with a variety of operating systems in Chapter 4. When you put an Apple IIGS application into the computer and turn on the power, you'll find yourself looking at a screen like the one in Figure 3.1.

It's called the Desktop, because the kind of work you do here is more or less like what you do at your desk: you shuffle papers around, you throw out the ones you don't need any more, you put things in file folders. The Finder, like all mouse-based applications, allows you to work within a familiar metaphor: how you work with the computer is as similar as possible to how you work without the computer.

What makes this possible is *icons*, small pictures that represent the things you're working with. You can move icons around on the screen just as you move pieces of paper around on the desk. Let's say you've written three memos and you want to put them all together so you can find them easily. Here's how you would do it, in two different environments:

- Pre-computer: You would take a manila folder out of your desk drawer, write "Memos" on it, and put the three memos inside.
- Finder: You would choose "New Folder" from the File menu, which causes a folder icon to appear on the Desktop (Figure 3.2). Then you would type "Memos" for the name of the folder (Figure 3.3).

Figure 3 • 1
Desktop



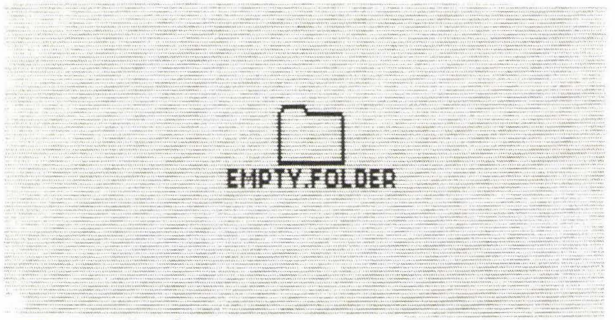


Figure 3 ▪ 2
Folder icon

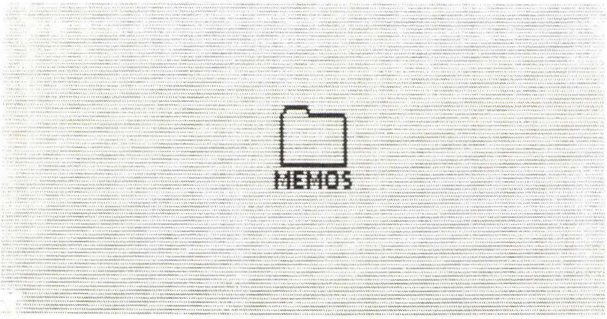


Figure 3 ▪ 3
Folder named "Memos"

Finally, using the mouse, you would drag the three memos inside (Figure 3.4).

This is about as close as you can get to the real world on a computer screen. It's certainly closer than the following, which is the procedure for grouping documents in folders (called *subdirectories*) on the ProDOS User's Disk, an earlier Apple II utilities disk.

1. Go to the Filer menu.
2. Press F for File Commands.
3. Press M for Make Directory.
4. Put in your disk and type the pathname, which consists of a slash, the name of the disk, another slash, and the name of the subdirectory.

In the real world, you don't have to *describe* all the steps of a job in order to get it done. You don't have to with icons, either. You do the job by taking hold of what you want and putting it where you want it to go—that is, by direct manipulation.

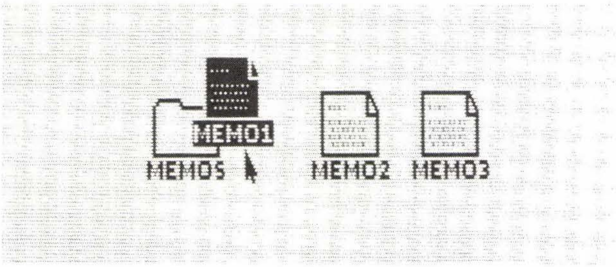


Figure 3 ▪ 4
Document icons being dragged into folder

A future version of the Finder will include a help program to guide you through its essential functions: you'll get to it simply by choosing Help Program from one of the Desktop menus. Once you've mastered the Finder (and it shouldn't take too long), you can remove the help program to secure more space on the disk—or you can leave it there as a quick reference.

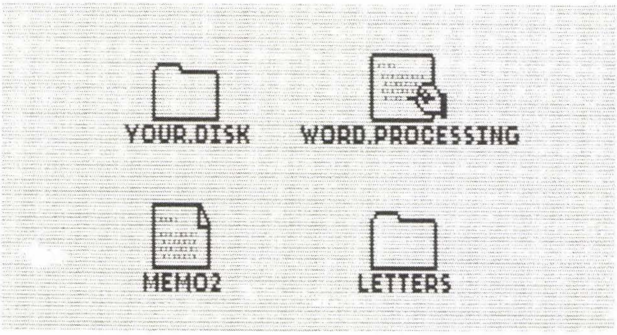
Elements of the Finder

Icons

The Finder presents you with four main kinds of icons: folder icons, application icons, document icons, and disk icons. Figure 3.5 shows you what they look like.

There are variations within each category: application icons may vary according to what kind of application they represent; docu-

Figure 3 ▪ 5
Desktop icons



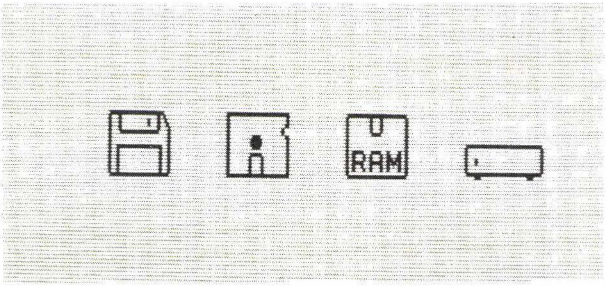


Figure 3 ▪ 6
Icons for 3.5-inch, 5.25-inch, RAM, and hard disks

ment icons usually reflect the application they were created with; and disk icons show you what kind of disk they represent (Figure 3.6). There's also the Trash icon, which looks like a trash can (Figure 3.7).

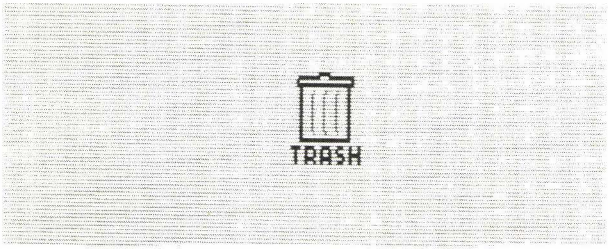
In general, icons have these things in common:

1. You can select them by clicking on them with the pointer, which tells the Apple IIGS that whatever you do next will apply to the icon you've selected (a selected icon is always highlighted).
2. You can name them by selecting them and choosing Rename from the Special Menu.
3. You can move them on the screen by pointing at them, holding down the mouse button, and moving the mouse.

Opening Icons

Disk and folder icons are like closed books, or closed boxes: you can open them to reveal what's inside. An open icon becomes a window (if you've read Chapter 2, you're familiar with windows and how they work), and in the window you see the icon's contents.

Figure 3 ▪ 7
Trash icon



When you open a disk icon, the icons you see in the window represent the documents and applications that are saved on that disk (Figure 3.8). Notice that the name of the disk shows up in the title bar at the top of the window.

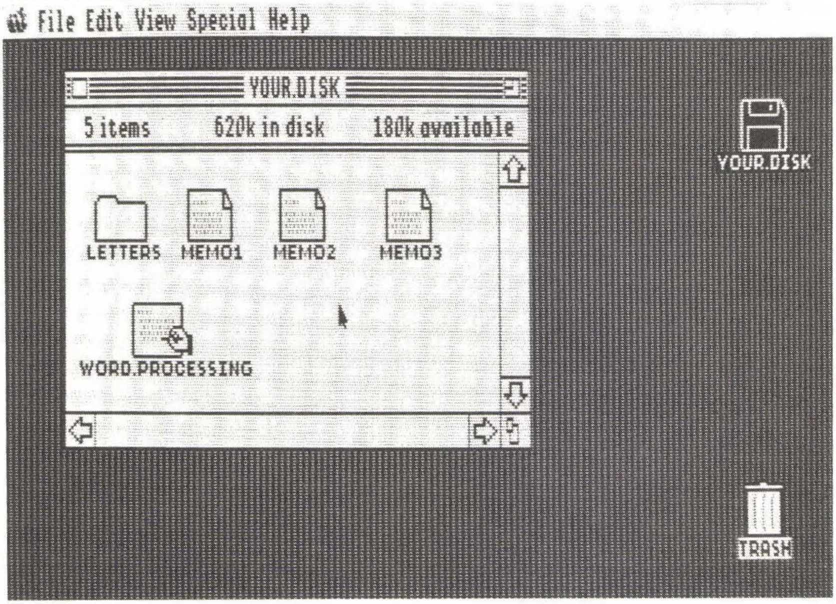
When you open a folder icon, you see the documents and/or applications you've stored in that folder. In this case, the title you've given the folder shows up in the title bar.

You can open an application icon too, but in this case instead of seeing a window with more icons in it, you go to the starting point of that application. Double-clicking on the icon for your word processing program, for example, takes you to the "blank page" on which you'll do your writing (Figure 3.9). Until you save the document you create here, its window is called "Untitled." After that, the title bar shows the title of your document.

Menus

Like every mouse-based application, the Desktop has a menu bar across the top of the screen. On the far left is the apple menu, which holds desk accessories (see Chapter 2); the other menus are File, Edit, View, Special, and Help. The manual you get with your Finder disk explains all the commands on each menu. Here we can give you a

Figure 3 • 8
Disk window


File Edit View Special Help
YOUR.DISK
5 items 620k in disk 180k available
LETTERS
MEMO1
MEMO2
MEMO3
WORD.PROCESSING
YOUR.DISK
TRASH

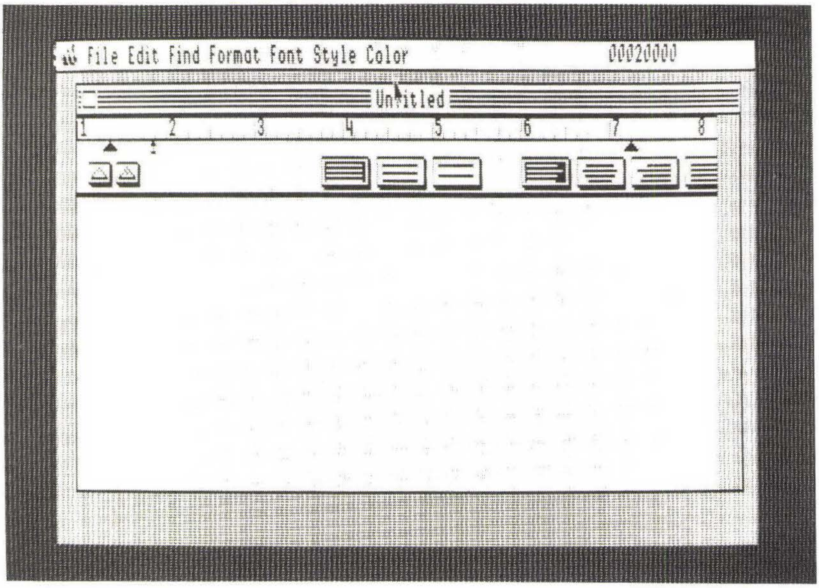


Figure 3 ▪ 9
Starting point of word processing program

better general idea of how things work by showing how you would do several of the most common Desktop tasks and explaining the menus as they come up.

Finder Tasks

Let's suppose you've bought an Apple IIGS word processing program—we'll call it *WriteRight*—that includes the Finder. When you put the disk into the Apple IIGS, turn on the power, and double-click on the disk icon, the Finder "finds" what's on the disk in your disk drive and puts an icon representing its contents on the screen.

Let's suppose you have set yourself a project: you're going to do some research on the bird life in your area, which you'll write up with your word processing program. Although you could save your documents on the application disk (the figure in the upper right corner of the disk window tells you how much memory space is left), you decide to save them on a separate disk where there will be room for all of them together. You put this disk—a brand new disk, just out of the box—into your second disk drive. Before you can save anything on it, however, you have to format it.

Formatting Disks

To format a blank disk, you choose Format from the Special Menu and follow the instructions in the dialog box that appears.

When the formatting is complete, you'll be asked to give the disk a name. These are the rules for naming an Apple IIGS disk:

1. The name must be no more than 15 characters long.
2. It must start with a letter.
3. It can have letters and numbers in it, but no spaces, and no special characters other than periods.

You decide to call your disk Birds. After you type the name, an icon for your new disk appears on the Desktop. You can rename this disk, if you ever want to, just by selecting it and typing a new name.

Launching a Program

Now you're ready to get to work. The first thing you do is launch your word processing program. (To "launch" a program just means to start it up.) To do this, either you select the application icon and choose Open from the File menu, or you simply double-click on the icon. Sometimes there are document-like icons that are actually part of the application. These are files that the program needs in order to operate. If, by mistake, you click on one of them instead of on the program's main icon, it won't matter; the computer will give you a message that says you can't open that file. Look for the icon that has the same name as the application you're using.

A few seconds after you've clicked on that icon, you find yourself in the application. You type up your notes on, say, the Rufous-Sided Towhee, save what you've done, and quit the word processing program. This brings you back to the Desktop. Your new document appears in the window of the disk it was saved on.

Putting Documents into Folders

Two weeks go by, during which you work hard. You've done notes on eight birds and you've written up the notes on five of those birds into informative, entertaining articles. Altogether, you have thirteen documents on your disk.

It's starting to get crowded—time to consolidate. You make Birds the active window (just by clicking in it) and choose New Folder from the File menu. This causes a folder icon to appear, which you name simply by selecting it and typing its name: Notes. Then you drag the appropriate document icons into the folder icon. (Actually you drag the document icons until they're close enough to the folder that the pointer is on the folder icon; when this happens, both the folder icon and the document icon will be highlighted, and you can release the mouse button. The document will "go inside" the folder.)

If you want, you can put folders inside of folders: within the folder called Notes, for example, you could put a separate folder for Woodpeckers, within which you'd have your notes on the Downy, the Hairy, and the Flicker. To work with your documents or move them somewhere else, you just open the folder. It expands into a window showing all its contents.

You decide to draw pictures of the birds to accompany each article. You buy yourself a graphics program, and for a few days you do nothing but draw. Pretty soon you've done eight drawings. You put most of them in a folder you call Drawings. But there's one—the Swamp Sparrow—that you've decided not to include in your collection. You want to throw it away.

Throwing Things Away

Every now and then you'll want to get rid of something—your Swamp Sparrow drawing, an old draft of an article, various extra copies of things that you won't be needing any more. You might also occasionally want to throw an application away, if, for example you have a copy of it on several different disks and you want to make more space on one of them.

To throw something away, you simply drag its icon to the Trash. This permanently erases your file from the disk.

When you're sure you want to get rid of the things you've thrown away, you choose Empty Trash from the Special menu. This permanently erases everything stored in the Trash.

Throwing things away increases the amount of available space on your disk. A figure in the upper right corner of the disk window indicates how much memory space you still have to work with. As you save documents onto your disk, this number goes down, and when it's below 20, you should probably start a new disk, or throw away whatever documents you no longer need.

It takes a while to fill up an 800K disk, though. These disks hold the equivalent of about 400 typed, single-spaced pages. Graphics take up space faster than text, but you could probably get from 50 to 100

of your bird drawings on one disk, depending on how large and complex they were. Disks aren't cheap (a box of ten double-sided disks usually goes for about $35), but they hold a lot, and they can be recycled for a long time before they wear out.

Making Copies of Documents and Disks

So now you have two folders, with several documents in each. You decide you'd better make a backup copy of your folders; if you lost them, or if they somehow got damaged, you'd have to do all your work over again. You put a new disk into your disk drive, format it, and name it Birds.Backup. Now how do you tell the computer you want to copy both your folders from Birds onto Birds.Backup? It's such a natural sort of operation that you can probably guess how to do it: you simply drag the folders from the Birds disk window into the Birds.Backup disk window (or onto the closed icon of Birds.Backup—both methods work the same way). Dragging a document or a folder icon from one disk to another makes a *copy* of that document or that folder (with all its contents) and leaves the original unaffected.

You can also copy the entire contents of one disk onto another disk by moving the first disk icon onto the second disk icon. If you dragged the Birds disk icon onto the Birds.Backup disk icon, you'd copy onto your new disk not only the folders and documents, but the application program as well.

Note that dragging one disk icon onto another doesn't just add the documents on the first disk to those on the second disk. It turns the second disk into a *replica* of the first. If there's anything on the second disk, it will be erased by this operation.

It will be erased, that is, unless the disk you're copying *to* is bigger than the one you're copying *from*. With most utilities, copying a whole disk (a volume copy, as it's called) works only if the disk you're copying and the disk you're copying to are both the same size. You can't copy a 5.25-inch disk onto your hard disk all at once; you have to move the files one at a time. The Finder makes this process easier: if you want to move the contents of a 3.5-inch or 5.25-inch disk onto your hard disk, you can move the icon of the small disk onto the icon of the large disk and the Finder will understand that you're saying, "Add the contents of the small disk to the contents of the large disk," not, "Replace everything on the hard disk with the contents of the small disk." (Of course, if you move the icon of your hard disk onto the icon of a smaller disk, you'll get an error message.)

Duplicating Documents

What if you want two copies of a document on the same disk? You might want to do this so that you could make revisions to one copy and leave the other unchanged. This is called *duplicating*, as opposed to copying. You do it by selecting the icon of the document you want to duplicate—let's say it's called Chickadee—and then choosing Duplicate from the File menu; this causes a new icon, called Chickadee1, to pop up in the disk window. Chickadee1 is exactly the same as Chickadee, except for its name.

Moving Documents to Another Disk

You're reorganizing again. This time you want to move something—not copy it—from one disk to another. Say you've decided to put your Notes and Drawings folders on two separate disks. There are two ways to get the Drawings folder off the Birds disk and onto a new disk:

1. Drag the folder to the window of the new disk (this copies it), and then drag the original of the folder to the Trash.

2. Hold down the Option key while you drag the folder to the window of the new disk. This moves it without leaving a copy behind.

Organizing the Desktop

It's possible to get to the point where you have a great many windows open on the Desktop at once. Even if you have only two disk drives, you can have more than two disk windows showing, because the icon for a disk that you've ejected remains on the screen, all its contents showing as usual in the disk window. (The icons on an ejected disk are displayed in grayish lines, however, to let you know that, although you can move them around, you can't actually do anything that will affect the disk—like change a document's name—until you put the disk back in.)

Therefore you can have several disk windows, overlaid by innumerable folder windows, all open at once. This is where the ability to move windows around comes in handy. You can arrange your windows on the screen so that corners of them are always available for clicking on (clicking makes a window active, bringing it to the foreground). When you're doing things that require you to see the contents of two disk windows at once, you can resize windows so that they're both visible. It becomes just like shuffling real pieces of paper, only on this Desktop it's not as likely that anything will fall on the floor or get thrown away by accident.

There are other ways, too, to arrange your Desktop to your liking. You can simply drag the icons around until they're in the positions you want them in. You can drag more than one at a time, too, by selecting several and moving them as a unit. Do this by drawing a selection rectangle around the icons you want to select: position the pointer on a blank area of the window, above and to the left of the documents, hold the mouse button down, and drag down and right. All the icons within the rectangle will be highlighted, and if you move one, they'll all move.

You can also select several documents at a time just by holding down the Shift key while you click on them. This is handy for situations where the icons are widely separated in a disk window, hard to group comfortably within a rectangle.

If you want, you can opt not to look at the contents of your disks as icons but to list them in words—by name, by date (the Apple IIGS keeps track of the date and time when you created each document), by size, or by type (that is, according to the kind of application they were created with).

All these commands are on the View menu, and you can switch among them rapidly. You'll probably find that you view your documents as icons most often; this is the only view in which you can move them around, and moving them around, as you have seen, is crucial to getting things done.

Desk Accessories

The Apple IIGS comes with two desk accessories: a calculator and a puzzle. You choose them from the Apple menu, which appears in the top left corner of the screen. This menu is present whether you're on the Desktop or within an application, so you can get at your desk accessories whenever youn need them.

If you're writing along about local sightings of the uncommon Blue Gmatcatcher, for example, and you come to a spot where you need to calculate what percentage this year's sightings were of last year's sightings, you can open the Apple menu and choose Calculator from the list of desk accessories. The calculator will appear on the screen, overlaid on your word processing program, and you can figure your percentage, click in the Calculator's close box to put it away, and go on writing.

The puzzle is handy for those times when you need a quick break from the apple menu and a little puzzle will appear on the screen—the kind where you slide the tiles around until you've got them in the right order. It will take you a minute or so to do, and by the time you're finished, your mind will be refreshed, or your caller will be back on the line.

Recycling Disks

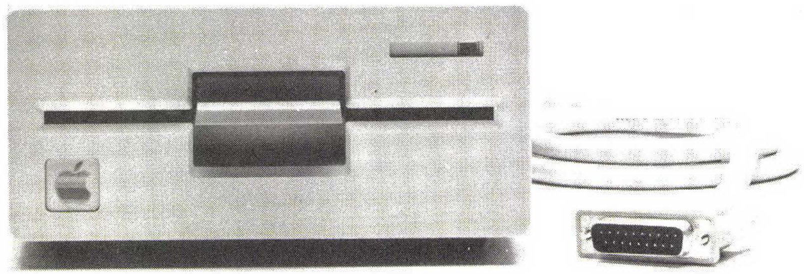
Sometime in the future, when you've finished your bird project, had it published as a book, and don't need all the documents on your disk any more, you can reformat your disk, rename it, and use it again for something else. You do this by choosing Erase Disk from the Special menu. All the documents will be erased, and the disk will be reformatted.

Using the Finder with Old Applications

You may have been an Apple II owner before you got your Apple IIGS. If so, you probably have a number of programs that you want to continue to use, and it's more than likely that they'll run as well as ever—or better—on the Apple IIGS. Fortunately, you don't have to give up the advantages of the Finder to use these older programs.

Using the Finder with 5.25-Inch Disks

When it comes to 5.25-inch disks, the Apple IIGS Finder has a few bumpy spots. This Finder isn't the perfectly consistent, reliably intuitive program that the Macintosh Finder is. With the Macintosh Finder, there's just one way to do things. With the Apple IIGS Finder, it's not that predictable.



The reason, of course, is that the Apple IIGS Finder had to be designed so that it would accommodate both the old and new personalities of the computer. It had to deal with several different types of disks and disk drives, and there was no way to make one method work for everything. If you're using 5.25-inch disks and drives, you'll find that things don't work exactly the way they've been described so far. Here are the things you need to know about using 5.25-inch disks with the Finder:

1. Launching applications

You may be using some ProDOS applications that are on 5.25-inch disks. The Finder won't be on these disks, but you can still use it with the application. Follow this procedure: Put your Finder Disk in your 3.5-inch drive and boot the Apple IIGS. Put your application disk in the 5.25-inch drive and choose Check Drives from the Special menu. When the icon for the application disk appears on the Desktop, open the disk and start the application. (Remember, since the Finder works only with ProDOS, you won't be able to use it with DOS 3.3 or Pascal applications. Lauch these just by putting them in the drive and turning on the Apple IIGS.)

2. Recognizing disks

If you put a 5.25-inch disk in your drive and start up the Apple IIGS, the Finder won't "see" this disk or put an icon for it on the screen. To have the Apple IIGS recognize that you've put a 5.25-inch disk in your drive, you choose Check Drives from the Special menu. This tells the Finder to look in all the drives attached to the computer and record what disks are in them. If at some point you remove a 5.25-inch disk and replace it with another one, the computer has no way of knowing that you've made a change. You need to choose Check Drives again for the new disk to be recognized.

3. Formatting disks

Some of the application programs you have on 5.25-inch disks may use the DOS 3.3 or Pascal operating systems. Instead of using the Finder to format disks to use with these applications, you'll use System Utilities, the utility program on the Apple IIGS System Disk. Double click on the System Utilities icon and follow instructions. (For more about using system utilities, see Chapter 4.)

4. Ejecting disks

You can eject a 3.5-inch disk either by choosing Eject from the File menu or by dragging the disk icon into the Trash, to signify that you're finished working with it. Either of these will cause the disk to be ejected from the disk drive. You can do the same things to the icon of a 5.25-inch disk, and the Finder will remove that disk's icon from the screen. It won't be able to eject it from the drive, however, since 5.25-inch drives have doors that have to be opened manually. Once you've ejected a 5.25-inch disk, even if you don't physically take it out of the disk drive, you have to choose Check Drives from the Special menu before you can work with it again.

The Joy of the Finder

The Finder is a major advance over old-style utilities. Even if you're so used to the various Apple II utilities programs that you could use them in your sleep, you'll soon come to recognize the undeniable advantages of the Finder, which include the following:

1. You don't have to take out whatever disk you're working on and put in a utilities disk in order to copy a document, or rename one, or format a disk.

2. The Finder eliminates the need to keep cataloging your disks in order to remember what documents are on them. Everything on a disk is always visible in the disk window.
3. Seeing what's on your disks in the form of icons instead of a list of file names lets you do your utilities tasks by means of direct manipulation instead of by typing strings of commands and file names or by selecting commands from layers of menus.
4. You can see the documents and applications on more than one disk at once, something that was not possible with old-style utilities.
5. You can organize your documents far more easily with folders than you could with pathnames and subdirectories (see Chapter 4).

In spite of all this, you can do things the old way if you want, or whenever the programs you're using make it necessary. While doing this isn't as simple as working with the Finder, it's not really so difficult, either. And soon it will be even easier. A utility program officially called the Finder (as opposed to the Apple II Desktop), modeled closely on the Macintosh Finder and requiring 512K of memory, will offer expanded functions. It will be available with the next few months watch for it.

Differences between the Apple IIGS Finder and the Macintosh Finder

If you have used the Macintosh Finder, you'll discover that the Apple IIGS Finder doesn't work in exactly the same way. Here's a list of differences:

1. Recognizing disks

The Macintosh can always recognize a disk placed in one of its disk drives when you're on the Desktop. (This is also true within many applications.) It can tell whether or not the disk is formatted for Macintosh, and it will give you a message ("Do you want to initialize?") if it isn't.

The Apple IIGS will automatically find 3.5-inch disks but will not recognize 5.25-inch disks until you choose Check Drives from the Special menu. This causes the Apple IIGS to "look" in the drive and recognize the disk.

2. Formatting

The Macintosh can "see" an unformatted disk whether you put it in the drive when you're on the Desktop or within an application, and when it does it displays a dialog box asking you if you want to initialize it. You don't have to decide what operating system to use, since all Macintosh programs use the same operating system.

The Apple IIGS can "see" 3.5-inch disks, and it will give you a Macintosh-like message if you insert an unformatted 3.5-inch disk in one of your drives while you're within the Finder application. Whether or not you can format a disk from within an application, however, is up to the application. The Apple IIGS *can't* see unformatted 5.25-inch disks without being told to look for them. When you put in such a disk, you choose Format from the Special menu, which brings up a dialog box asking you what drive your disk is in.

3. Opening documents

From the Macintosh Desktop, you can double-click on a document's icon to open that document (and along with it the application it was created with).

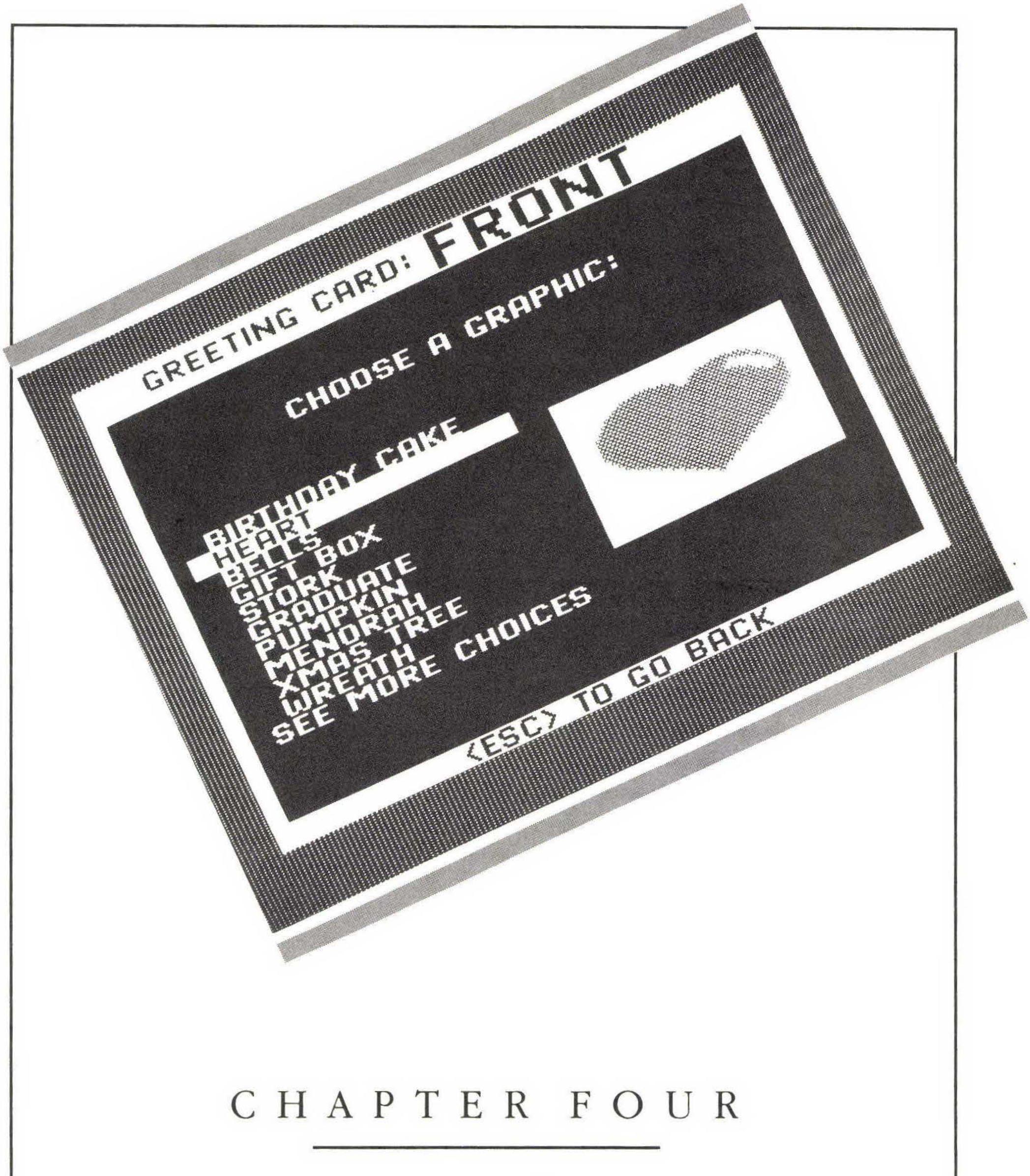
From the Apple IIGS Desktop, you must first open the application. Then you open the document from within it. (There are a few exceptions to this. See the manual that came with your computer, the *Apple IIGS System Disk Guide.)*

4. Throwing documents away

When you put a document in the Macintosh Trash, it stays there until one of four things happens: (1) you open an application; (2) you turn the power off and restart the computer; (3) you choose Empty Trash from the Special menu; (4) the computer needs more memory space to work with, in which case it empties the Trash automatically.

When you put a document in the Apple IIGS Trash, it is erased from the disk instantly.

At press time, the Apple IIGS mouse utilities were still under development. They were being modeled as closely as possible on the Macintosh Finder. Some changes have been necessary, however, so the final product may differ in some respect from the program described in this chapter.



GREETING CARD: FRONT

CHOOSE A GRAPHIC:

BIRTHDAY CAKE
HEART
BELLS
GIFT BOX
STORK
GRADUATE
PUMPKIN
MENORAH
XMAS TREE
WREATH
SEE MORE CHOICES

<ESC> TO GO BACK

CHAPTER FOUR

Classic Applications

Apple II programs that were developed before the introduction of the Apple IIGS won't use the mouse interface or take advantage of the Apple IIGS's super high-resolution graphics, phenomenal sound, and greater memory size, but they will run. That's great news to accountants who swear by the *BPI General Accounting Package*, to teachers whose whole curriculum is designed around *AppleWorks*, and to addicts of *Flight Simulator II*, or some other computer game, who want to get an Apple IIGS but don't want to give up the Apple II programs they love. It's also great news to the developers of these and other Apple II classics.

The makers of Coca-Cola found out the hard way that you can't tell people that a new formula is better than the original. Apple President John Sculley, who learned consumer marketing as president of Pepsi-Cola, didn't even try. Apple bent over backwards to make sure the classic applications would work on the new machine so people could choose the new or the old according to their own needs and criteria.

"Compatibility was our number one goal," says Harvey Lehtman, system software manager for the Apple IIGS project. "There was no point in making an Apple II if it didn't run Apple II software. Whenever we reached a decision point in the design or implementation of the Apple IIGS and the decision was between a new feature and compatibility, compatibility generally won out."

Making a Good Thing Better

Most Apple II developers won't be satisfied with mere compatibility. They will adapt their Apple II classics to take advantage of the Apple IIGS's new features. By switching from 5.25-inch disks, which were the standard storage medium for Apple IIe's and Apple IIc's, to 3.5-inch disks with six times the storage capacity, they can add modules that enable their programs to take advantage of the better sound, graphics, and additional memory on the Apple IIGS without interfering with the way the same program works on an Apple IIe or an Apple IIc.

AppleWorks, the best-selling integrated application program, written by Rupert Lissner, has already been upgraded to use the Apple IIGS's greater memory size. *AppleWorks* was unique among Apple II applications to begin with in letting you have more than one document in memory at one time. In the upgraded version you're still "limited" to 12 documents in memory, but the documents can be much larger.

Marketeer Lee Collings, who was in on the original planning sessions for the Apple IIGS, predicts that many classic applications will be upgraded to take advantage of the Apple IIGS's new sound. "Most programs have their sound routines in separate modules anyway," says Collings. "In *One-on-One* (the basketball game from Electronic Arts where you can choose to be either Dr. J or Larry Bird in a face-off between the two) there's probably one subroutine for crowd noise, another for sneaker sounds. They can write new modules for the Apple IIGS so that when the program comes to one of those routines it branches one way if you're running it on an Apple IIGS and another way if you're running it on a IIe or IIc. You'll see the same sort of thing with graphics," says Collings. "Especially since the new graphics mode is so much easier to use than the old graphics—and it comes out a lot prettier."

New Features for Old Applications

You can expect, then, to see a lot of Apple II classics get facelifts in honor of the Apple IIGS. Even the applications that aren't enhanced to take advantage of the Apple IIGS will derive some benefits from running on the new machine, however.

Speed

The old applications will run faster on the Apple IIGS, thanks to the faster processor and a little trick called *shadowing* devised by hardware engineer Rob Moore. When you start up a program on the Apple IIGS, it gets loaded into "fast RAM," memory that is accessed at the 816's regular 2.8 megahertz (MHz) clock rate. If it's a program designed for earlier models of the Apple II, certain parts are also copied into "slow RAM," a 128K section of memory that is designed to be accessed at 1 MHz, the processing speed of the 6502 processor used in earlier models of the Apple II. The only parts of the program that are shadowed into slow RAM are those that are speed dependent: the parts of the program that send information to the screen and exchange information with peripherals. The parts that aren't speed dependent are accessed from fast RAM. The result is that the whole program runs about three times faster than it would on an Apple IIe.

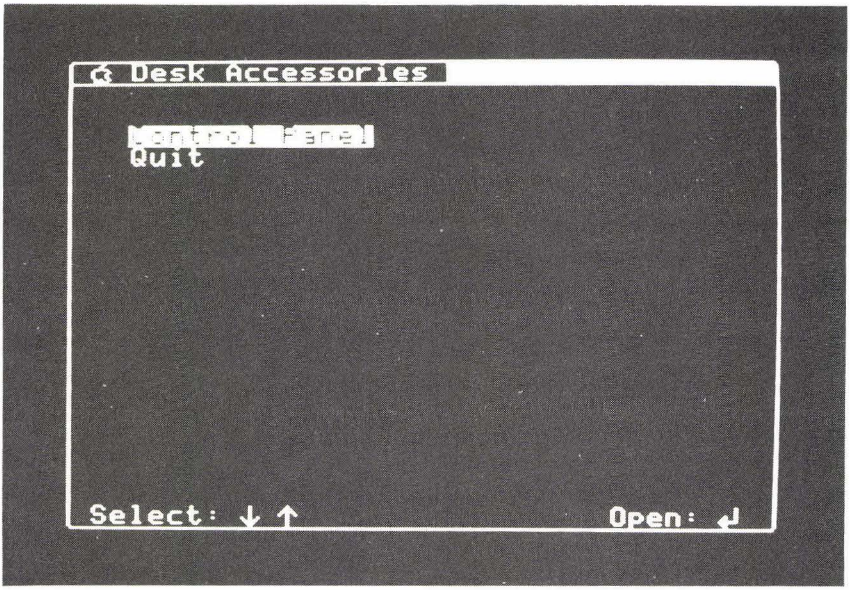
This maneuver goes above and beyond the call of compatibility. Compatibility could have been achieved by moving the whole program into slow RAM and running it from there. Moore's selective shadowing scheme allows old programs to get the benefit of faster processing without compromising other features of the program. If a particular program absolutely has to run at the old speed, users have the option of setting the Apple IIGS speed to 1 MHz using the Control Panel explained in Chapter 6.

Access to Desk Accessories

Another Apple IIGS extra is the ability to access desk accessories such as a calculator and a notepad without leaving your main application. Even if you haven't launched your application by using the Finder (see Chapter 3), you can still get to the Desk Accessories Menu by pressing Apple-Control-Esc. (See Figure 4.1.)

You may have seen desk accessories like Jeeves and Pinpoint advertised for the Apple II before the Apple IIGS came along, but those

***Figure* 4 • 1**
Desk Accessory Menu



could be accessed only from *AppleWorks*. The Apple IIGS desk accessories can be accessed no matter what application you're using.

New Look and Feel

Another Apple IIGS feature that can enhance Apple II classics is the Control Panel program. You can use this program to change the color of the text, background, and border displayed on the screen, to change the repeat speed of the keys on the keyboard, and to do many other things to give old programs a new look and feel. If you liked the old feel, you can leave things set up the way they are on other Apple II's. You can even turn the speed back to Apple IIe speed if you find that the faster speed is making your shoot-em-up games a little too challenging. You'll learn more about the Control Panel program in Chapter 6.

What's So Great About Classics?

Why would you want to use an old program that doesn't even know your mouse exists, when you could be pointing, clicking, and dragging your way through applications that use a consistent user interface?

For one thing, there are about 10,000 classic applications, compared with the few dozen mouse-based, Apple IIGS-specific applications that will be available when the Apple IIGS is first introduced. For a while you might not be able to find a new application that does exactly what you want to do.

For another thing, you might need to use the same application as others in your office or school, and they may have models of the Apple II that don't run the Apple IIGS-specific applications.

You may want to take advantage of the free "public domain" software written by amateur programmers that's available at users groups. You won't find many mouse-based applications among the freebies. If you want to take advantage of the bargains, you'll need to learn how to cope with classic applications that use a less predictable interface.

If you've used other models of the Apple II, you probably have some programs you're comfortable with, and maybe even devoted to. If these programs do everything you want them to do, there's no reason to switch to a new program. More important, if you've been using a program for a long time and have a lot of data you still need to refer to—accounting records or database records—it might not be possible or worth the trouble to convert it for use by another program.

Finally, a lot of people prefer keyboard-controlled programs to mouse-based programs. To them, it's faster to type commands than to use the mouse to choose a command from a pull-down menu because their hands don't have to leave the keyboard.

Choosing Classic Applications

You can find out about new applications for the Apple IIGS by reading *A+, In Cider,* or one of the other monthly computer magazines that review software for the Apple II. The best way to find out about classic applications is to get one of the software catalogs available at computer stores, bookstores, and libraries. One of the best is the *Whole Earth Software Catalog*. The nice thing about this catalog is that the reviews are mostly short, opinionated quotes from experienced users about why they like a particular program. There's no attempt at objectivity, so you feel like you're getting the straight scoop as you would if you asked a friend's opinion. Also they don't attempt to be comprehensive. They review only what they consider the top four or five applications in each category for the top-selling computers. If you're overwhelmed by the number of programs available for the Apple II family of computers, you'll appreciate that someone has separated the wheat from the chaff. If you'd rather read everything about every different word processor, spreadsheet, database, game, or whatever so you can make up your mind for yourself, you're better off getting one of the software catalogs devoted exclusively to the Apple II family, such as the *Addison-Wesley Book of Apple Computer Software.* See the Appendix for information on where you can get these and other resource materials.

One criterion by which software is rated in these magazine reviews and software catalogs is ease of use. This is a major consideration because classic applications don't conform to a standard interface the way mouse-based applications do.

The Classic Interface

The standard interface for classic applications is no standard at all. The only way to learn how to use a classic application is by trial and error, by reading the manual that came with the application, or by calling the friend who recommended it every time you have a question. It's a good thing there are a lot of Apple users, because you can go through a lot of friends this way.

While there are no hard and fast rules about how classic applications work, it's helpful to understand in a general sort of way how keyboard-controlled programs differ from mouse-based applications.

Controlling Programs from the Keyboard

In keyboard-controlled programs, you tell the program what you want to do by typing commands at the keyboard. Typing the letter S, for example, might tell the program to Save your document on a disk. The program knows to interpret your keypress as a command, and not as a letter to be displayed on the screen, if you press the key while holding down one of the command keys: Control, Apple, or Option. (See Figure 4.2.) (On other models of the Apple II, Option is called Solid Apple and Apple is called Open Apple.)

Where possible, keyboard-controlled applications are designed so that the key you press with Control or Apple is the first letter of the thing you want to do. Pressing Control and P, for example, might tell the program to print something; pressing Control and D might tell the program to delete something, and so on. Problems arise when there is more than one function that starts with the same letter. P could stand for Print or Paste.

Using command keys is an effective way to make the keyboard do double-duty as a typewriter and as a program controller, but it has its drawbacks. The obvious drawback is that you have to remember what the various key combinations do or else you have to look in a book every time you want to print, save, or delete something. Many software developers, sensitive to this problem, provide cheat sheets

***Figure* 4 ▪ 2**
Keyboard

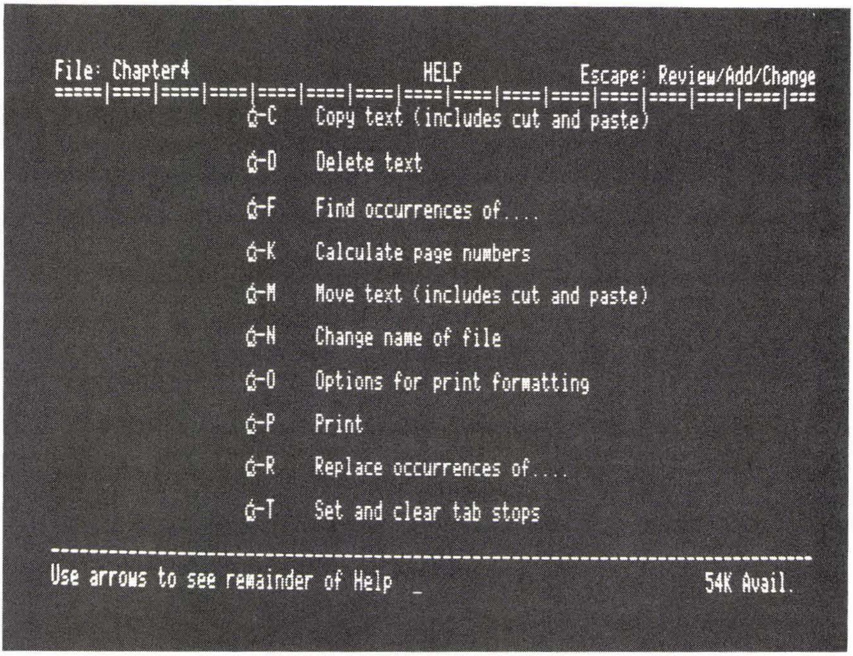
with their manuals—command summaries that you can prop up next to your computer and use for quick reference. The AppleWorks command summary is propped up *inside* the computer—that is, it is *online*. If you can't remember what key to press to underline a word or put it in bold, you can press Apple-? and refresh your memory. (See Figure 4.3.)

Quark, designers of *Word Juggler*, a popular word processor for the Apple II, went so far as to provide a set of keycaps labeled with their *Word Juggler* functions. The manual explains how to replace the keycaps on your computer with the *Word Juggler* keycaps. Quark even provides the keycap extractor. These are the lengths some developers are willing to go to make keyboard-controlled applications a little less confusing.

The trouble with the Quark solution is that different programs assign different functions to the keys and it's not practical to pull out your keycap extractor and rearrange the keyboard every time you change applications.

Another way developers try to minimize the confusion that results when you switch from one application to another is to provide *integrated applications*—several applications that use the same com-

***Figure* 4 ▪ 3**
AppleWorks *Help Screen*



```
File: Chapter4                    HELP           Escape: Review/Add/Change
=====|====|====|====|====|====|====|====|====|====|====|====|====|====|====|===
                  -C   Copy text (includes cut and paste)

                  -D   Delete text

                  -F   Find occurrences of....

                  -K   Calculate page numbers

                  -M   Move text (includes cut and paste)

                  -N   Change name of file

                  -O   Options for print formatting

                  -P   Print

                  -R   Replace occurrences of....

                  -T   Set and clear tab stops

--------------------------------------------------------------------------------
Use arrows to see remainder of Help _                               54K Avail.
```

mands to perform the same functions. By getting a package like this, you don't necessarily get the best applications of their kind, but you can count on all the applications working much the same way, and you get the added benefit of being able to insert part of a document created with one kind of application into a document created with another kind of application.

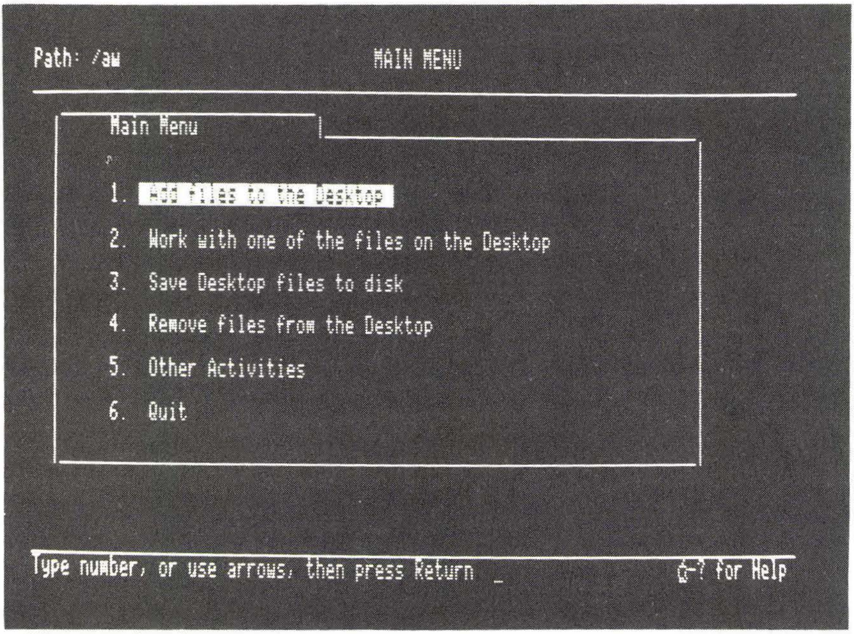
The popularity of *AppleWorks* is partly due to the excellence of the individual applications that make up the package, but it's also largely due to people's desire for a consistent user interface. In 1985 *AppleWorks* outsold all other Apple II applications. In a 1985 reader survey conducted by *A+* magazine, *AppleWorks* was the most popular program in four out of twelve software categories: word processing, information management, spreadsheet, and integrated.

AppleWorks was so popular that it spawned a whole family of *AppleWorks* add-ons: *Moneyworks, ReportWorks, BusinessWorks, MegaWorks,* and many more, using the *AppleWorks* interface. By virtue of the program's popularity, the *AppleWorks* interface is the closest thing there is to a standard interface for keyboard-controlled applications. Understanding how the *AppleWorks* interface works will give you a conceptual understanding of how all keyboard-controlled applications work and how they differ from mouse-based applications.

Using Keyboard Controlled Applications

Keyboard-controlled applications start with a menu that takes up the whole screen. This first menu is called the *main menu* and it shows you all of your top-level options. (See Figure 4.4.) You choose the option you want by typing the number or letter in front of the option and pressing Return. Depending on the complexity of the program, doing this could lead straight into the activity you want to do or to a more specialized menu. In *AppleWorks* it leads to a more specialized menu that is displayed on top of the main menu. You can see just enough of the main menu in the background to remind you where you are in the program. (See Figure 4.5.)

This is the main difference between mouse- and keyboard-controlled applications. In mouse-based applications, all your options are available all the time, thanks to the menu bar at the top of the screen. In keyboard-controlled applications, you follow a certain route through the program. Each menu is like a fork in the road. If you take the left fork, you have one set of choices. If you take the right fork, you have a different set of choices. If you need choices that are down



```
Path: /aw                            MAIN MENU

     Main Menu

     1. Add files to the Desktop

     2. Work with one of the files on the Desktop

     3. Save Desktop files to disk

     4. Remove files from the Desktop

     5. Other Activities

     6. Quit

Type number, or use arrows, then press Return _          ⌘-? for Help
```

***Figure* 4 ▪ 4**
AppleWorks *main menu*

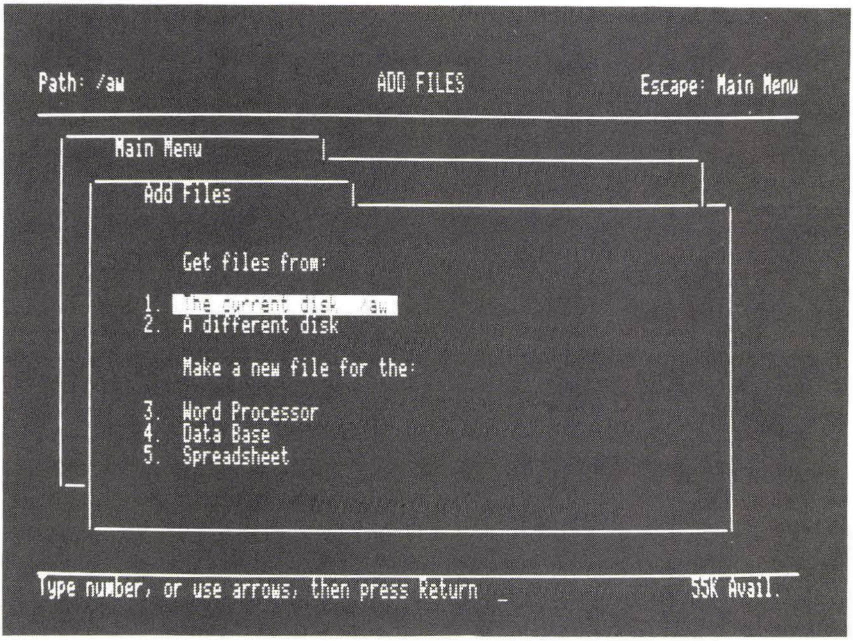
the other road, you have to go back to the crossroads before you can get where you want to go.

In *AppleWorks* and in most other keyboard-controlled programs, you return to a previous menu—the most recent crossroads—by pressing Esc. Return and Esc are two keys you use a lot in keyboard-controlled applications. Return confirms a choice, sends you farther down one of the roads in the program, whereas Esc cancels a choice or takes you back to the most recent crossroads.

Starting Up

In *AppleWorks*, your first choice is whether you want to start a document from scratch or work on a document that's stored on a disk. Actually, you won't see the word "document" in *AppleWorks* or other classic applications. "Document" is a word Macintosh introduced in an effort to humanize the language used in applications. The term is catching on in newer applications, but older applications call them files.

If you choose to start a file from scratch, you proceed to the work area of the program. If you choose to get a file from a disk, *AppleWorks* shows



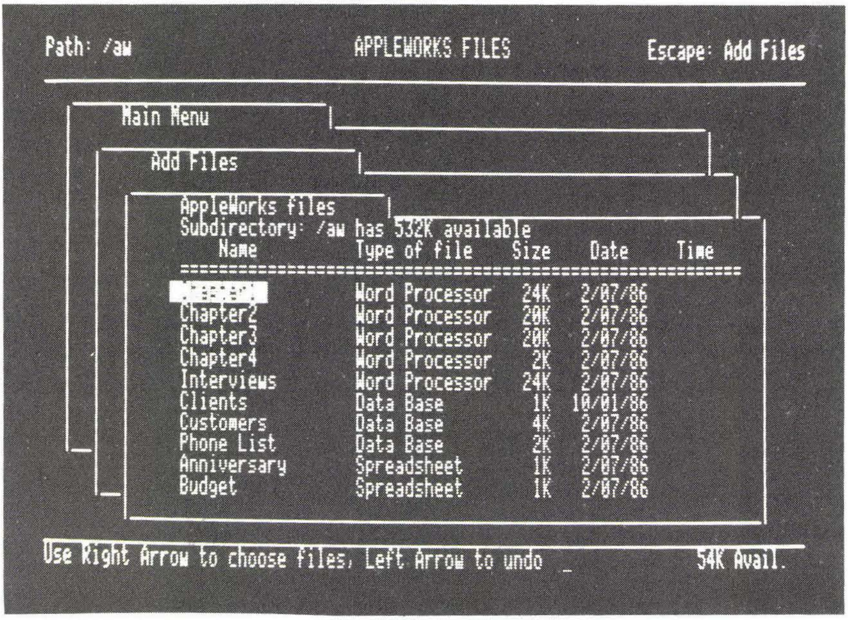
```
Path: /aw                    ADD FILES                 Escape: Main Menu

     Main Menu
        Add Files

             Get files from:

          1. The current disk  /aw
          2. A different disk

             Make a new file for the:

          3. Word Processor
          4. Data Base
          5. Spreadsheet

Type number, or use arrows, then press Return _            55K Avail.
```

***Figure* 4 ▪ 5**
AppleWorks *secondary menu*

you a list of all the files on the disk and lets you mark the ones you want to load into memory. (See Figure 4.6.)

This is a luxury you won't find in many classic applications. Instead of showing you a list of what's on the disk, most classic applications will ask you to type the name of the file you want to load. If you can't remember whether you named your file CHAP1 or CHAPTER1 or INTRO or FIRST, you have to try every possibility until you finally get it right. Until you hit on the exact name that you gave the file when you saved it, the computer will just keep displaying FILE NOT FOUND. It won't even tell you when you're "cold" and when you're getting "warmer." If you don't like the trial-and-error approach, you can use the "list directory" command to see what's on a given disk. Usually there's a command like this built somewhere into the application. If there isn't, you have to quit the application and use the list directory command on a utility disk to find out what your file is called. That will teach you to be creative with your file names.

AppleWorks, on the other hand, makes selection easy by displaying everything that's on the disk. You just point to what you want to work on and it's loaded into the memory of the computer and displayed on the screen.



```
Path: /aw                      APPLEWORKS FILES                  Escape: Add Files

    Main Menu
      Add Files
        AppleWorks files
          Subdirectory: /aw has 532K available
               Name          Type of file     Size     Date      Time
          ==========================================================
          [illegible]        Word Processor    24K    2/07/86
          Chapter2           Word Processor    20K    2/07/86
          Chapter3           Word Processor    20K    2/07/86
          Chapter4           Word Processor     2K    2/07/86
          Interviews         Word Processor    24K    2/07/86
          Clients            Data Base          1K   10/01/86
          Customers          Data Base          4K    2/07/86
          Phone List         Data Base          2K    2/07/86
          Anniversary        Spreadsheet        1K    2/07/86
          Budget             Spreadsheet        1K    2/07/86

Use Right Arrow to choose files, Left Arrow to undo _                  54K Avail.
```

***Figure* 4 ▪ 6**
Directory

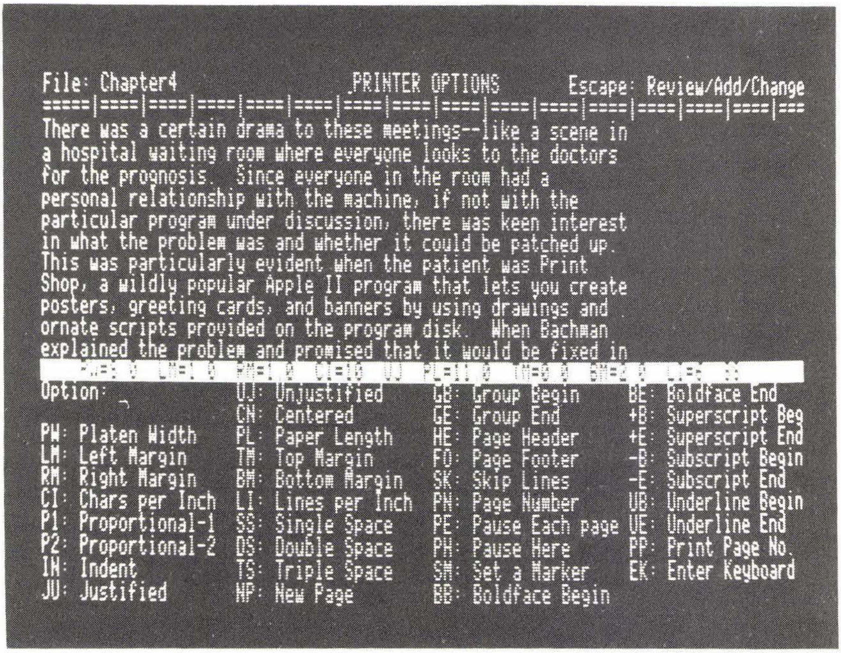
Whether you choose to start a file from scratch or work on one that is stored on a disk, you leave the world of menus and move into the working area of the application. If it's a word processing application, the work area is a blank screen like a page in a typewriter. If it's a spreadsheet, it's a screen divided into rows and columns. Regardless of what application it is, there's a blinking box or blinking line that shows you where you are. Anything you type will appear in that location.

Mouse-based applications call this the *insertion point* because it's the point where the text you type will be inserted. In classic applications, it's called a *cursor.* As you type, the cursor moves to the right to show you where you are. If you make a typing mistake, you can erase what you just typed by pressing Delete, just as you can in mouse-based applications. But what if you discover that you've made a mistake in an earlier paragraph? Moving the mouse will have no effect in a keyboard-controlled application. You can drag that mouse from one corner of your desk to the other and the cursor won't budge. Let go of your mouse and find the arrow keys on your keyboard—that's how you move the cursor in keyboard-controlled applications.

Pressing the Up Arrow moves the cursor one line up, pressing the Down Arrow moves the cursor one line down, pressing the Left Arrow moves the cursor one character to the left, and pressing the Right Arrow moves the cursor one character to the right. If you hold down one of the arrow keys instead of pressing and releasing it, the cursor keeps moving in the direction shown on the arrow.

Using the arrow keys is straightforward. The label on the key gives you a good idea of what the cursor is going to do. For longer documents, though, you'll wish you had a scroll bar as you do in mouse-based applications—some way to leap to the beginning or the end of your document. This is where control keys come in handy. *AppleWorks'* answer to the scroll bar is to let you press one of the number keys in combination with the Apple key to jump to different parts of your document. Pressing Apple-1 moves the cursor to the beginning of your document, and pressing Apple-9 moves the cursor to the end. Pressing Apple-5 moves the cursor to the middle, and you can figure out the rest for yourself.

***Figure* 4 • 7**
AppleWorks *options*



```
File: Chapter4                PRINTER OPTIONS        Escape: Review/Add/Change
=====|====|====|====|====|====|====|====|====|====|====|====|====|====|====|===
There was a certain drama to these meetings--like a scene in
a hospital waiting room where everyone looks to the doctors
for the prognosis.  Since everyone in the room had a
personal relationship with the machine, if not with the
particular program under discussion, there was keen interest
in what the problem was and whether it could be patched up.
This was particularly evident when the patient was Print
Shop, a wildly popular Apple II program that lets you create
posters, greeting cards, and banners by using drawings and
ornate scripts provided on the program disk.  When Bachman
explained the problem and promised that it would be fixed in
   PW=8.0  LM=1.0  RM=1.0  CI=10  UJ  PL=11.0  TM=0.0  BM=2.0  LI=6  SS
Option:          UJ: Unjustified       GB: Group Begin       BE: Boldface End
                 CN: Centered          GE: Group End         +B: Superscript Beg
PW: Platen Width      PL: Paper Length      HE: Page Header       +E: Superscript End
LM: Left Margin       TM: Top Margin        FO: Page Footer       -B: Subscript Begin
RM: Right Margin      BM: Bottom Margin     SK: Skip Lines        -E: Subscript End
CI: Chars per Inch    LI: Lines per Inch    PN: Page Number       UB: Underline Begin
P1: Proportional-1    SS: Single Space      PE: Pause Each page   UE: Underline End
P2: Proportional-2    DS: Double Space      PH: Pause Here        PP: Print Page No.
IN: Indent            TS: Triple Space      SM: Set a Marker      EK: Enter Keyboard
JU: Justified         NP: New Page          BB: Boldface Begin
```

Special Effects

Suppose you want to center a heading or put a word in italics. In a mouse-based application you would select the words you wanted centered or italicized, you would choose the command that did that, and you would see the effect on your screen.

In classic applications you achieve these special effects by using *control characters*—invisible characters that will tell the printer to turn a particular feature on or off. In *AppleWorks* you insert a control character by putting the cursor where you want something to happen, pressing Apple-O (for options), then choosing the option you want from the choices displayed on the screen. (See Figure 4.7.) Typing DS inserts a control character that will tell your printer to turn on double spacing. Typing SS inserts another control character that tells the printer to go back to single spacing. As you can see, there are all sorts of things you can do by using control characters. The frustrating thing is that you can't see the effect until you print your document.

Saving a File

Before you can save a file on a blank disk, you have to format the disk, a process that divides the disk into numbered sections where information can be stored and relocated systematically. With *AppleWorks* there is a formatting option right on the program disk. Even if you discover halfway through creating a document that you haven't formatted a disk to save it on, you can press Esc to get back out to the main menu, choose the Special Activities option, choose the Format option, format your disk, and your document will still be waiting for you when you get back to it. Then it's just a matter of choosing the Save option from the main menu and telling the program what disk to put it on.

This is unusually considerate. With most programs, if you don't have a formatted disk handy, you have to leave the application, and lose the document you just created, before you can format a disk for saving the document that's no longer in memory to save! At moments like this you're not in a very good mood anyway, and the utility disk wants to know which of three formats to use. This brings up the most challenging aspect of using classic applications—and the reason protective programs like *AppleWorks* put a formatting option on the program disk: the variety of Apple II disk operating systems.

Variety of Disk Operating Systems

A *disk operating system* is a special program on every application disk that loads applications and documents from disks into memory and saves documents to disks.

Applications designed to take advantage of the Apple IIGS's built-in mouse tools and greater memory size all use ProDOS, the latest Apple II operating system. Classic applications use one of three operating systems: DOS 3.3, ProDOS, or Pascal.

DOS 3.3

DOS 3.3 is the final revision of the original operating system developed for the Apple II. It was developed in 1979, at a time when disks had a maximum storage capacity of 143K, and DOS used that 143K very efficiently. It was particularly efficient for 1979, because until that time people had been storing programs and documents on cassette tape, a time-consuming and chancy proposition that required plenty of user participation. With DOS, all of the instructions that told the computer how to load the application were on the disk with the application program. All the user had to do was put the disk in the disk drive, listen to a little whirring and thumping from the disk drive, and wait for the first screen of the application to appear on the screen. (Actually, in those days you weren't waiting for a fancy title screen, you were waiting for a square bracket and a blinking square. When you saw that, you could type RUN LITTLE BRICK OUT, or whatever, and start using the program.)

When DOS was the primary Apple II operating system, a disk called the DOS 3.3 System Master came with every disk drive. On that disk were *utilities*—programs for initializing disks (the DOS term for formatting), copying disks, deleting files, renaming disks and files, and doing all sorts of other chores related to managing and manipulating the information stored on disks. Because every user had access to the programs on the System Master disk, application writers didn't have to use up space on their program disks with initializing and copying routines. They could refer users to the System Master disk. The file management routines didn't fill all the space on the System Master, so Apple threw in a game called Little Brick Out, an arcade-style Ping-Pong game, as well as AppleVision, a demonstration of the Apple II's sound and high-resolution graphics capabilities—a stick figure doing the Texas Two-Step to a rendition of Turkey in the Straw.

ProDOS

DOS 3.3 was great for 5.25-inch disks, but it wasn't designed with larger storage devices in mind, so Apple developed a replacement called ProDOS, short for Professional Disk Operating System. The ProDOS equivalent of the System Master was a disk called the ProDOS User's Disk. It had utility programs similar to those on the System Master plus one important option that wasn't—an option for creating subdirectories.

Creating subdirectories on a disk is like putting file folders in a drawer—it gives you a way of keeping related documents together. Subdirectories aren't really necessary on 5.25-inch disks, because you have only 20 or 30 documents on each disk and it's easy to find what you're looking for even if you can't remember the exact name. With larger-capacity disks it's important to have some way of organizing information. (The folders you learned about in Chapter 3 are graphic representations of subdirectories.)

Apple started promoting ProDOS as the DOS 3.3 replacement in 1982, but developers took their time making the transition, and for a while you got two disks when you bought a disk drive: the DOS 3.3 System Master and the ProDOS User's Disk. You had to know which operating system your applications used so you would know how to initialize or format disks; the two operating systems formatted disks in totally different ways.

Pascal

Meanwhile there was another operating system for Apple II applications, and it was being used in the popular *PFS* series of applications, which included *PFS:File, PFS:Write,* and *PFS:Report*. These programs were, and still are, very popular because they are so well designed and easy to use. The secret to their popularity is they don't overwhelm you with a lot of features and options. They give you the features you need and make it easy for you to use them. You may eventually grow out of a PFS program, but you'll never forget what a nice introduction it was to using a computer. The PFS manuals are much too protective ever to mention the word "Pascal" or "operating system." This approach would be good if you never had to use anything but a *PFS* program.

While programs written in BASIC and assembly language are perfectly happy letting DOS or ProDOS handle their disk chores, programs written in the Pascal programming language need their own operating system. Neither the utilities on the DOS 3.3 System Master nor those on the ProDOS User's Disk work on files created with Pascal programs, so Pascal programs have to include utilities for formatting, copying, and everything else on the program disk.

The Operating System Babel

The assortment of Apple II operating systems creates all sorts of confusion for new users. If your first experience is with a Pascal program like *PFS:File,* you get used to having a formatting option and other utilities right on your application disk. If your next application uses DOS 3.3 or ProDOS, it seems like a terrible nuisance to have to swap disks before you have access to utilities.

If you use a lot of applications, it's conceivable you could have some that use DOS 3.3, some that use ProDOS, and some that use Pascal. If you do, you need to be proficient at using the utilities on the System Master, the utilities on the ProDOS User's Disk, and the utilities on your various Pascal program disks. You have to make a point of keeping track of how all your disks are formatted. You can't save a document created with a DOS 3.3-based program on a Pascal-formatted disk, or documents created with a Pascal-based program on a ProDOS-formatted disk. The only safe thing to do is write the operating system on the label and keep at least one blank disk formatted for each operating system so you're never caught without a formatted disk when you need to save a document.

System Utilities: The Universal Utility Disk

When the Apple IIc was being developed, Sue Espinosa, now manager of the user education department, proposed that the system software group create a "universal utility disk" that could include all of the DOS 3.3, ProDOS, and Pascal utilities on one disk. The Apple IIc was designed with new users in mind, and to Espinosa it didn't make sense to simplify setup if you couldn't simplify formatting and other essential activities at the same time.

The universal utility disk took some doing, but it got done and the idea worked out so well that it was adapted for the Apple IIe. The real name of that disk is System Utilities. Like other applications designed for earlier models of the Apple II, System Utilities can be used on the

Apple IIGS; you may even need to use it if you have applications that use a variety of disk operating systems. The Finder, for all its wonderful features (see Chapter 3) is designed mainly to support ProDOS applications.

System Utilities is on the Apple IIGS System Disk. The *System Disk Guide* explains how to use it.

If the variety of operating systems sounds like a nuisance, you can stick with new applications and classics that use ProDOS. However, if you're willing to master three sets of utilities, which is the only aspect of operating systems that you really need to deal with, you can travel anywhere in Apple II country.

Well, almost anywhere. If you get some of your software from old-timers or from the dark, dank archives of an Apple Users Group that's been around since 1978, you may happen upon programs that use DOS 3.2, the first released version of DOS. DOS 3.2 programs will run on the Apple IIGS, but they need a "jump start."

When disks are initialized, they are divided into numbered tracks and sectors. Tracks are concentric circles like the grooves in a record. Each track is marked off into sections called sectors. DOS 3.2 divides each track into 13 sectors, whereas DOS 3.3 divides each track into 16 sectors. This makes it possible to cram more information onto each disk but makes DOS 3.3 incompatible with DOS 3.2. To get around this compatibility problem, there's a program on the DOS 3.3 System Master that converts DOS 3.2 disks from 13 to 16 sector format. If you don't want to do that, you can run your DOS 3.2 programs by first loading a program on the System Master called START13. The instructions for using START13 are in the *DOS 3.3 Manual* and in the troubleshooting section of the *Apple IIGS Owner's Manual.* They're in the troubleshooting section because if you don't know that a program is DOS 3.2-based and you try to start it up without using START13, you won't get far. You'll either get the message CHECK STARTUP DEVICE or a disk drive that won't stop whirring.

Nobody said it was easy being a time traveler.

In Pursuit of Compatibility

While compatibility with classic applications was a major consideration at every stage of the design process, it wasn't completely achieved on the first internal release of the Apple IIGS ROM's. Like any software, the programs in ROM have to go through an exhaustive debugging process. When the software being debugged is part of a brand-new computer, the programmers aren't the only ones looking for bugs. The system software group has a whole army of testers to help them.

Long before the first Apple IIGS's rolled off the assembly line and found their way onto dealers' shelves, the test group was loading all sorts of Apple II applications into Apple IIGS prototypes and reporting any anomalies to the system software group as "Bug Reports." Each Monday at 1:30 the firmware doctors would explain to a gathering of engineering and marketing managers why, for example, there wasn't enough contrast between the stars and the sky in a particular constellation program, and what they were going to do about it.

Fern Bachman, the spokesman for system software at these gatherings, would generally start by saying, "That's not a bug, that's a feature." In the end, though, there was usually some way to tweak the firmware so the classic, whatever it happened to be, would run on the new machine.

There was a certain drama to these meetings—like a scene in a hospital waiting room where everyone looks to the doctors for the prognosis. Since everyone in the room had a personal relationship with the machine, if not with the particular program under discussion, there was keen interest in what the problem was and whether it could be patched up. This concern was particularly evident when the patient was *Print Shop,* a wildly popular Apple II program that lets the user create posters, greeting cards, and banners by using drawings and ornate scripts provided on the program disk. (See Figure 4.8.) When Bachman explained the problem and promised that it would be fixed in the next ROM release, everyone breathed a sigh of relief. Tester Linda Curry, who was leading the meeting and has tested so many programs that it's hard to imagine her being partial to any one, heard the verdict and abandoned her professional detachment to exclaim, "Thank goodness! *Print Shop* is my favorite." Judging from the number of *Print Shop* banners you see in banks, schools, and libraries, Curry would not have been the only mourner if they hadn't found a cure for *Print Shop.*

Rocky's Boots was another classic program singled out for special attention. *Rocky's Boots* goes far beyond the electronic flashcard type of educational program. It lets youngsters create a little universe by assembling electronic machines that move and beep and generally cause a commotion that they can control—and understand, because they built it. The motivation for learning circuit design and troubleshooting skills comes out of their desire to build something. (See

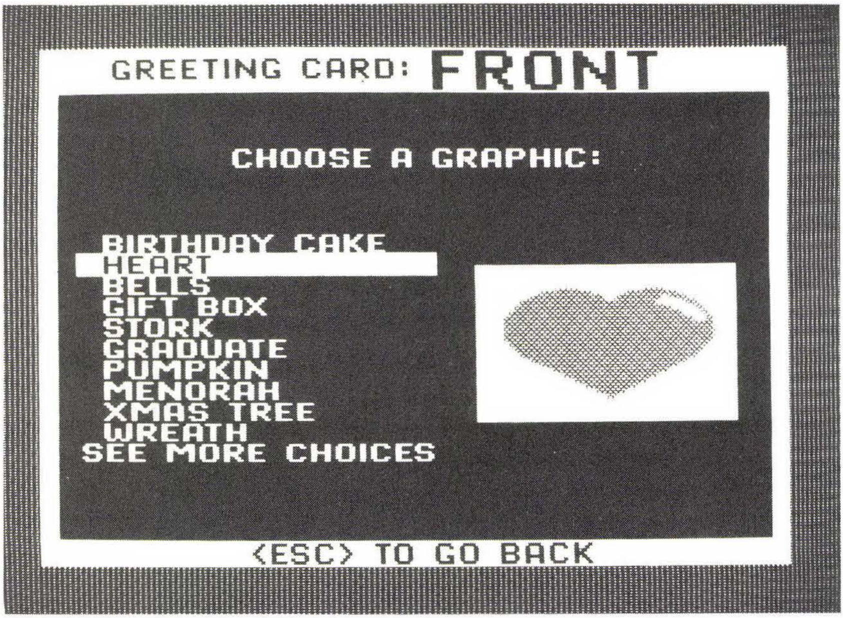
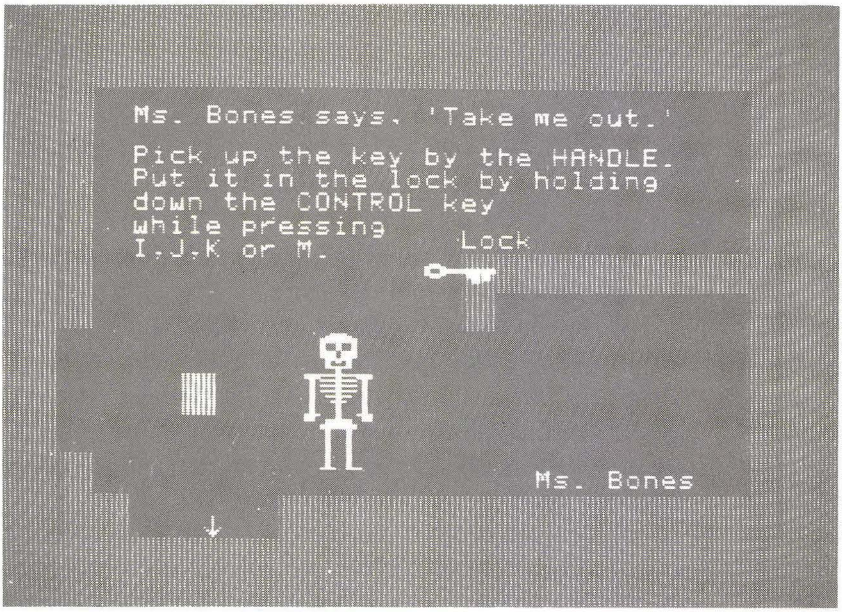


***Figure* 4 ▪ 8**
Print Shop

Figure 4.9.) *Rocky's Boots* is often held up as the model of what a computer can be as an educational tool, so there was concern when it didn't start right up on the new machine.

It didn't take long for Bachman to find the problem. Programs are supposed to access the routines in ROM through clearly marked doors called *firmware entry points.* When firmware gets adapted for a new machine, things may get moved around a lot, but the doors stay in their same locations to preserve software compatibility. But programmers don't always use doors. If they see a way to do something better or faster, they sometimes go in through a window. They create their own entry points and when the firmware changes, those windows disappear and their programs don't run anymore. That's what happened to *Rocky's Boots.*

Normally when a program is caught "breaking and entering" through an illegal entry point, the action item is to "notify the developer" so he can fix the problem and release a new version of the program for the new machine. (If the violation is particularly flagrant, or if it's a chronic offender, the action item is "shoot the developer.") But *Rocky's Boots* is so popular among educators and parents of school age children that Bachman, a father of three himself, made its window a legal entry point.



Ms. Bones says, 'Take me out.'

Pick up the key by the HANDLE.
Put it in the lock by holding
down the CONTROL key
while pressing
I,J,K or M.

Figure **4 ▪ 9**
Rocky's Boots

A similar problem arises when developers try to speed up communication with peripherals (input/output) by talking directly to the hardware on interface cards instead of using the I/O routines in ROM. "It's not like they're being jerks, or anything," says Williams. "They're using the card to do things it was never designed to do. They're doing things where every millisecond counts." Talking to the hardware on the card gave them the speed they wanted, but when the Apple IIGS came along and used different hardware, their programs wouldn't work anymore. This was the case with many communications programs, including Apple's own *Access II.* It talked directly to a chip on the Super Serial Card, the interface card used to connect modems to the Apple IIe. All of the functions of the Super Serial Card are built into the Apple IIGS, so you can attach modems directly to the modem port instead of using an interface card. But the serial interface that's built in uses different hardware from the interface card. The firmware can disguise this difference from programs that use the firmware, but if the program talks directly to the hardware on the card, it won't work with modems connected to the modem port. This isn't a serious compatibility problem, because those programs still work if you connect your modem by using a Super Serial Card. It's just something people have to be aware of.

The reason the built-in serial interface uses different hardware from the serial interface card for the Apple IIe was to enable the Apple IIGS to use the same network used by the Macintosh. A network is a way of linking several computers together in an office or classroom so they can share peripherals and exchange information without sending information through a modem. With the use of a chip called the SCC instead of the ACIA chip on the Super Serial Card, one chip could provide both network support and a serial interface for the Apple IIGS the same way it does in the Macintosh. This is a rare instance where the chance to implement a new feature took precedence over compatibility with Apple II software. With the same network technology in both machines, it will be possible, down the road, to connect Apple II's and Macintoshes to the same network so users can share expensive peripherals like hard disks and laser printers and can exchange electronic mail.

Another compatibility problem arose from the way ProDOS (version 1.1.1) handled interrupts. Interrupts are to the microprocessor what a phone call is to you when you're trying to watch the Super Bowl—an interruption, a distraction from the task at hand. To the processor, an interrupt might be someone pressing a key while it's doing a calculation. ProDOS can enable or disable interrupts. If interrupts are disabled, it's as if the processor has a secretary telling callers that he's in a meeting and can't be disturbed. If interrupts are enabled, the secretary is out to lunch, and anyone can come barging in at any time.

In ProDOS 1.1.1 interrupts are disabled unless the developer decides to enable them. This setup made sense because there was nothing built into the Apple IIe or the Apple IIc that generated interrupts. "Leaving interrupts enabled would have made as much sense as getting an answering machine when you don't have a telephone," Rich Williams explains. The problem with leaving interrupts disabled on the Apple IIGS is that you can't get to your desk accessories. Therefore ProDOS had to be revised so that interrupts would be enabled.

If you have an application with the old version of ProDOS—if nothing happens when you try to access desk accessories—you can take it to your Apple dealer and he will update the disk with the interruptable version of ProDOS, version 1.2.

The only other area of incompatibility results from the way hardware handles shadowing, that otherwise ingenious trick that makes it possible for classic applications to run three times faster on the

Apple IIGS than on the Apple IIe or Apple IIc. Shadowing, remember, involves copying those parts of programs that need to run at slow speed into a special part of memory called "slow RAM" and not copying the parts that can run fast. This solution works perfectly with most classic applications. However, there are some applications that put display information—which is speed dependent—in a part of RAM that isn't shadowed. It doesn't affect many programs, but it affects enough that rather than notify developers, Bachman decided to create a special desk accessory that would do the firmware equivalent of shadowing for that special area of memory. The desk accessory is called Alternate Display Mode. If you run a program that displays a screen full of 2's instead of a sensible opening screen, that's your clue that you need to use Alternate Display Mode. All you have to do is choose Alternate Display Mode from the Desk Accessory Menu and the program will work the way it's supposed to.

If you have any questions about whether a particular program will run on the Apple IIGS, contact your Apple dealer. He has access to the latest compatibility information via AppleLink, a network that links Apple dealers to Apple "central" in Cupertino, California.

Compatible Programs

Here's a list of favorites that passed the compatibility test with flying colors:

Access II (Apple Computer, Inc.)[1]
Alien Action (DLM)
Alligator Alley (DLM)
Alphabet Zoo (Spinnaker Software Corp.)
AppleWorks (Apple Computer, Inc.)
AppleWriter 2.1 (Apple Computer, Inc.)
Apple Logo (Apple Computer, Inc.)
Arcade Machine (Brøderbund)[3]
ASCII Express (United Software Industries)[1]
Backup II (Apple Computer, Inc.)
Bank Street Writer (Brøderbund)[4]
BPI Accounts Payable (BPI Systems)
BPI Accounts Receivable (BPI Systems)
BPI Business Analyst (BPI Systems)
BPI General Accounting (BPI Systems)

BPI Inventory Control (BPI Systems)
BPI Payroll (BPI Systems)
Bumble Games (The Learning Company)
Cavern Creatures (Datamost)
Charles Goren: Learning Bridge Made Easy (CBS Software)
Comp-U-Solve (Educational Activities, Inc.)[2]
Computer SAT Preparation (Harcourt Brace Jovanovich, Inc.)
Crypto Cube (Designware)
DazzleDraw (Brøderbund)—slide show doesn't work
DOS 3.3 System Master (Apple Computer, Inc.)
Dragon Mix (DLM)
Flight Simulator II (SubLOGIC Corp.)[4]
Fortran (Apple Computer, Inc.)
Gertrude's Secrets (The Learning Company)
Instant Pascal (Apple Computer, Inc.)
Jane (Arktronics)[4]
KidWriter (Spinnaker Software Corp.)
Koala Painter (Koala Technologies Corp.)[3]
Magic Spells (The Learning Company)
Mastering the SAT (CBS Software)
Mastertype (Scarborough Systems, Inc.)
Math Blaster! (Davidson and Associates)
Moptown (The Learning Company)
MousePaint (Apple Computer, Inc.)
Music Construction Set (Electronic Arts)
Pascal 1.3 (Apple Computer, Inc.)
PFS:File (Software Publishing Corp.)
PFS:Graph (Software Publishing Corp.)
PFS:Plan (Software Publishing Corp.)
PFS:Write (Software Publishing Corp.)
Pinball Construction Set (Electronic Arts)
Print Shop (Brøderbund)
Projectiles II (Vernier Software)
QuickFile II (Apple Computer, Inc.)
Reader Rabbit (The Learning Company)[3]
Rocky's Boots (The Learning Company)
Speed Reader II (Davidson and Associates)
Spell It! (Davidson and Associates)
Star Search (Earthware)
Stickybear ABC (Field Publications)
Stickybear Numbers (Field Publications)

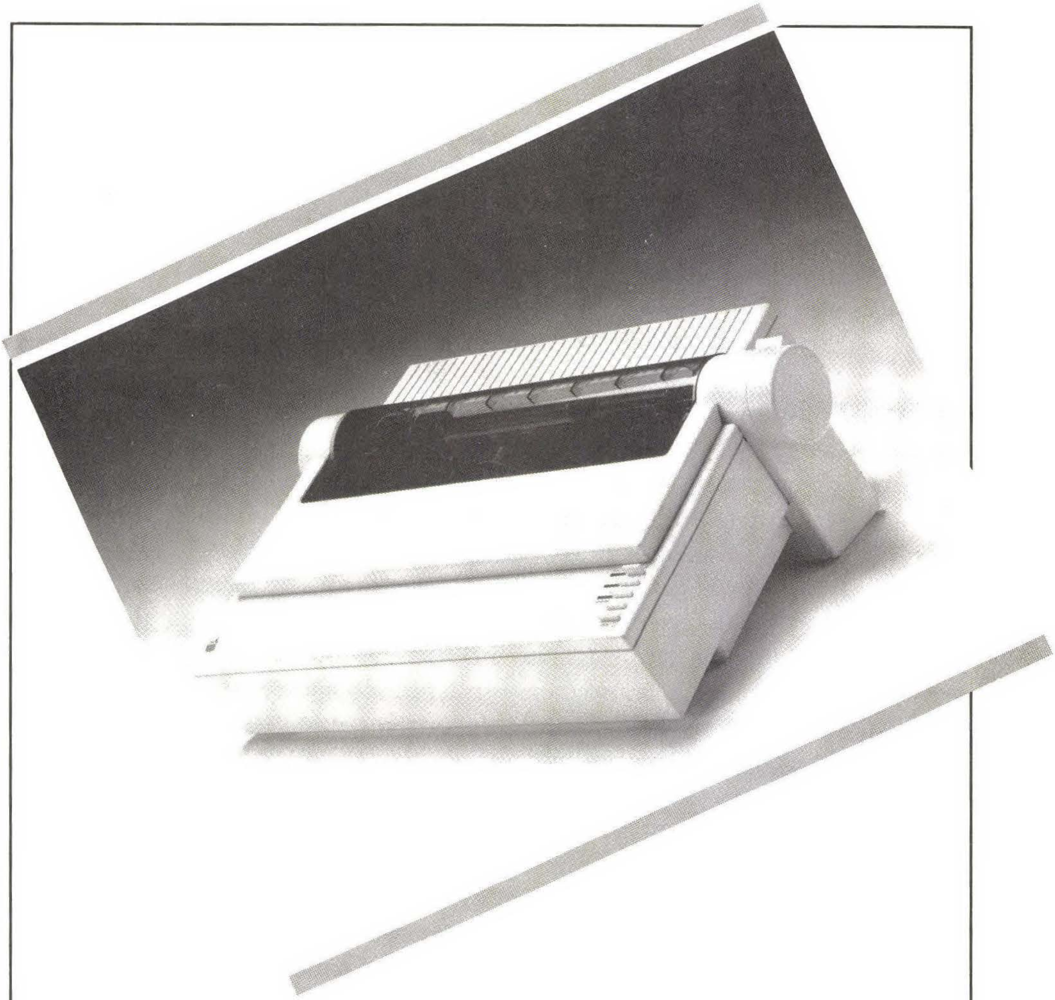
Stickybear Opposites (Field Publications)
Stickybear Reading (Field Publications)
Stickybear Shapes (Field Publications)
Story Tree (Scholastic Software, Inc.)
Super PILOT (Apple Computer, Inc.)
Swyftcard (Information Appliance)
System Utilities 2.1.2 (Apple Computer, Inc.)
Terrapin Logo (Terrapin, Inc.)
Typing Tutor III (Simon & Schuster, Inc.)
Volcanoes (Earthware Computer Services)
Wizard of Words (Advanced Ideas, Inc.)
Word Attack (Davidson and Associates)
Zaxxon (Datasoft)[4]

[1] Modem must be connected to serial interface card, not modem port.
[2] Use Alternate Display Mode on Desk Accessory Menu.
[3] Must set System Speed in Control Panel to Normal.
[4] Set Control Panel to 40 Columns.

Troubleshooting Tips

If a classic application doesn't work, try these possible remedies before you contact your Apple dealer or the manufacturer of the program:

- Change the 40/80 Column setting in the Control Panel.
- Change the System Speed in the Control Panel from Fast to Normal.
- Use Alternate Display Mode, a desk accessory you can choose after pressing Apple-Control-Esc to get to the Control Panel program.



CHAPTER FIVE

Peripherals

Some computers come as part of a computer system—a package that includes a monitor, a disk drive, and sometimes even a modem. With the Apple IIGS, you decide what goes into the package. Your display device can be a $500 RGB color monitor, or the TV that's already in your den. Your storage device can be a $1400 hard disk with a capacity of 20 megabytes, or a $300 3.5-inch drive with a capacity of 800K. Your printer can be a $600 dot matrix printer or a $6000 laser printer. You can go high end, low end, or anywhere in between. You can get everything at once or build your system gradually. You have a lot of choices.

The trouble with having choices is you have to make decisions. First you have to decide what kind of devices to get, then you have to decide which manufacturer to get them from.

You could get all of your standard peripheral devices from Apple. The advantage of doing this is that you can be sure that the devices will automatically work with the Apple IIGS and with software designed for it. It also means that your computer system will be color coordinated. Your monitor, disk drives, and other peripherals will have the same platinum-colored cases as your Apple IIGS. (See Figure 5.1.)

The disadvantage to restricting your choice to Apple peripherals is the same disadvantage inherent in getting integrated applications instead of choosing a word processor, database, and spreadsheet program on their own merits. You could be sacrificing features and economy for ease of use. On the other hand, if you've ever tried to figure out the baud rate, data bit, stop bit, and parity settings for a printer, you might be inclined to sacrifice a lot more than that to get devices that work automatically with your software. Still, it's important to understand that you're making a choice.

Monitors

The first decision you need to make when you're choosing a monitor is whether color is important to you. Color is used to advantage in educational programs, art programs, business graphics, and games. If any of these categories holds special appeal, you should get a color monitor. If you also use business applications like word processing, database, and spreadsheet applications, you should get a color monitor that switches to monochrome or that is capable of displaying text in color.

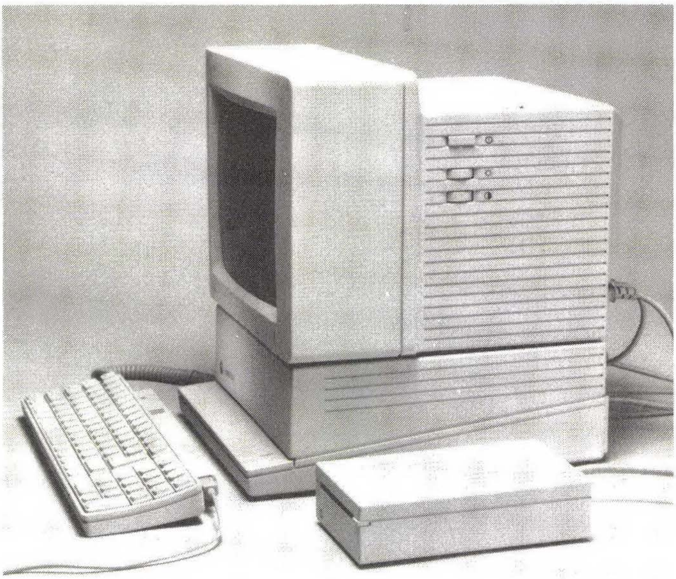


***Figure* 5 ▪ 1**
Apple IIGS and peripherals

If you don't use color applications, you can save money by getting a monochrome monitor.

Apple's monochrome monitor has a white on black display. You can also get monitors that display green on black or amber on black. Some people feel that amber on black causes less eyestrain, or that green on black has more contrast and is more readable. Others prefer black and white because it's more like a typewritten page. Spend some time sitting in front of each kind of display and see which you like best.

Apple has two color monitors: the AppleColor Composite and the AppleColor RGB monitors. (See Figure 5.2.) The composite monitor uses NTSC technology, the standard set by the National Television Standards Committee. The other color monitor uses RGB (Red Green Blue) technology, another way of transmitting a color picture.

The RGB color monitor has sharper resolution and brighter colors than the composite monitor but it's more expensive. With the RGB color monitor, for example, you can display text in color even in 80-column mode. The composite color monitor switches to black and white mode to display text.

If color is important to you, look at a variety of programs using both types of monitors. It may not be important to you to be able to display text in color, but you may decide that the brighter colors are worth the difference in price.

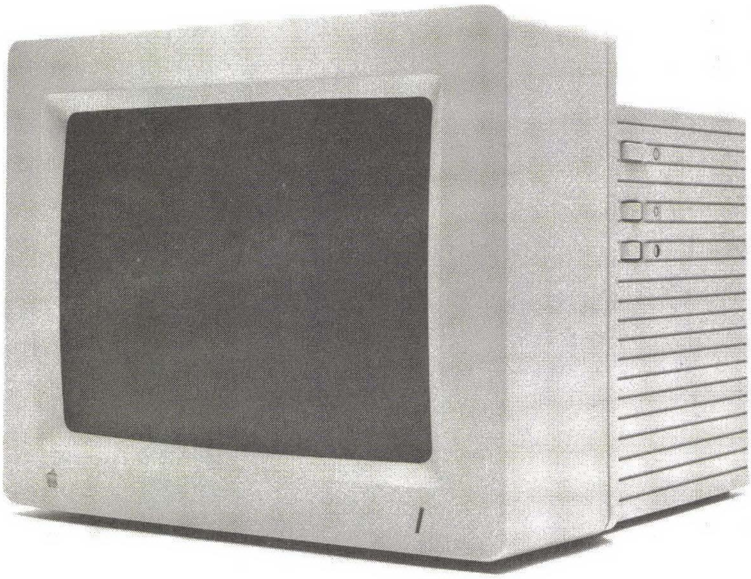


Figure 5 ▪ 2
AppleColor RGB monitor

Using a Television Set as a Monitor

You can also use a television set as a display device. Some new televisions are designed to do double duty as television sets and computer monitors; these can be plugged directly into one of the monitor ports on the computer. Most television sets need to be connected to the computer via a radio frequency (RF) modulator. RF modulators designed for the Apple IIGS have a 15-pin connector that you can plug into the RGB color monitor port.

Unless your television set is designed to be used as a monitor, the resolution won't be as sharp as what you get with a regular monitor and you will be limited to a 40-column display.

Disk Drives

There are three kinds of disk drives you can use with the Apple IIGS: 5.25-inch drives, 3.5-inch drives, and hard disk drives. (See Figures 5.3 and 5.4.)

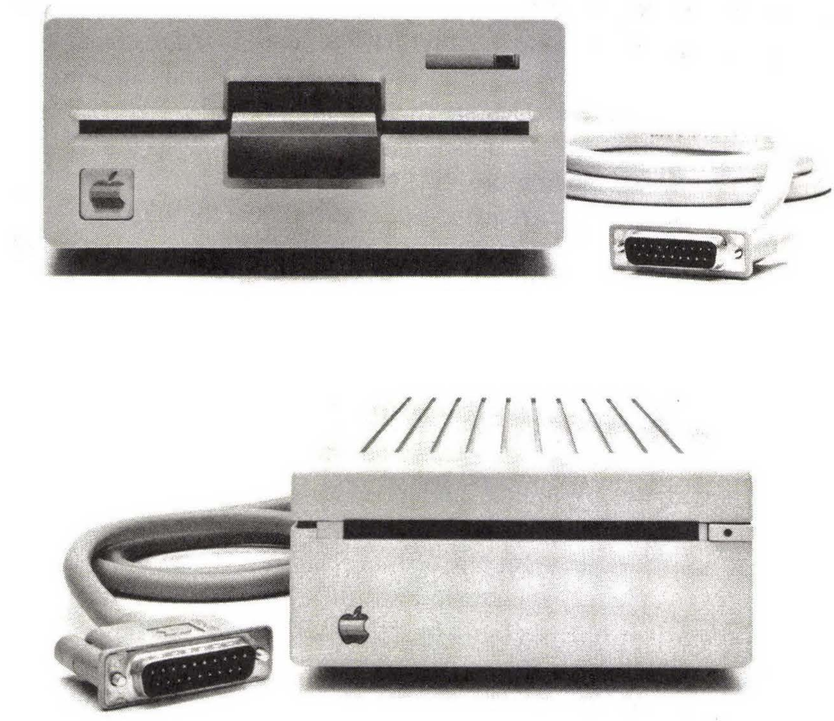


Figure 5 • 3
3.5-inch drive and 5.25-inch drive

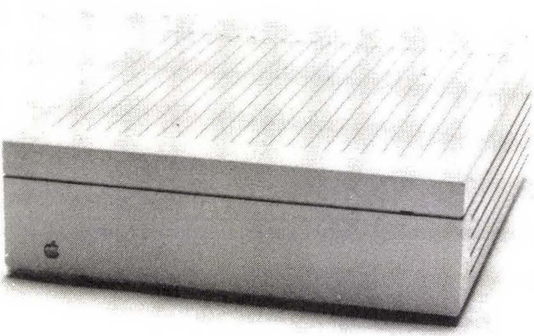


Figure 5 • 4
Hard disk drive

The 5.25-inch drive was the first type available for the Apple II family. It uses 5.25-inch disks, each with a storage capacity of 143K. The 5.25-inch drives were an incredible breakthrough for their time; having one meant you didn't have to load programs from and save documents on cassette tape. Today, however, it makes more sense to get a 3.5-inch drive. Despite their smaller physical size, 3.5-inch disks hold almost 6 times as much data as 5.25-inch disks and the drive doesn't cost that much more.

When 3.5-inch drives became available for the Apple II in 1985, many programs were repackaged to include two versions, one on a 3.5-inch disk and another on a 5.25-inch disk. New Apple II applications designed to take advantage of the Apple IIGS's greater memory size may not fit on 5.25-inch disks—they may be available only on 3.5-inch disks. For example, both the training disk and the System Disk that come with the Apple IIGS are 3.5-inch disks and require a 3.5-inch drive.

If you already have a 5.25-inch drive, there's no reason not to use it with the Apple IIGS, and if you want to swap disks with others who are using only 5.25-inch disks, it makes sense to get one; otherwise it makes more sense to get one or more 3.5-inch drives.

Apple's 20-megabyte hard disk, the HD20, costs much more than a 3.5-inch drive, but it has 25 times the storage capacity, and you don't have to swap 25 disks in and out of a disk drive to get at all that storage area. It's all accessible at one time. If you work with a large number of applications or with large volumes of data, it's probably worth the investment. Another benefit of using a hard disk is faster access time, which is the time it takes the computer to load things from a disk into memory or save things from memory to a disk.

Apple's 20-megabyte hard disk uses the new Small Computer System Interface (SCSI, pronounced "scuzzy") standard. This interface makes it possible to daisy-chain a second hard disk or other SCSI device to your hard disk without using up another slot or buying another interface card.

Memory

While you have the top open to insert the interface card for your hard disk, you'll probably want to add a memory expansion card to the special memory expansion slot. (See Figure 5.5.) Apple's memory expansion card adds up to one megabyte of RAM to the 256K that's built into the computer.

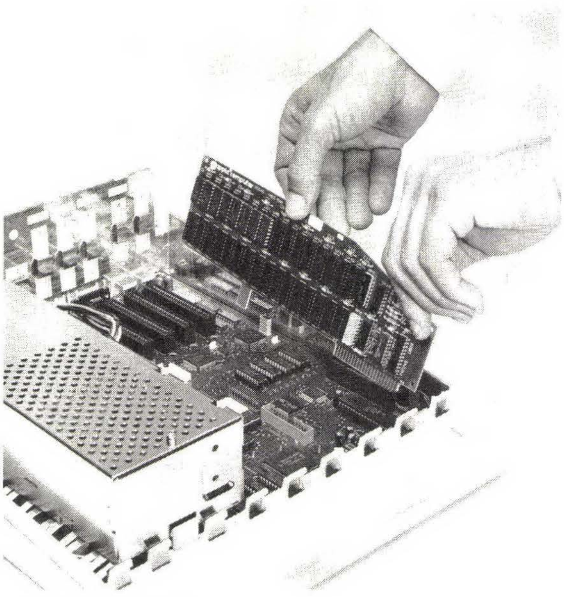


Figure 5 ▪ 5
Memory expansion card

If you want more memory than that, you'll probably be able to find 4-megabyte cards from outside vendors.

The memory expansion slot can also be used to add ROM. Instead of buying application programs on disks, you can get a card that has application programs permanently stored in its ROM. The computer accesses the programs in ROM the same way it accesses programs stored on disks. For this reason, the ROM on a memory expansion card is called a ROM disk. You can use the Control Panel, described later in this chapter, to designate the ROM disk as your startup device. If you do, you'll go straight into one of the applications on the card as soon as you turn on the power switch.

Printers

When you choose the Print command in a program like *AppleWorks,* the program shows you a list of printers and tells you to select yours from the list. Then it tells the program how to package the information it sends to your printer. If your printer is on the list, or compatible with one of the printers on the list, life will be much simpler than if you have to answer a lot of questions about your printer's baud rate, data format, preferred line length, and so on.

The Apple ImageWriter II is the printer most likely to be on the list. If you want to make life easy on yourself, this is the printer for you. It also happens to be a very good printer. (See Figure 5.6.)

The ImageWriter II is a dot matrix printer that can print in color and in any of three modes: Draft, Correspondence, and Near Letter Quality. The mode you select determines how fast the ImageWriter prints. Its top speed, for draft, is 250 characters per second (cps); correspondence speed is 180 cps; its near letter quality speed is 25 cps.

Dot matrix printers form characters by pressing different combinations of pins into a ribbon, thereby producing a pattern of dots—a dot matrix—in the shape of a character on the page. The more dots used to form each character, the less space between the dots and the better the character looks. The ImageWriter II achieves near letter quality—a resemblance to type produced by an electric typewriter—by going over each character several times. That's why it's so much slower than the other two modes. If both speed and letter quality are important to you, you can get a dot matrix printer that uses more pins to form characters rather than more passes over the characters.

You can also get letter quality print by getting a letter quality printer; it forms characters the way a typewriter does, by pounding a plastic

Figure 5 • 6
ImageWriter II

or metal block of type in the shape of a letter against a ribbon. There's no denying the professional appearance of a document produced with a letter quality printer, but the price you pay for the professional appearance is lack of flexibility. You can't use a letter quality printer for graphics or for producing a variety of type sizes and styles the way you can with a dot matrix printer. What you see on the letter quality printer's print wheel is what you get on the page. (The print wheel is usually shaped like a flower, which is why you'll also hear letter quality printers described as daisywheel printers.)

If noise level is an important consideration, there are a couple of other types of printer you should consider: thermal transfer and ink jet. Dot matrix and letter quality printers are sometimes described as *impact printers* because they form characters by pounding something into a ribbon. Doing this makes a noise that can sound like machine gun fire. Thermal transfer and ink jet printers are nonimpact printers. Thermal transfer printers form characters by using patterns of dots, just like dot matrix printers, but instead of pinheads being pounded into a ribbon, the tips of the pins give off heat which liquefies the ink on the ribbon and leaves a dot on the page. Ink jet printers spray dots of ink onto the page for an output that resembles what you get with dot matrix printers. If silent printing is important to you, look into these nonimpact printers.

If you need very high quality printing and money is no object, Apple makes a very quiet printer called the LaserWriter. The LaserWriter uses laser technology to give you documents that look as if they came from a typesetter. Until recently, the LaserWriter was available for use only with the Macintosh, but now you can connect the Apple IIGS to a LaserWriter by using an AppleTalk cable and connector.

Modems

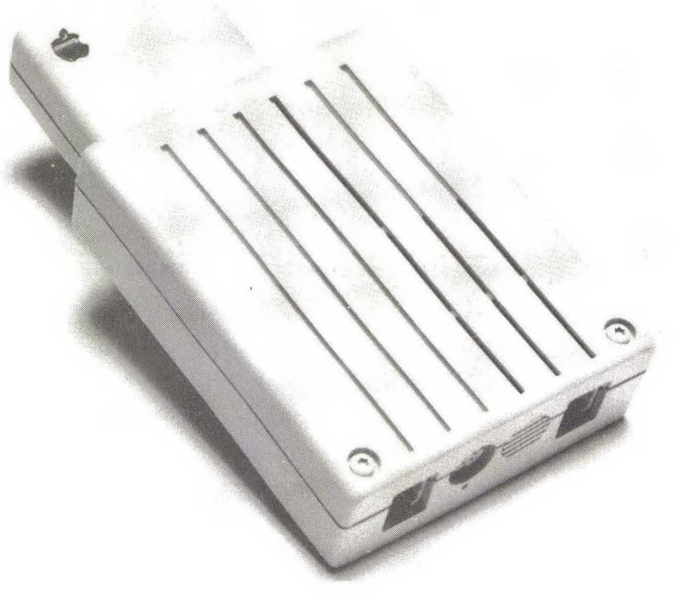
Modems have a reputation for being hard to use. Actually, the problem isn't the hardware, but rather the alien user interface used by many information services, and the difficulty of figuring out how a variety of remote computers are configured so you can have your communications program send information in a form they will be able to decode.

Hayes was the first company to make modems for the Apple II and it is still the last name in modems. It is such a dominant name in the field that other modems are advertised as Hayes-compatible to assure you that they will work with all the communications software designed over the years for Hayes modems.

Different modems can send and receive information at different transmission speeds, called *baud rates,* measured in bits per second (bps). The most common transmission speed is 1200, but older modems have a maximum baud rate of 300 and newer modems can exchange data at 2400. Many modems give you a choice of baud rates. The important thing is that the devices on both ends of the phone line use the same baud rate.

The Apple Personal Modem can operate at 1200 or 300 baud, is Hayes-compatible, and has two handy features called *auto-dial* and *auto-answer*. With auto-dial, you can use the Apple IIGS's built-in clock to tell your communications program to log on to another computer at a particular time of day or night and send a document to that computer without your being there to oversee the operation. With auto-answer, if you leave your computer and modem turned on with a communications program in control, your modem can answer the phone and receive input from a remote computer even when you're not around. These are fairly common features among sophisticated modems. The unique thing about the Apple Personal Modem is that it doesn't sit on the desktop like most other modems. It plugs into a wall outlet. (See Figure 5.7.)

Figure 5 • 7
Apple Personal Modem



Joystick and Hand Controls

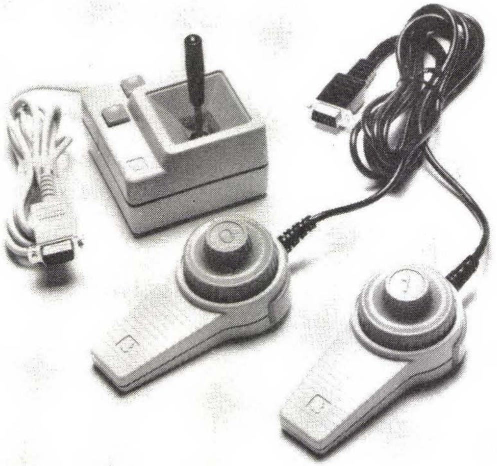
While Apple Computer, Inc. prefers to promote the Apple II as an education machine and a personal productivity tool, it also makes a very good game machine. Some games are designed to be controlled from the keyboard; others can be controlled with the mouse. Some require a joystick or a pair of hand controls. (See Figure 5.8.)

A joystick looks like a miniature version of the joystick in an airplane. You move it forward, back, left, and right to control the action in many game programs. Buttons on the base of the joystick are used to fire torpedos, hit tennis balls, or whatever action needs to occur in the program you're using.

Hand controls are a pair of little boxes with knobs on top and buttons on the side. Turning the knobs on hand controls has the same effect as manipulating the stick of a joystick. The buttons work the same way as the buttons on the base of a joystick.

Serious game players will want to test a variety of models with their most challenging shoot-em-ups. If esthetics are more important to you than the contour of the stick and the position and responsiveness of the buttons, skip the test drive and get a platinum-colored joystick from Apple. Hand controls are available from Apple too, but only in beige, the color of earlier models of the Apple II.

Figure 5 ▪ 8
Joystick and hand controls



Other Devices

In addition to those standard peripherals for personal computers, there are a lot of other things you can connect to the Apple IIGS. The following list will give you some idea of the possibilities.

Adaptive Devices: A computer with open architecture—an accessible main circuit board—is essential to users who need to adapt the computer for special input and output devices. The Apple IIGS is ideal for disabled users because all of the adaptive devices designed for earlier models of the Apple II will work on the Apple IIGS, including braille printers, voice input devices, and speech synthesizers. Contact the National Rehabilitation Information Center (NARIC) for information on other adaptive devices for the Apple II.

NARIC
4407 8th Street NE
Washington, DC 20017
(202) 635-5826

Amplifier: The Apple IIGS can generate sophisticated sound, but hearing it through the tiny speaker that's built into the Apple IIGS isn't nearly as glorious as hearing it broadcast through a larger speaker. You can beef up the sound by getting an amplifier card.

Co-processor: A co-processor is a guest processor. It uses the memory, peripherals, and other features of the host computer to run software designed for computers that use a different kind of processor. The most popular co-processor is the Z80, which was used in the Kaypro II, Radio Shack's TRS-80 Model III, and the Osborne 1. Adding a Z80 card to your computer lets you use software designed for these machines, including *WordStar,* a word processing program that has been around almost as long as personal computers. There was a time only a few years ago when *WordStar* was considered reason enough to get a Z80 card. Today you need a better reason than that. One good reason is that there are thousands of public domain programs that use the Z80's CP/M operating system.

Digitizer: A digitizer is a camera that converts the images it photographs into a digital form you can display on a monitor, save on a disk, and print on a printer.

Drawing Pad: Someone once said that drawing a picture with a mouse is as easy as writing your name with a bar of soap. If you can't get used to drawing with a mouse, you can do your drawing with a pen: all you need is a special drawing pad and software designed to receive input from it. A drawing pad records the movements of a special pen and creates the same image on the screen.

Hard Disk Backup: A hard disk is a fast and convenient way to store large amounts of information, but as with any disk, you should have a spare copy of the information stored on it. Because a hard disk can hold more data than other storage devices, there's that much more to lose if something goes wrong. One backup solution is a second hard disk. If your hard disk is the type that is plugged into a SCSI (Small Computer System Interface) card, you can daisy-chain the second hard disk to the back of the first. That's a high-priced solution, though. You can keep copies of individual documents on 3.5-inch disks, but that can be time-consuming, since it takes 25 3.5-inch disks to store all the information on a full 20-megabyte hard disk. Corvus has a device that makes a backup using a standard VHS or Beta video cassette recorder (VCR). If you already have a VCR, that might be the cheapest solution. If you don't have a VCR, you can get a device called a *streaming tape machine* that backs up the information on a hard disk using cassette tape. While tape recorders aren't practical as primary storage devices because of their sequential data access, they're perfectly good as secondary storage devices and they're cheaper than other backup solutions.

Home Control Devices: By connecting special contraptions to various outlets throughout your house and plugging appliances into those contraptions, you can use the built-in clock in your computer to turn lights on and off at preset but apparently random hours, start the coffee at 7 A.M. on weekday mornings and 11 A.M. on weekends, turn sprinklers on and off, and all sorts of things. These aren't the easiest devices to install and program, but if you like a challenge, it's an interesting way to keep your computer occupied when you're not using it for more mundane tasks.

Light Pen: Some programs let you draw pictures and choose from menus by pointing to options on the screen with a special device called a light pen. This method is even more direct than moving the mouse to move a pointer on the screen. Holding your hand up to the screen to draw can be tiring, however, and there isn't a lot of software that recognizes the light pen as an input device.

Music Keyboard: You can get music composition programs that accept input from a musical keyboard. You press a note on the keyboard and that note appears on the electronic music paper on the screen. It's like having a secretary taking musical dictation.

Network: work is a chain of computers and peripheral devices linked together by cable. You can connect up to 32 computers in an AppleTalk network. When your computer is part of a network, you can send messages to others on the network without using a modem and you can more easily justify the cost of an expensive peripheral device like a laser printer because everyone on the network can share it.

Plotter: A plotter gives you a paper copy of images you create with Logo and other graphics-oriented programs. A plotter uses colored felt-tip pens to put computer pictures on paper. While plotters aren't as versatile as dot matrix printers—you can't use them to print text and spreadsheets—they are very quiet and produce more professional drawings because they really draw the graph, chart, or picture instead of forming it out of dots.

Connecting Peripherals

There are two ways to attach peripherals to the Apple IIGS. You can plug things into ports, or you can plug things into interface cards and plug those interface cards into slots on the main circuit board.

Using the ports is more convenient. Also there's a port designed for every standard device, so if you're the kind of person who gets squeamish at the sight of chips and resistors, you never have to open the Apple IIGS lid. You can plug in your keyboard, your mouse, your monitor, your printer, your modem, and your disk drives as easily as plugging a TV into the power outlet in the den—more easily, in fact, because you don't have to figure out which antenna wires wrap around the VHF screws and which wrap around the UHF screws.

But even if you start out with devices exclusively designed for the ports, it won't be long before you venture under "the hood." If curiosity doesn't get the best of you, the desire for more memory will.

"There's no such thing as too much memory or too much disk storage," says Lee Collings. "If you have a megabyte of memory on your desk, you can have 12 spreadsheets in memory at once—one for every month of the year—plus one big consolidated spreadsheet that incorporates information from all the other spreadsheets. You may not always need that many files open at once, but when you do, it's nice to know you can have them. It's like having 8 cyclinders in a car: you don't use it all the time, but you're glad you have the extra horsepower when you're trying to get around a truck."

Outside Connections

There are eight ports on the back of the Apple IIGS. Most of them are designed to connect a single device like a printer or a modem to the computer. Two of the ports, however, allow for connecting a whole string of devices through one port. You can connect up to four disk drives to the disk drive port and an almost unlimited string of input devices to the Apple Desktop bus, starting with the detached keyboard. Therefore there's almost no limit to the number and variety of things you can attach to your computer even before you open the lid and take advantage of the slots. (See Figure 5.9.)

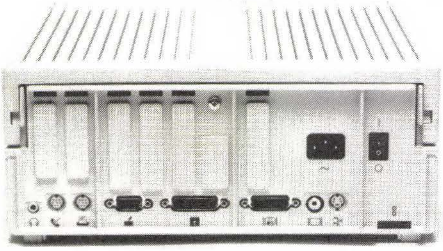
Sound Port

The sound port is for attaching stereo headphones to the Apple IIGS. The sound that comes out of the headphones won't be stereo (there's only one speaker inside) but those are the kind of headphones you should use. Headphones come in handy when you're using your Apple IIGS in a classroom, in an office, or in a shared house where the sounds from your application could disturb the people around you.

You can also use the sound port to connect the Apple IIGS to outside speakers like the ones connected to your stereo system.

If you don't want to use headphones and you're concerned about disturbing people, you can adjust the sound coming out of the speaker by using the Control Panel program described in Chapter 7. You can even turn off the sound completely if you want to.

***Figure* 5 ▪ 9**
Ports



Serial Ports

The ports marked with the printer and modem icons are identical serial ports. *Serial* describes one way that the Apple IIGS can exchange information with peripheral devices. Devices that use a serial interface send information one bit after another along a single wire, in contrast to devices that use a parallel interface, which send the eight bits that form each character along parallel wires.

Although they are identical ports, they are labeled with different icons because one is set up for use with serial printers like the Apple ImageWriter series and the other is configured for use with a modem.

The configuration of a port determines the speed and the form in which information will be sent from the computer to the device. If you are connecting a device that uses a configuration different from the ImageWriter settings or communicating with an information service or remote computer that uses other than the standard information service settings, or if you want to use a printer in the modem port or a modem in the printer port, you can change the configuration of the port to conform to the specifications for your device by using the Control Panel program.

While it's not hard to change port settings by using the Control Panel program, it can be hard to find out what settings a particular device uses. If troubleshooting isn't your idea of a good time, get a device that works automatically with one of the serial ports, or plan on getting help.

Here are the standard settings for the printer and modem ports:

Settings	Printer Port	Modem Port
Device Connected	Printer	Modem
Line Length	Unlimited	Unlimited
Delete First LF After CR	No	No
Add LF After CR	Yes	No
Echo	No	No
Buffering	No	No
Baud	9600	1200
Data/Stop Bits	8/1	8/1
Parity	None	None
DCD handshake	Yes	Yes
DSR/DTR handshake	Yes	Yes
XON/XOFF	No	No

You can also use one of the serial ports to link your computer with others in an AppleTalk network. The advantage of forming a computer network is that you can exchange electronic mail with other computers in an office without using modems and you can share expensive peripherals like laser printers and hard disks. In a classroom network, all the computers can share the disk drives and programs at the teacher's station and the teacher can send electronic coaching to students who need it without neglecting others in the class.

If you use one of the serial ports as an AppleTalk network port, you need to reconfigure the port for that purpose by using the Control Panel program. If a printer is part of your AppleTalk network, you can use your printer port to connect your computer to the network and use your remaining serial port for a modem connector. If you want to use a printer at your desk besides using the one on the network, you can connect your computer to the network by using the modem port. If you want to use a printer and a modem and also connect to an AppleTalk network, you need either to get a *port extender* (a box that lets you plug in two or more devices and choose between them by flipping a switch) or to connect one of your devices to an interface card in a slot.

Game Port

The game port is for connecting a joystick or hand controls (game paddles) to the Apple IIGS. If you have old-fashioned joysticks or hand controls, you can still connect them to the GAME I/O socket on the main circuit board as you could on earlier models of the Apple II and on the Apple IIe.

Though it uses the same type of connector as the mouse port on the Apple IIc, you can't connect a IIc mouse to the Apple IIGS game port.

Disk Drives

You can connect up to four disk drives to the computer by using the disk drive port: two 5.25-inch drives and two 3.5-inch drives. If you have both types of drives, you must connect the 3.5-inch drives closest to the computer in the daisy chain.

When you turn on the power, the computer looks for a program disk in the first 5.25-inch drive in the daisy chain. If you don't have a 5.25-inch drive in the chain, or if your 5.25-inch drive is empty, the computer will check the first 3.5-inch drive in the chain. You can change the startup device by using the Control Panel program.

If you have a 3.5-inch drive designed for the Apple IIe, you can connect it directly to the disk drive port. You don't need the controller card. In fact, connecting a 3.5-inch drive by using a controller card affects the way some programs work, so you should connect 3.5-inch drives to the port.

RGB Color Monitor

RGB stands for Red Green Blue. The significance of the icon over the RGB monitor port, a screen with three patterns, is that RGB monitors send color signals to the screen along three separate wires, a process that keeps the colors purer than the alternative, NTSC technology, in which the three color signals are encoded and compressed in order to be sent on one wire.

There are two ways of transmitting RGB signals: *analog* and *digital*. The analog RGB technology produces a broader range of colors and richer colors than the digital technology, and it's the interface that's built into the Apple IIGS. That's the good news. The bad news is that the standard RGB interface for the Apple IIe was digital RGB. Monitors that used the RGB interface card designed for the Apple IIe's AUX. CONNECTOR slot can't be connected to the RGB port on the Apple IIGS. And you can't use the Apple IIe RGB card, because there is no AUX. CONNECTOR slot on the Apple IIGS's circuit board.

You can also plug an RF modulator into the RGB color monitor port in order to use a television set as a display device. Be sure the RF modulator you get has a 15-pin connector designed for the Apple IIGS.

Monochrome Monitor

The port with the unpatterned monitor screen icon is used for connecting a monochrome monitor or an NTSC color monitor to the Apple IIGS.

Apple Desktop Bus

A bus is a port that can accommodate a whole string of devices. Unless your Apple IIGS is an upgrade from an Apple IIe (in which case your keyboard is built into the case), you'll probably connect your keyboard to the desktop bus and daisy-chain other devices, like the mouse, to the connector on the keyboard.

The detached keyboard and mouse are the only Apple Desktop bus devices currently available from Apple, but inventive developers are bound to think of other input devices to attach to the bus: drawing tablets, light pens, game controllers, and more.

Power

The remaining outlet on the back panel is for the power cord. The only thing you need to know about that is that you won't be able to do much until you plug in a power cord. You should also know that some devices draw their power from the computer (disk drives, for example) while others are designed to be plugged into an electrical outlet (monitors, hard disks, printers). Since there are many devices that need power and few homes that have four or five sockets per outlet, you should plan on getting a power strip along with your computer. A power strip is a device with four or six outlets and a switch that lets you turn all your devices on and off at once.

Security Lock

The opening in the lower right corner as you face the back panel is for an optional security lock. It fits into that opening and anchors the computer to a desk or cart so it can't walk away from your office, home, or classroom.

Inside Connections

If you're new to computers, it may come as a shock that you are encouraged to open the top and poke around inside. It may even seem a little intimidating. But as you watch more experienced users casually popping the top, adding cards, even adding chips to the main board, you realize that there's nothing to lose and much to gain by a more intimate relationship with the slots that will come to symbolize for you what the West symbolized to Daniel Boone: elbow room—possibilities you can't even imagine until you get there. (See Figure 5.10.)

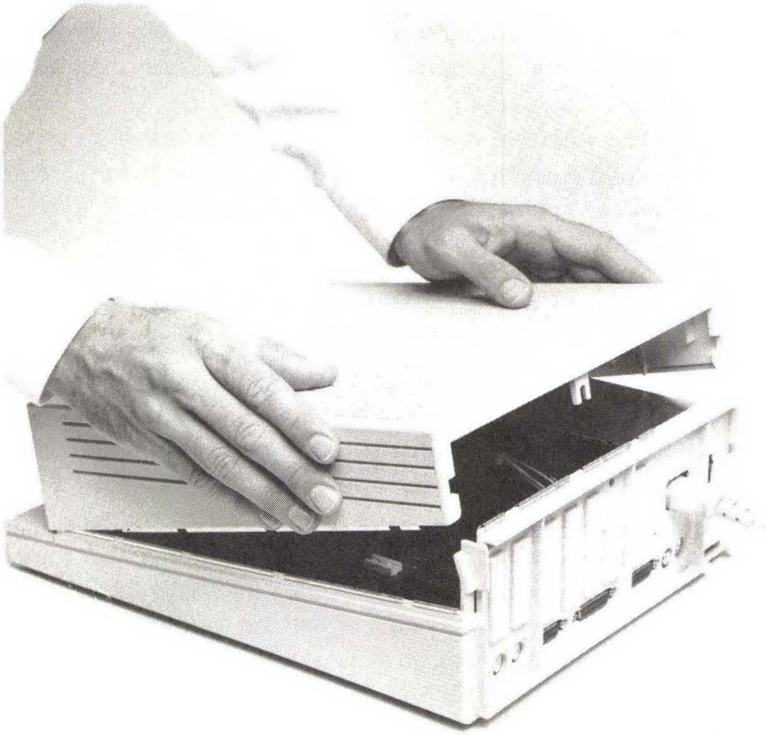


Figure 5 ▪ 10
Popping the top

Westward Ho

Before you start filling your slots with interface cards and heading out for the frontiers of computer science, you need to understand the relationship between slots and ports.

Ports didn't exist on early models of the Apple II, so Apple II software looked for devices connected to slots. When ports were built in, they were designed to act like, or emulate, particular slots containing cards, so that old programs could recognize the devices. Because of this emulation, you can't use a port and its corresponding slot at the same time. You have to activate one or the other by using the Control Panel program. Unless you say otherwise, ports are active. This is the case presumably so beginners who get devices designed for ports won't have to deal with the Control Panel until they're feeling more adventurous. The following chart shows you which port or built-in feature corresponds to which slot.

Printer Port	Card in Slot 1
Modem Port	Card in Slot 2
80 Column Firmware	Card in Slot 3
Mouse Firmware	Card in Slot 4
Disk Drive Port (with 3.5-inch drive)	Card in Slot 5
Disk Drive Port (with 5.25-inch drive)	Card in Slot 6
AppleTalk	Card in Slot 7

You'll learn more about these and other Control Panel settings in Chapter 6.

Peripheral Compatibility Issues

Just as there are classic applications, there are classic peripherals—devices designed for earlier models of the Apple II. You can use most classic peripherals on the Apple IIGS the same way you used them on other models of the Apple II, but there are exceptions.

80-Column Cards

There was a slot in the Apple IIe, called the AUX. CONNECTOR slot, that was designed for different variations of the 80-column card. A regular 80-column card made it possible to display 80 columns instead of the standard 40. An extended 80-column card gave you the 80-column display and 64K of additional memory. The RGB extended 80-column card gave you the 80-column display, 64K of additional memory, and an RGB color monitor interface. There is no AUX. CONNECTOR slot in the Apple IIGS, because the features you got by putting these 80-column cards in the slot are built into the Apple IIGS; therefore you can't use any of these cards.

The built-in features in the Apple IIGS are identical to the features you got by adding an 80-column card, with the exception of the RGB color monitor interface. The Apple IIGS's RGB color monitor support is for analog monitors while the RGB support you got by adding an RGB extended 80-column card was for digital RGB color monitors. If you have an RGB color monitor designed to be used with an RGB extended 80-column card, you won't be able to use it with the Apple IIGS.

Serial Interface Cards

You can connect a serial device to the Apple IIGS by using a serial interface card, but you don't need to. The interface for two serial devices is built into the Apple IIGS, so you can connect one serial device to the printer port and one to the modem port on the back panel. If the serial device was designed to be used with a serial interface card, you'll need to get an adapter cable before you can attach the device to the port, because the connector on the back of the computer is different from the connector on the card. If you already have a serial interface card and you don't want to bother getting an adapter cable, go ahead and connect the serial device to the card and put the card into a slot.

There are cases where programs that work with a device connected to a card won't work with the same device connected to a serial port. *Apple Access II* and *ASCII Express* are two well-known examples. Programs like these, which talk to the hardware on the serial card instead of using the serial interface firmware, won't work unless you connect your modem to a serial interface card.

Disk Controller Cards

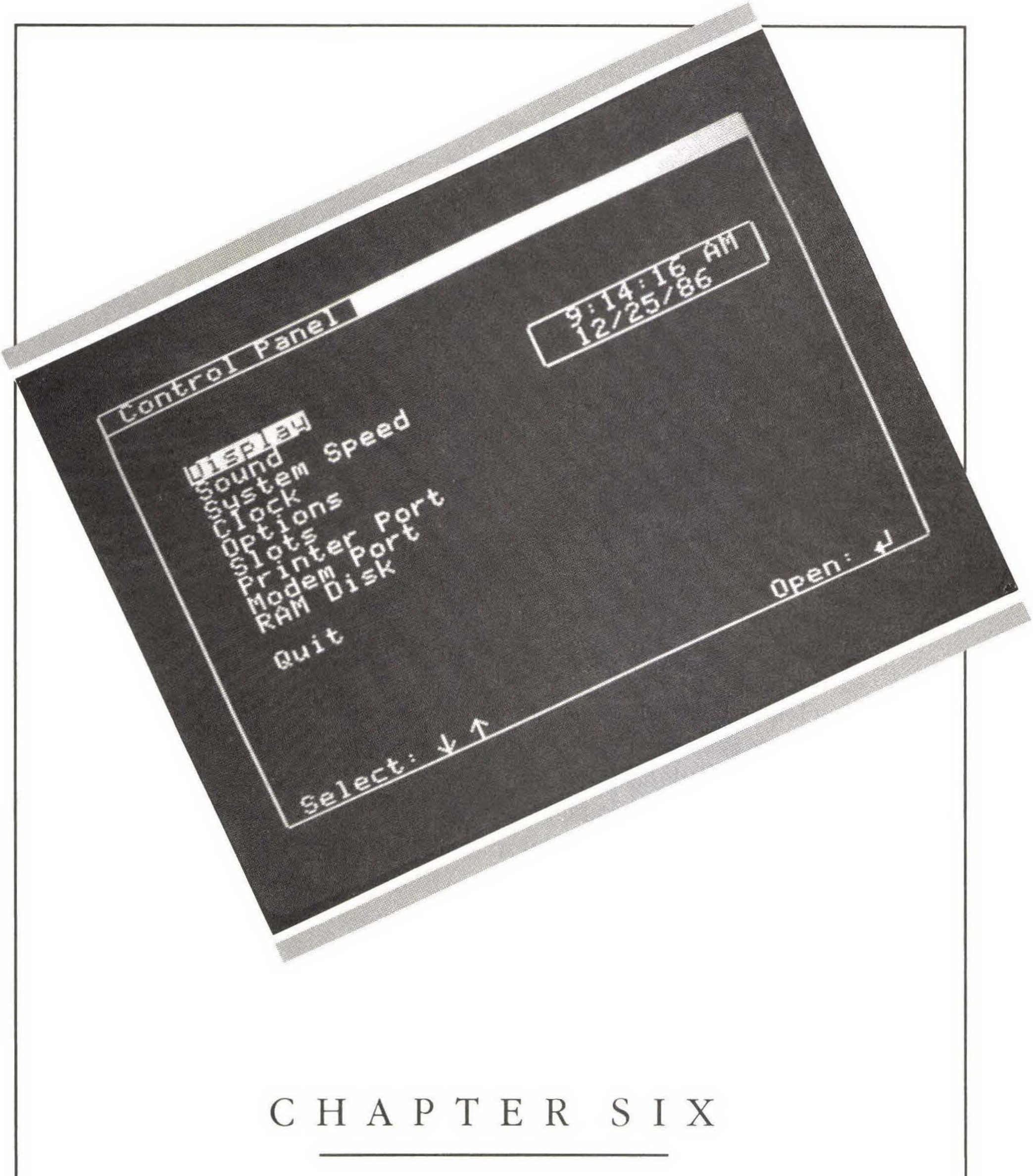
You can connect 3.5-inch drives and 5.25-inch drives by using disk drive controller cards, but you don't need to. If your drives have 19-pin connectors that fit the disk drive port, you can plug one drive into the port and daisy-chain other devices to it. If you have Disk II-type 5.25-inch drives, you'll still need to use your disk drive controller card.

Memory Expansion Card

There are two memory expansion cards available from Apple that you can use with the Apple IIGS. One, called the Apple II Memory Expansion Card, is designed for any of the general-purpose slots on the main circuit board. The other, called the Apple IIGS Memory Expansion Card, is designed for the special memory expansion slot. You can use the memory on either card as a RAM disk (memory that is accessed by the computer as if it were a disk), but only the Apple IIGS Memory Expansion Card can be accessed by new Apple IIGS applications as an extension of the RAM that's built into the computer.

Other Compatibility Questions

If you have questions about whether a particular device, or category of device, is compatible with the Apple IIGS, ask your Apple dealer. If the dealer doesn't know the answer to your question, he or she can get an answer by using the AppleLink network that links Apple dealers to Apple's technical support group in Cupertino, California.



```
Control Panel
                              9:14:16 AM
                              12/25/86
  Display
  Sound
  System Speed
  Clock
  Options
  Slots
  Printer Port
  Modem Port
  RAM Disk

  Quit

Select: ↓ ↑                        Open: ↵
```

CHAPTER SIX

Control Panel Program

The Control Panel is a program in ROM that lets you customize several aspects of your computer system, including the configuration of the serial ports, the colors of the text, background, and border on the display, the responsiveness of the mouse, the time interval before a pressed key starts repeating, the speed of the processor, and lots more. You can also use the Control Panel program to set the time and date of the built-in clock.

There are two ways to change Control Panel settings. One way is by accessing the program in ROM directly. You can do this either by starting up the computer with the Option key pressed down; or by pressing Option-Control-Reset, if the power is already on and you don't mind quitting the application you're using in the process; or by pressing Apple-Control-Esc, if the power is on and you want to return to your application after using the Control Panel.

The other way to change Control Panel settings is by choosing the Control Panel option from the Apple menu in the Finder. (See Figure 6.1.)

The difference between the two ways of changing Control Panel settings is the interface. The Finder's Control Panel interface is mouse-based, just like everything else about the Finder. The interface for the Control Panel in ROM is keyboard-based. You'll learn how to use the Finder's Control Panel in the *System Disk Guide* that comes with the Apple IIGS. You'll learn how to use the ROM version of the Control Panel in the *Apple IIGS Owner's Guide*. Changing settings by either method has the same effect.

The Control Panel

The following sections discuss the various system characteristics you can change in order of their appearance on the Control Panel menu in ROM. (See Figure 6.2.)

Display

The Display options are for customizing the look of the display. First you specify whether you're using a color or monochrome monitor. Then you can choose whether you want text displayed in 40 or 80 columns, and what color you want for the border, background, and text that is displayed on the screen. (See Figure 6.3.)

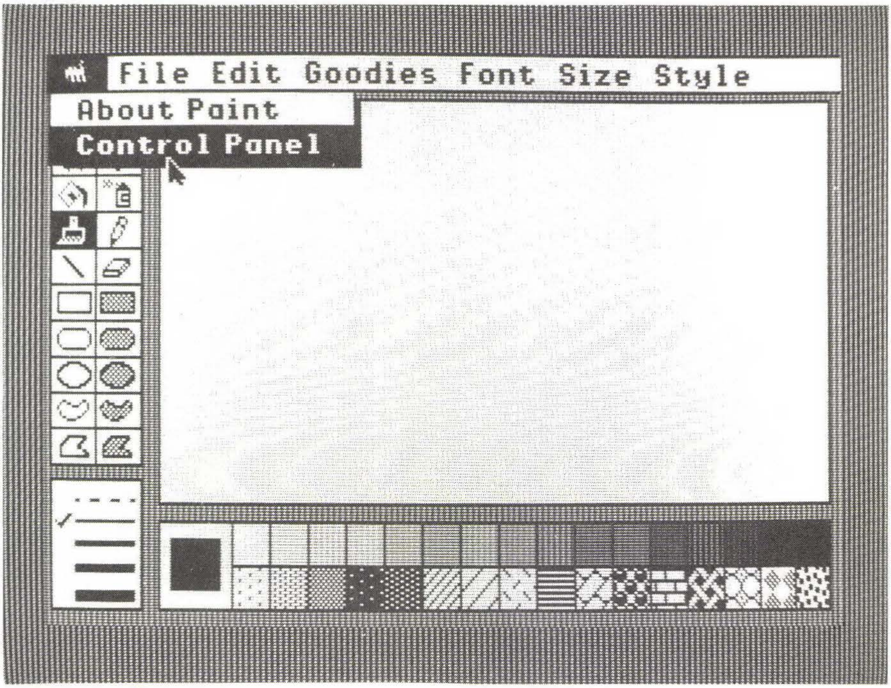


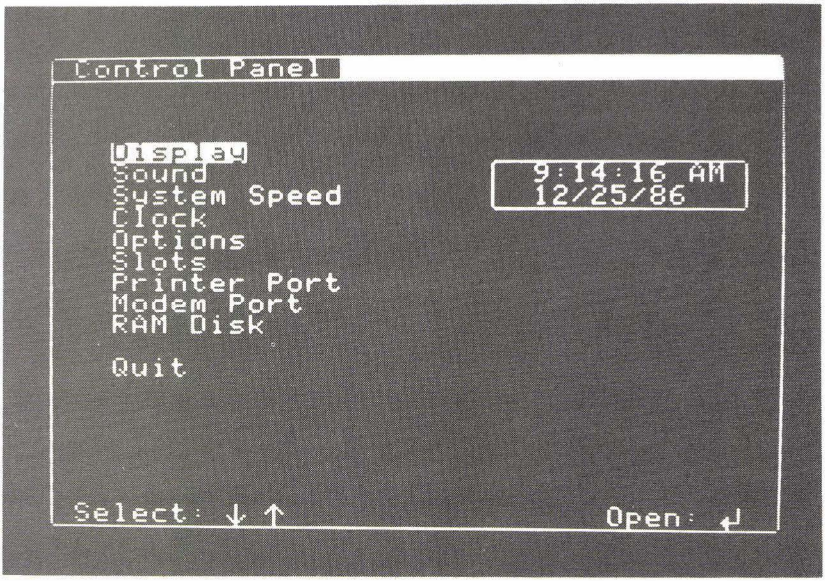
Figure 6 ▪ 1
Apple menu

Type

Most of the time you should select Color if you're using a color monitor and Monochrome if you're using a monochrome monitor, but in some classic applications the display will look better if you choose Color for a monochrome monitor or Monochrome for a color monitor. This is because the Apple IIGS generates color slightly differently than earlier models of the Apple II did. Use the setting that gives you the better display. With new applications you can leave it set to Color with color monitors and Monochrome if you have a monochrome monitor.

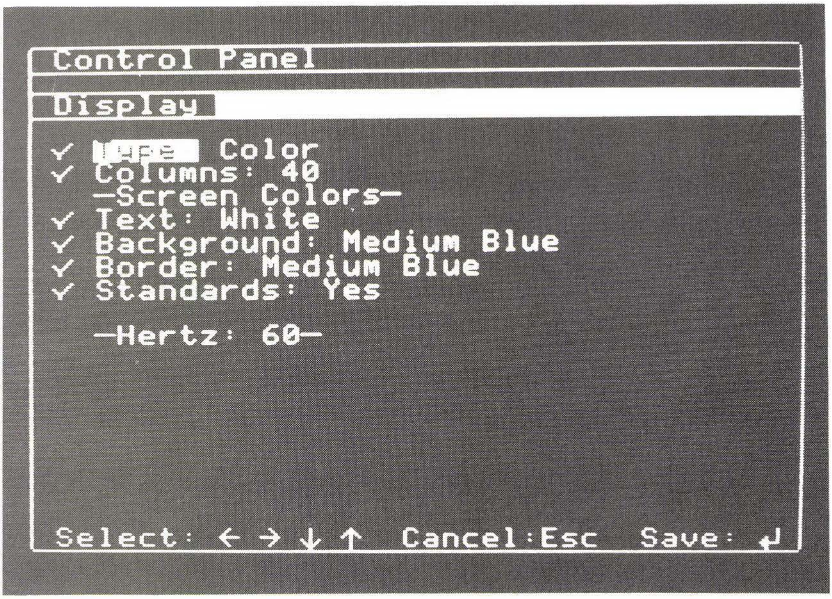
Columns

Column refers to a column of characters displayed on the screen. The original Apple II was designed to display a maximum of 40 columns. This was all that TV sets could handle. Monitors can display 80 columns clearly, so that has been an Apple II option almost from the dawning of Apple II history. On early Apple II's and on the Apple



```
Control Panel

Display                              9:14:16 AM
Sound                                12/25/86
System Speed
Clock
Options
Slots
Printer Port
Modem Port
RAM Disk

Quit

Select: ↓ ↑                          Open: ↵
```

***Figure* 6 ▪ 2**
Control Panel menu in ROM



```
Control Panel

Display
✓ Type   Color
✓ Columns: 40
  -Screen Colors-
✓ Text: White
✓ Background: Medium Blue
✓ Border: Medium Blue
✓ Standards: Yes

  -Hertz: 60-

Select: ← → ↓ ↑  Cancel:Esc  Save: ↵
```

***Figure* 6 ▪ 3**
Display

IIe you had to add an 80-column card, but the 80-column capability is built into the Apple IIGS.

Some programs are designed so that information is displayed in 40 or 80 columns, and they will work that way regardless of whether you choose 40 or 80 columns in the Control Panel. Other applications will ask you whether you want to use a 40- or 80-column display and they will display information that way regardless of the 40/80 column setting in the Control Panel. Some classic applications will only work if you have Columns set to 40. If you have trouble starting an old program on the new machine, try changing the column setting.

Screen Colors

There are two ways that applications can put text on the display. One way is by using the text generator in ROM. The other way is to treat text as graphics and draw it on the screen according to instructions in an application program.

With programs that use the text generator in ROM, you can choose the color of the text, the background, and the border from these 16 colors: pink, light green, yellow, aquamarine, white, deep red, dark blue, purple, dark green, dark gray, medium blue, light blue, brown, orange, light gray, black.

You see these colors only if you're using an RGB color monitor, because NTSC color monitors switch into monochrome mode for displaying text. Therefore with monochrome monitors and NTSC color monitors, the "color" you choose for text and background appears as a shade of gray.

As you change colors by using the Control Panel program, the Control Panel display changes to show how different choices look. There are some wonderful combinations—like the white on royal blue that is the standard setting—but there are also some ghastly combinations, and some where there is so little contrast that you can barely see the text against the background.

Frequency Setting

The Hertz setting shown at the bottom of the Display screen indicates the frequency of signals sent to the monitor. Different countries have different Hertz standards. The U.S. standard is 60 Hertz. If the computer is sending signals to the monitor at the wrong Hertz setting, the image on the screen will either be rolling or slightly off center. *The Apple IIGS Owner's Guide* explains how to restore the correct setting.

Sound

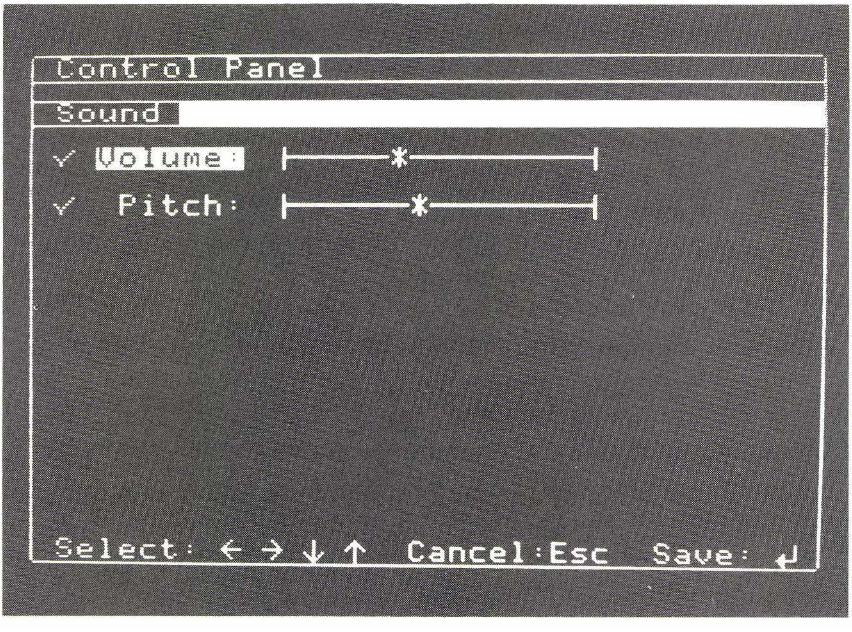
The Apple IIGS has a built-in speaker that broadcasts the music, speech, and assorted sound effects that programmers use in their applications. You can adjust the volume of these sounds and change the pitch of the beep that accompanies error messages by using the Sound option in the Control Panel. (See Figure 6.4.)

As you press the right or left arrow to change the volume or pitch, a beep lets you hear the effect of your change.

System Speed

The Apple IIGS can operate at one of two speeds: normal (1 MHz) and fast (2.8 MHz). The names are somewhat misleading because the fast speed is normal for the 816 processor in the Apple IIGS, but if

***Figure* 6 ▪ 4**
Sound

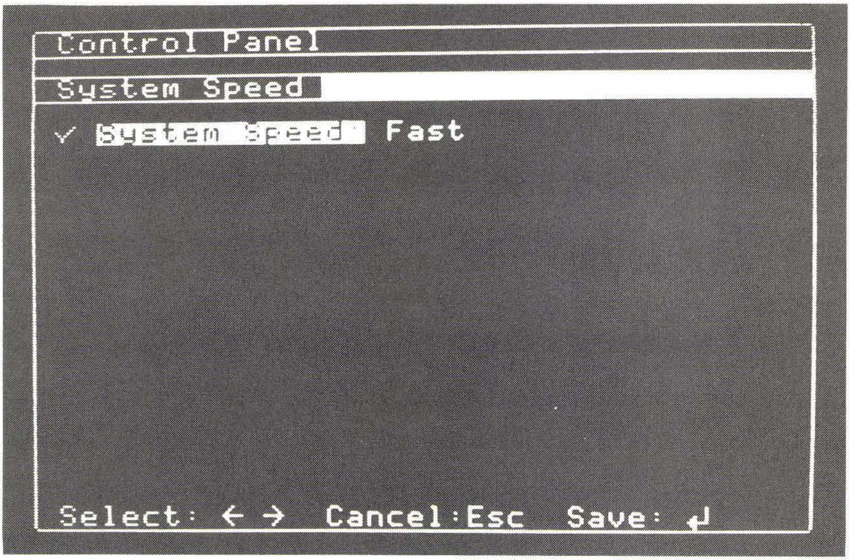


Control Panel

Sound

✓ Volume: |———*————————|

✓ Pitch: |————*———————|



Control Panel

System Speed

✓ System Speed Fast

Select: ← → Cancel:Esc Save: ↵

***Figure* 6 ▪ 5**
System speed

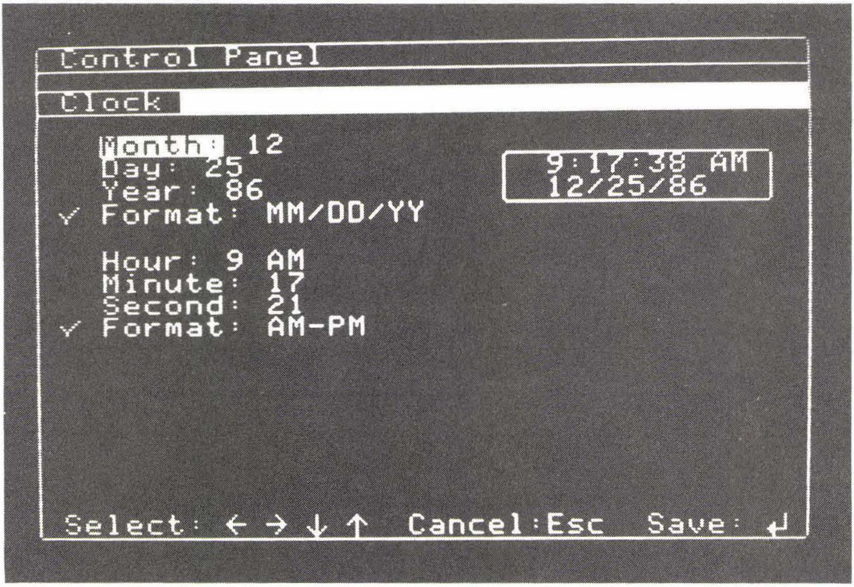
fast were called "normal," normal would have to be called "slow" and there's nothing very slow about 1 MHz.

The Apple IIGS will run all programs at the faster speed. This speed will add new zip to old applications, especially those that involve a lot of sorting (databases) and calculating (spreadsheets). But the fast speed may affect the timing of some games and other animated classic applications. If ''fast'' speeds up games to the point where they are unplayable, you can change the Apple IIGS's system speed from fast to normal using the Control Panel. (See Figure 6.5.)

In a few cases the method of copy protection a program uses is speed dependent. If a program uses a speed dependent copy-protection scheme, you won't be able to run it at all on the Apple IIGS until you change the speed to normal. If you have trouble getting a classic application to work on the Apple IIGS, try changing System Speed. You may have to restart the computer by using the on-off switch before the new speed will go into effect.

Clock

The Apple IIGS has a built-in clock. New applications can use the clock to mark files with the time and date when you created or last revised them. This feature is helpful when you have two disks with docu-



```
Control Panel

Clock
   Month: 12                          9:17:38 AM
   Day: 25                            12/25/86
   Year: 86
✓  Format: MM/DD/YY

   Hour: 9 AM
   Minute: 17
   Second: 21
✓  Format: AM-PM

Select: ← → ↓ ↑  Cancel:Esc  Save: ↵
```

***Figure* 6 ▪ 6**
Clock

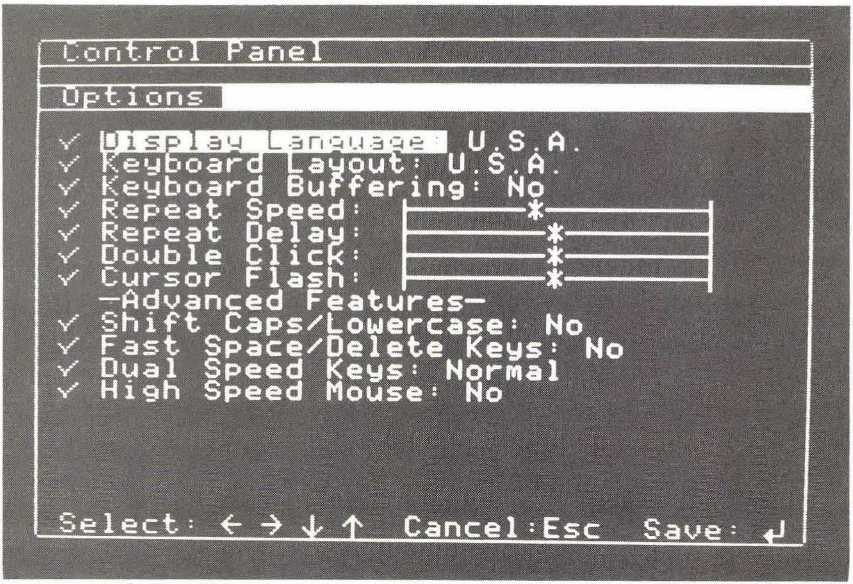
ments named "Report" and you don't know which is the latest. Having a built-in clock also makes it possible to tell your communications program to send a document to another computer in the middle of the night when rates are cheapest.

Desk accessories can use the built-in clock to display the time on your screen in all sorts of elaborate ways. There are desk accessories for the Macintosh that show the changing time on a full screen drawing of Big Ben. No doubt you'll be able to get equally creative clock accessories for the Apple IIGS.

You can set the time and date of the built-in clock by using the Control Panel program. (See Figure 6.6.) You can also choose how you want the time and date presented. For time, you can use the 24-hour format popular in military and European circles, or the AM/PM format. For date, you can display month first, or day first in the European fashion.

Options

Options is a catch-all term for settings you can change that affect the behavior of the keyboard and the mouse. (See Figure 6.7.)



```
Control Panel
Options
✓ Display Language: U.S.A.
✓ Keyboard Layout: U.S.A.
✓ Keyboard Buffering: No
✓ Repeat Speed:       |-------*--------|
✓ Repeat Delay:       |----------*-----|
✓ Double Click:       |----------*-----|
✓ Cursor Flash:       |----------*-----|
  -Advanced Features-
✓ Shift Caps/Lowercase: No
✓ Fast Space/Delete Keys: No
✓ Dual Speed Keys: Normal
✓ High Speed Mouse: No

Select: ← → ↓ ↑  Cancel:Esc  Save: ↵
```

Figure 6 • 7
Options screen

Keyboard Layout

The character generator in ROM is set up to display an S on the screen when you press the S key on the keyboard. However, you can use the Control Panel to change the keyboard layout from standard to Dvorak or to one of eight international keyboard layouts. It won't change the physical layout of the keyboard, but it will change the character that is displayed when you press a given key.

Dvorak is a keyboard arrangement that promotes greater typing speed and efficiency by locating the most-used keys in the middle or "home row." In the Dvorak layout the E key is where the D key is on a standard keyboard. If you select the Dvorak option without getting replacement Dvorak keycaps, you'll get an E when you type a D. If the keys you type don't match the characters that appear on the display, check to see if someone has been playing with your Control Panel.

Apple IIGS's that are sold in France will come with a French keyboard and with the Control Panel set up for the French keyboard layout. Earlier models of the Apple II sold in other countries had to have a special character generator ROM designed for that country. With the Apple IIGS, the keyboard layout information for eight countries is in one ROM. This means that multilingual users can use the Control Panel to alternate between keyboard layouts instead of alternating between two computers.

Keyboard Buffering

The keyboard buffer is a small area in memory where keystrokes are stored when the processor is processing and can't immediately deal with them. When the processor finishes whatever it was doing, it retrieves the keystrokes from the buffer and deals with them in the order in which they came in. You can turn the buffer on or off by using the Keyboard Buffering option.

Keyboard buffering lets you get quite a bit ahead of some programs. While the processor is busy saving a document on a disk, you can be typing instructions to have it printed, in draft mode, starting on page 3. The trouble with buffering is that if you don't remember right, or don't type right, you could come back to discover that it printed three copies rather than starting on page 3. If keyboard buffering gets you into trouble, you can turn it off. If you trust your memory and your typing or if you like surprises, you can turn it on.

Repeat Speed and Repeat Delay

When you hold down a key on the keyboard it repeats. You can change the speed at which it repeats by using the Repeat Speed option.

Having keys repeat is a handy feature when you want a row of dashes or dots, but it's a nuisance if you wanted just one "o" at the end of "Hello" and you got two because you let your finger rest too long on the last letter of the word. If you find the keyboard giving you two characters for the price of one keypress, you can change the interval of time before a pressed key starts repeating by using the Repeat Delay option.

Double-Click Speed

You can also change the computer's responsiveness to clicks of the mouse button. Pressing and releasing the mouse button twice in rapid succession is called double-clicking. It's a shortcut you can use in mouse-based programs to, for example, select and open an icon in one step. The two clicks are interpreted as a double click only if the successive clicks are done within a certain time interval. If you have trouble doing a double click in the time allocated by the standard Control Panel setting, you can change the interval.

Cursor Flash

The cursor (insertion point) flashes to distinguish it from other characters on the screen. If the interval between flashes irritates you or distracts you or just doesn't quite suit you, you can speed it up or slow it down by using the Cursor Flash option.

Advanced Features

The Advanced Features on the Options screen aren't for everyone. They're for speed and efficiency fanatics.

Using Shift to Get Lowercase

When you set the Shift Caps Lowercase option to Yes, you can type lowercase letters when the CAPS LOCK key is down by holding down the Shift key while you press the letter. Shift works the opposite of the way it does when CAPS LOCK isn't down.

Turbo Delete and Space

By using the option called Fast Space Delete Keys, you can press Control-Delete in order to double the speed of the Delete key. Also you can press Control-spacebar in order to double the speed at which spaces are added when you press the spacebar.

Dual Speed Keys

By setting Dual Speed Keys to Yes, you can double the speed at which the cursor moves by holding down an arrow key while you press Control.

High Speed Mouse

While you're learning to maneuver the mouse, the pointer might seem to have a life of its own, but when you've become an old hand, you may decide you want the mouse to be a little more responsive. You can turbo-charge the pointer—make it move twice as far for a given mouse move—by turning on the High Speed Mouse option in the Control Panel.

Slots

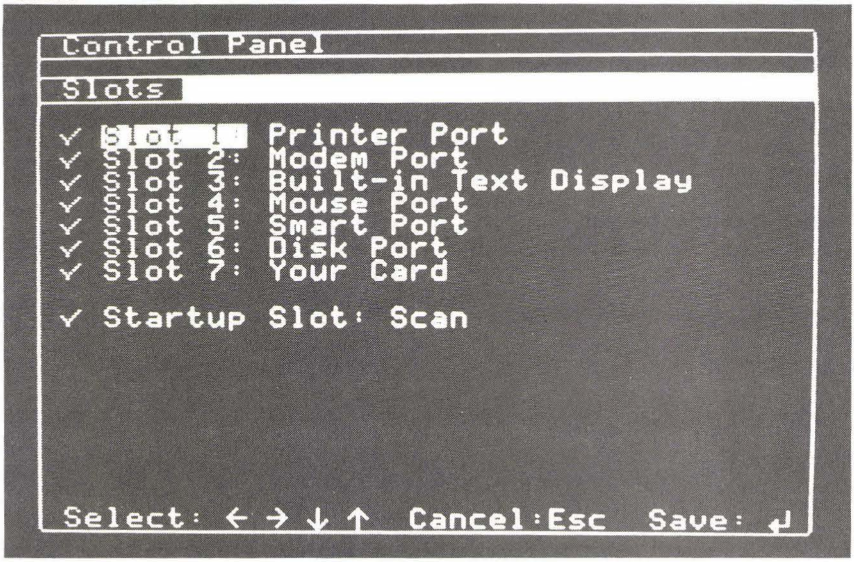
The ports on the back of the Apple IIGS are designed to emulate slots containing interface cards so that software that looks for devices connected to slots will find devices connected to ports. It's all part of keeping the Apple IIGS compatible with software that already exists for the Apple II family.

You can't have a port and its corresponding slot active at the same time, but you can easily activate one or the other by using the Control Panel program. (See Figure 6.8.) Keep in mind that you have to restart the computer before your new slot or port setting will go into effect.

Ports are active unless you explicitly activate a slot. When you activate a slot, the words "Your Card" appear next to the slot number in place of the words Printer Port, Modem Port, Disk Drive Port, and so on.

You may be wondering about the Smart Port listed on the Slots screen. There is no such port on the back panel. That's too bad

***Figure* 6 ▪ 8**
Slots



```
Control Panel

Slots

✓ Slot 1: Printer Port
✓ Slot 2: Modem Port
✓ Slot 3: Built-in Text Display
✓ Slot 4: Mouse Port
✓ Slot 5: Smart Port
✓ Slot 6: Disk Port
✓ Slot 7: Your Card

✓ Startup Slot: Scan

Select: ← → ↓ ↑  Cancel:Esc  Save: ↵
```

because it would have been interesting to see what icon they would have come up with. "Smart port" refers to the disk drive port when it's being used by a certain category of devices called "smart port devices." Smart port devices appear to be connected to an interface card in slot 5, while 5.25-inch drives appear to be on a disk controller card in slot 6.

Startup Slot

The last option on the Slots screen is Startup Slot. The standard startup slot setting is "Scan." This means that when you turn on the power, the computer will start at the highest-numbered slot and go down the row of slots looking for a startup device.

If you have a 5.25-inch drive connected to the disk drive port, it appears to be in slot 6, so the computer will start up from it if there's a disk in it. If there's no 5.25-inch drive or no disk in the 5.25-inch drive, the computer will proceed to slot 5, where it will find a 3.5-inch drive connected to the disk drive port, if there is one.

This process is called *scanning*. If you don't want the computer to start up from the device in the highest-numbered slot, you can designate a particular slot as the startup slot. If you want to start up from a hard disk connected to slot 4, you can set Startup Slot to 4 and the computer will start up from the device in that slot, even if you have devices connected to slots 7, 6, and 5.

Serial Ports

Different serial devices are set up to exchange information with computers at different speeds and in different data formats. When you connect serial devices to a computer by using a serial interface card, you can set switches on the card to control how information is sent. (See Figure 6.9.)

On the Apple IIGS the serial interface is built in, but you can set software "switches" that accomplish the same thing by adjusting the serial port settings in the Control Panel. (See Figure 6.10.)

It's a good idea to try using your printer or modem before you change any of the settings. The printer port is set up to exchange information with the Apple ImageWriter series of printers. If you're using an ImageWriter or a printer that uses the same settings as the

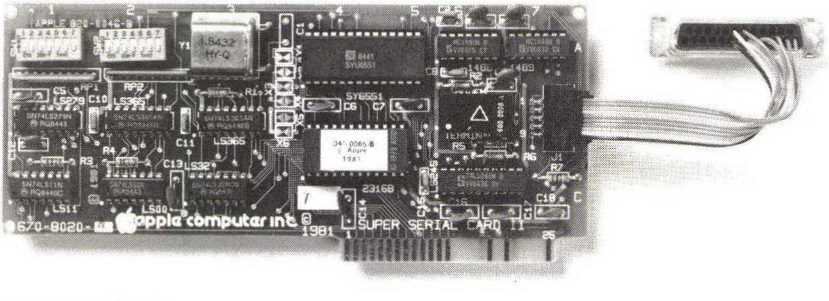
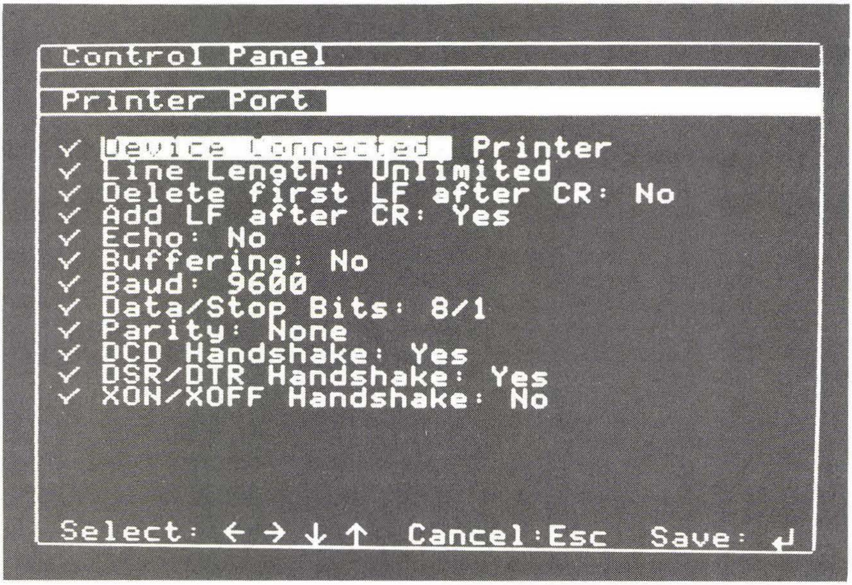


***Figure* 6 ▪ 9**
Serial interface card

ImageWriter, you won't need to change anything. And if you're using a modem to communicate with The Source or CompuServe at 1200 baud, you won't need to change any of the modem port settings. The only time you will need to change serial port settings is when your printer or the information service requires different settings. Even then, many applications will override Control Panel serial port settings, so you can tell the application what kind of printer you have or how the information service wants to exchange information and can leave the Control Panel settings as they are.

If you do need to change settings, find out what specifications the

***Figure* 6 ▪ 10**
Printer port settings



```
Control Panel

Printer Port

✓ Device Connected: Printer
✓ Line Length: Unlimited
✓ Delete first LF after CR: No
✓ Add LF after CR: Yes
✓ Echo: No
✓ Buffering: No
✓ Baud: 9600
✓ Data/Stop Bits: 8/1
✓ Parity: None
✓ DCD Handshake: Yes
✓ DSR/DTR Handshake: Yes
✓ XON/XOFF Handshake: No

Select: ← → ↓ ↑  Cancel:Esc  Save: ↵
```

device needs before you change anything. If you can't find all the specifications, set the ones you can find and leave the others. If the device doesn't work, contact your dealer or the manufacturer of the device.

Here's what the various settings control:

Device Connected: The serial port on the left is set up for a modem and the serial port on the right is set up for a printer. You can connect a printer to the modem port and a modem to the printer port, but you have to change some settings—starting with "Device Connected."

Line Length: If the printer or application doesn't give you a way of setting the right margin—if your words keep printing right off the page—you can specify the maximum number of characters per line by using this option.

Delete LF After CR: LF stands for linefeed. CR stands for carriage return. If both the printer and an application are inserting linefeeds after carriage returns, if everything comes out unintentionally double-spaced, you can delete one of the extra linefeeds by using this option.

Add LF After CR: If your printer doesn't automatically add a linefeed after every carriage return (if lines keep printing on top of each other), you can use this option to tell the computer to do it.

Echo: Some modems send a copy of every character they receive back to the transmitting device as a way of confirming that the message came through. This copy, called an *echo*, is sent by full-duplex modems but not by half-duplex modems. If you are sending messages to a half-duplex modem and want to see the message on your own screen, you can have your own computer do an echo. It doesn't tell you whether the message got through intact, but it does tell you what it looked like when it was sent. If you ever get ttwwoo of every character on your screen, it means that you're dealing with a full-duplex modem and you have echo turned on in the Control Panel.

Buffering: A buffer is an area in RAM where information is kept when the computer or peripheral device is busy and can't process it right away. Leave buffering off unless the manual that came with the device says to turn it on.

Baud: Baud is the speed, measured in bits per second (bps), that the devices will use to exchange data. The most common setting for modems is 1200 or 300 baud; 9600 baud is the most common setting for printers.

Data Stop Bits: Data bits refers to the number of bits used to represent each character of information. Characters can be represented with 7 or 8 data bits. One or two stop bits are used to indicate where one string of data bits ends and the next begins. It's important that the computer and the device agree on the format of the data they exchange.

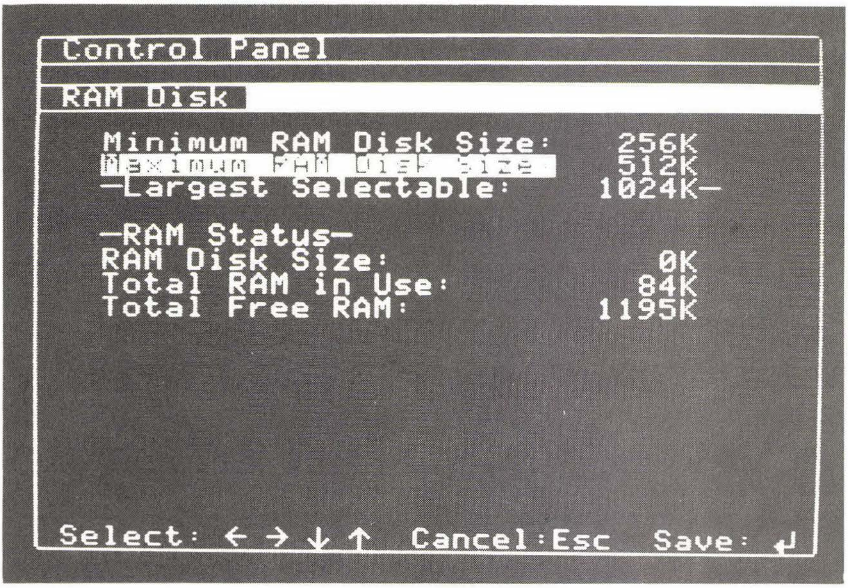
Parity: Parity means error checking. The options are no parity (the most common setting), odd parity, and even parity. If the devices are using odd parity as an error checking system, the transmitting device adds an extra bit set to 0 or 1 to make the bits for that character add up to an odd number. For example, if the data string is 0000011, the transmitting device would add another 1 to make the total an odd number. If the devices are using even parity, the transmitting device adds either a 0 or a 1 to make the total come out even.

Handshake Signals: DCD stands for Data Carrier Detect. DSR stands for Data Set Ready and DTR stands for Data Terminal Read. XON and XOFF are ASCII characters. XOFF tells the transmitting device to stop transmitting. XON tells the transmitting device to resume transmitting. All of these settings regulate the exchange of information between the computer and a peripheral device. Don't change the settings unless the manual that came with the device specifies an alternative setting.

RAM Disk

There are two ways you can use the memory on your Apple IIGS memory expansion card. With new programs, you can use it as an extension of the RAM that's built into your computer. New programs will see that it's there and will use it without any special instructions from you.

With classic applications, you can use the memory on the card as a RAM disk—memory that is accessed like a disk but that is temporary like other RAM (the information in it disappears when you turn off the



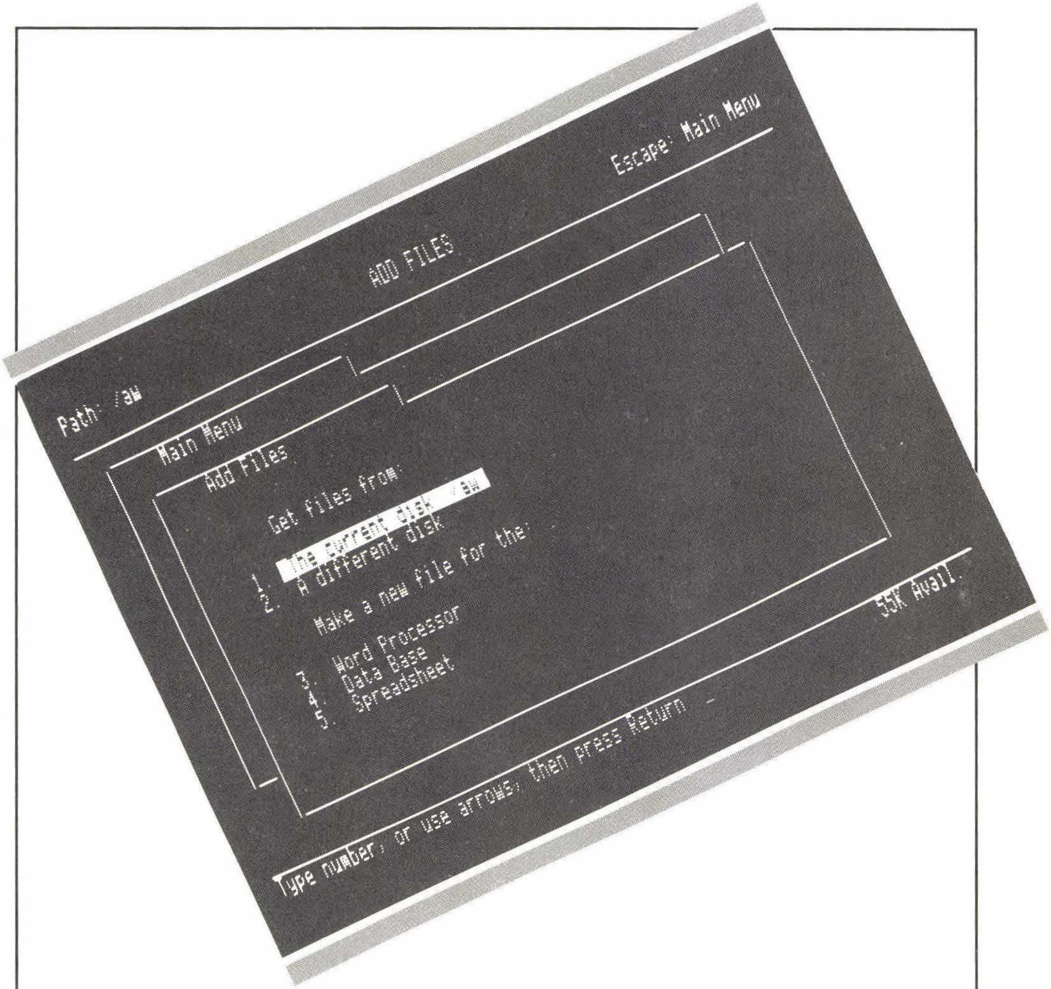
```
Control Panel

RAM Disk

    Minimum RAM Disk Size:      256K
    Maximum RAM Disk Size:      512K
    -Largest Selectable:       1024K-

    -RAM Status-
    RAM Disk Size:                0K
    Total RAM in Use:            84K
    Total Free RAM:            1195K

 Select: ← → ↓ ↑  Cancel:Esc  Save: ↵
```

***Figure* 6 ▪ 11**
RAM disk

computer's power). The advantage of copying classic programs onto a RAM disk is that those programs don't know about all the memory that's available on the Apple IIGS and they use the memory sparingly as if there were only 48K or 64K in the whole system. To save space in memory for your documents, these classic applications will put only part of the program in memory at a time and access the program disk in the disk drive when they need to put a different part of the program into memory. If you run these programs from a RAM disk instead of from a regular disk drive, the access time is much faster.

The drawback of using the memory on your card as a RAM disk is that once you put information in the RAM disk on the card, that space is no longer available to new applications that access the memory on the card as an extension of the RAM that's built into the computer.

The RAM disk option in the Control Panel lets you set a limit on how much room on the card is available to new applications and how much is available for use as a RAM disk. (See Figure 6.11.) If you want to leave all the space for new applications that use the space automatically, leave the maximum and minimum RAM disk sizes set to 0.



```
Path: /aw                                  Escape: Main Menu

                        ADD FILES

Main Menu
  Add Files

    Get files from:

    1. The current disk  /aw
    2. A different disk

    Make a new file for the:

    3. Word Processor
    4. Data Base
    5. Spreadsheet

Type number, or use arrows, then press Return        55K Avail.
```

CHAPTER SEVEN

Basic Computer Concepts

If computers are so wonderful, why doesn't every family in America own one? Everyone has a television; everyone has a phone. You hear a lot about the computer revolution and how it's going to change the future, but so far it's changed the present for only a relatively small (though fast-growing) number of people.

One reason for this is that computers are still pretty complicated machines. You can't just press a couple of buttons and expect them to do what you want. Until computers are as simple to use as phones and televisions, there's going to be some effort involved in learning how to use them.

It's not a tremendous effort; after all, children in elementary school are learning in droves. And the Apple IIGS, because of the mouse interface (see Chapter 2), is much easier to learn and use than its Apple II forebears. Still, it is helpful to be familiar with certain basic words and concepts. This chapter covers enough of them to allow you to approach the Apple IIGS without the nervousness you might feel toward a total stranger. It answers these three questions:

- What is an application program, and what do I use one for?
- What is programming, and why would I ever want to do it?
- What happens to the information I put into the computer?

Application Programs

A computer is nothing without a program. Your Apple IIGS, in spite of all its dazzling features, would sit sadly still on your desk if it had no programs to work with.

Although a computer program looks to most people like miles of incomprehensible squiggles, it's really no more than a list of instructions, written in a form that a computer can understand. These instructions might tell the computer how to display words on the screen, for example, or add two numbers together, or erase an alien spaceship when you press a certain key at a certain moment. You decide what you want the computer to do and choose a program that tells the computer how to do it.

More likely than not, what you want to do with your Apple IIGS falls into one of six categories, and for each category there's a kind of program (called an *application* program):

What You Want to Do	What Kind of Application You Need
1. Write	Word processing programs
2. Calculate	Spreadsheet programs
3. Organize information	Database programs
4. Draw graphs or pictures	Graphics programs
5. Send information to other computers	Communications programs
6. Play games	Entertainment programs

You might call these the "Big Six" software categories. ("Software" and "computer program" mean the same thing.) There are plenty of programs that don't fit into any of these categories; the more than 10,000 programs written for the Apple II family (most of which will run on the Apple IIGS) include unique programs that don't fit into any category at all. Still, the vast majority of computer users spend most of their time on one or more of the Big Six.

Word Processing

When you want to write a letter, a memo, a poem, or a doctoral dissertation on your Apple IIGS, you'll use a word processing program to do it. Word processing programs make the computer function more or less like a typewriter—with some significant differences. Because the words you type appear on a screen rather than on a piece of paper, you can change them around and correct mistakes without resorting to gooey white fluid or erasers that leave gritty smudges all over the page. Most people abandon their typewriters joyfully when they see what word processing programs can do.

A few examples:

1. Suppose you discover, while reading over a paragraph you've just written, that you've typed "indepence" when you meant to type "independence." Adding three extra letters to a typewritten paragraph is likely to mean retyping at least a line, and possibly an entire page. With a computer, you add the missing letters, and the rest of the words in the paragraph simply move over to make room for them. It also works the opposite way when you find that you've typed "independendence" and need to get rid of three letters; the rest of the words in the paragraph move back to fill up the space.
2. You've written the final sentence of your report on Trends in Artichoke Marketing, and not until you proofread it do you realize that paragraph 5 should really be paragraph 2. With a

typewriter, you'd have to wad up your report and start over. With a computer, you can cause paragraph 5 to disappear from its current location and direct it to reappear again as paragraph 2.

3. After you've completed your short story (29 pages), you decide to change the heroine's name from Lucinda to Blake. On a typewriter that would mean 433 separate corrections. But you can ask a word processing program to find every occurrence of "Lucinda" in the story and change it to "Blake," and the whole operation can be completed in a minute or so.

Just about all word processing programs offer the features above—text insertion and deletion, "cut and paste," and "search and replace," as they're called in computer jargon. Most of them also offer other features, such as:

- Word wrap. When you come to the end of a line, you don't have to press the Return key (which is the equivalent of the carriage return on a typewriter). The program moves you automatically to the next line.
- Easily adjustable margins and tabs. You can type a letter with margins of one inch on either side, then change your mind and reset the margins to half an inch without retyping. Tabs work the same way: you can change them at any point, and the text will adjust itself.
- Automatic page numbering. You can tell the program what number you want your document to start with, and where you want the number to appear, and it will do the whole job for you. If you add a page or take one out, it will adjust the numbering accordingly.
- Headers and footers. You might want the date to appear at the bottom of every page, or the title of your report to show up in the top right corner. By establishing a header or footer, you tell the Apple IIGS to do this for you.
- Mail merge. Many word processing programs will let you write a letter, leaving spaces for names, addresses, and key words, and then will merge it with a list of recipients. The program prints out multiple copies of the letter, putting a different person's name and address on each one.

How you do all these things depends on what program you're using. With the Apple IIGS you have a choice between mouse-based and keyboard-based word processing programs, and you'll probably want to go for mouse-based. They're easier to learn, and faster to use.

Once you've perfected your document on the screen, you'll want to transfer it to a piece of paper. With some programs, this is easy. You select the word "Print" from a list of choices, and the printer types your document exactly as you see it on the screen. This is called "WYSIWYG"—"what you see is what you get." If your letter is double-spaced on the screen, it will be double-spaced on paper; if the margins are 3/4 inches wide on the screen, they will be on the paper too.

There are many programs, however, in which the correlation is not so direct. These programs can't display type on the screen so flexibly. They might not be able to double-space lines or show varied margins. Therefore they offer you a list of printer options and you choose what you want by typing the appropriate commands. You might type "LM1.5" to set the left margin at 1½ inches, or "DS" to double space. This is not quite so convenient as "wysiwyg," since you can't see exactly how your text is going to look before you print it, but you soon get a feeling for what the print commands will do, and eventually you will be able to use them to specify complex formats, with bulleted paragraphs, hanging indents, and even footnotes.

There are a few things that most word processing programs aren't very good at. One of them is typing words into printed forms, where you have to get the word exactly on the line or inside the little box. A typewriter (or a ball point pen) is still handiest for this. Another is typing two columns to a page. Only a few word processing programs allow you to do this with any ease.

For the most part, however, word processing programs are miracles of efficiency. More people use their personal computers for word processing than for anything else. Unless your writing is strictly confined to phone numbers and grocery lists, you'll want to get one for your Apple IIGS.

Spreadsheets

A spreadsheet is used for working with numbers. You can use one for any sort of numerical calculation that's done in rows and columns, such as . . .

- Making up the quarterly income statement for your business
- Budgeting your household expenses
- Keeping track of daily attendance at an art museum
- Doing your arithmetic homework
- Figuring out how much your monthly payment will be if you put $40,000 down on a decaying mansion in Florida

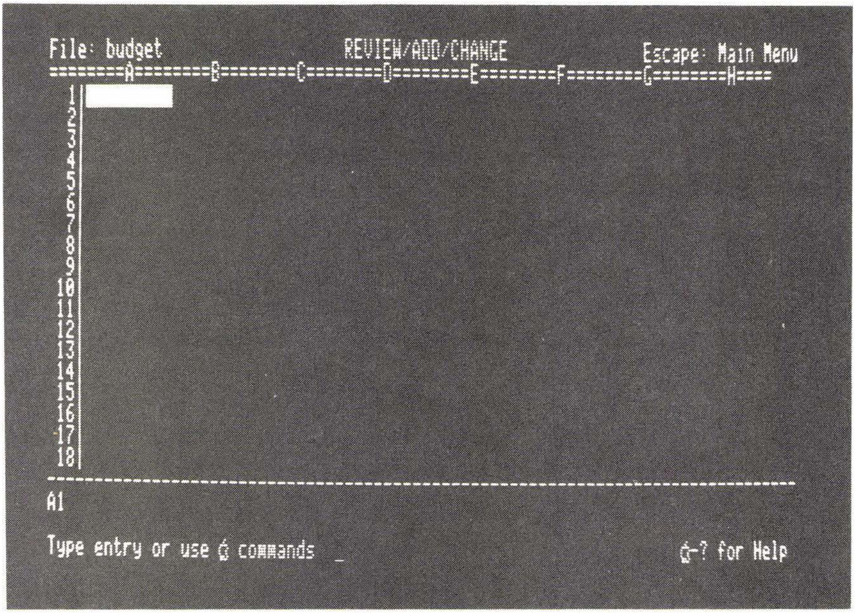
You can imagine a spreadsheet program as an enormous piece of ledger paper with certain magical properties. A typical computer spreadsheet looks like Figure 7.1.

Just as in a ledger book, the rows are labeled with numbers and the columns are labeled with letters. The intersections of the rows and columns are called *cells*, and in the cells you may type either words or numbers. Most spreadsheets have several hundred rows and columns—rather more than a typical page of ledger paper and, though not perhaps quite adequate for the Pentagon's budget, still certainly big enough for most business and personal uses.

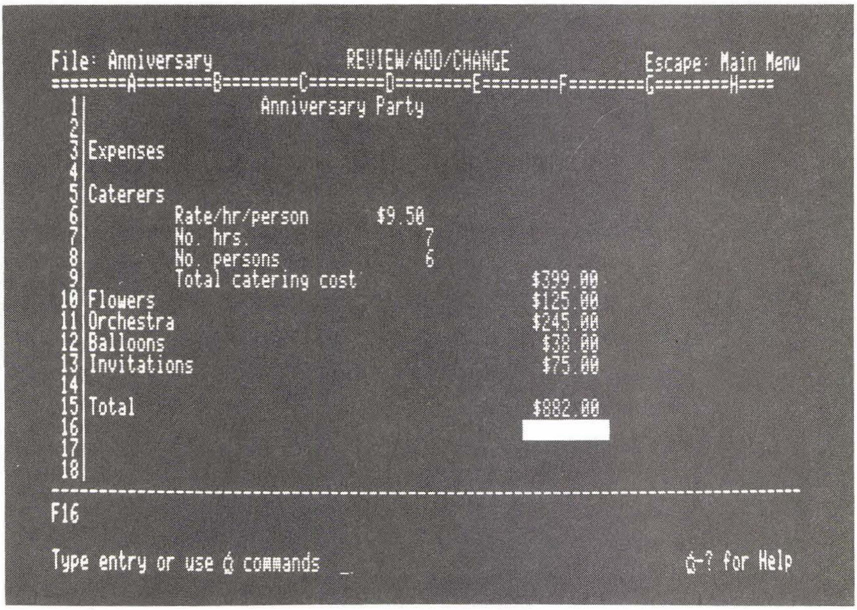
To illustrate how a spreadsheet works, let's suppose that you are planning a fairly elaborate party, someone's fiftieth wedding anniversary. On your spreadsheet you type a list of all the things you'll need to buy and how much each of them will cost (Figure 7.2).

Then you want to total them up. This is where the spreadsheet begins to work for you. In the cell where the total will go, you enter a formula rather than a number (Figure 7.3). The formula says to the spreadsheet program, "Take the numbers in cells F9 through F13, add them together, and put the result here." The Apple IIGS does the addition for you.

***Figure* 7 ▪ 1**
Blank spreadsheet

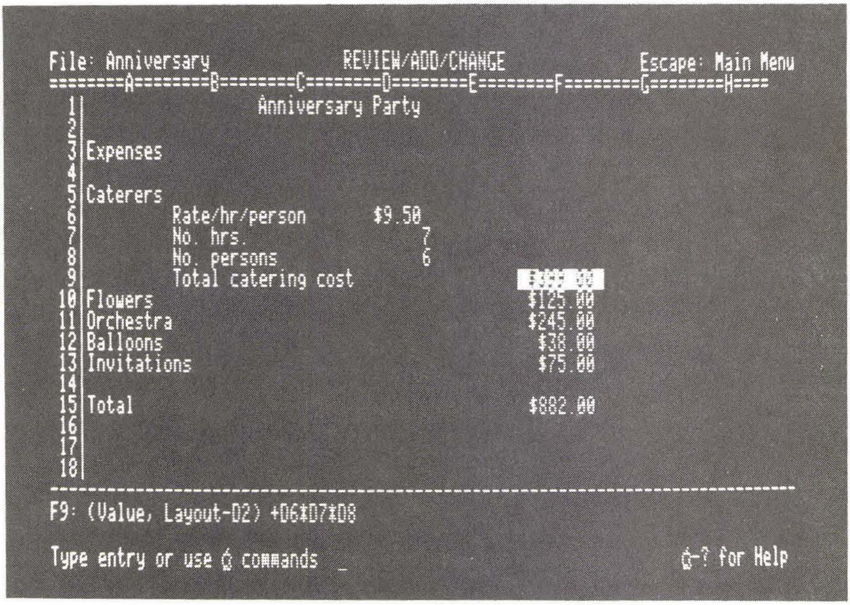


```
File: budget                 REVIEW/ADD/CHANGE                Escape: Main Menu
========A========B========C========D========E========F========G========H====
 1|
 2|
 3|
 4|
 5|
 6|
 7|
 8|
 9|
10|
11|
12|
13|
14|
15|
16|
17|
18|
------------------------------------------------------------------------------
A1

Type entry or use  commands _                                    -? for Help
```



```
File: Anniversary              REVIEW/ADD/CHANGE              Escape: Main Menu
=======A========B========C========D========E========F========G========H====
 1|                    Anniversary Party
 2|
 3|Expenses
 4|
 5|Caterers
 6|         Rate/hr/person        $9.50
 7|         No. hrs.                  7
 8|         No. persons               6
 9|         Total catering cost                  $399.00
10|Flowers                                       $125.00
11|Orchestra                                     $245.00
12|Balloons                                       $38.00
13|Invitations                                    $75.00
14|
15|Total                                         $882.00
16|
17|
18|
---------------------------------------------------------------------------
F16

Type entry or use Ó commands _                                  Ó-? for Help
```

Figure 7 ▪ 2
Spreadsheet with list typed in



```
File: Anniversary              REVIEW/ADD/CHANGE              Escape: Main Menu
=======A========B========C========D========E========F========G========H====
 1|                    Anniversary Party
 2|
 3|Expenses
 4|
 5|Caterers
 6|         Rate/hr/person        $9.50
 7|         No. hrs.                  7
 8|         No. persons               6
 9|         Total catering cost                  $399.00
10|Flowers                                       $125.00
11|Orchestra                                     $245.00
12|Balloons                                       $38.00
13|Invitations                                    $75.00
14|
15|Total                                         $882.00
16|
17|
18|
---------------------------------------------------------------------------
F9: (Value, Layout-D2) +D6*D7*D8

Type entry or use Ó commands _                                  Ó-? for Help
```

Figure 7 ▪ 3
Spreadsheet formula highlighted

Now suppose the caterers inform you that their rates have gone up. Instead of charging $9 an hour, they'll be charging you $11.50 an hour. If you had written your figures with a pencil instead of an electronic spreadsheet, you'd have to do some erasing and refiguring. Here you simply change the relevant numbers, and the new total is calculated for you instantly.

This ability to recalculate numbers that depend on other numbers is the spreadsheet's most sparkling talent. You might have a complex system of dependencies—a tax bracket that depends on an income level that depends on an hourly rate, for example—in which a change in any figure could affect several others. A spreadsheet lets you change any number you want and watch the effects of the change ripple across the whole thing. You can ask "what if" questions and get instant answers: What if I charge $20 an hour instead of $18? What if my income drops by 10 percent? What if my expenses for January are more than they were for December?

With a spreadsheet program on your Apple IIGS, you can do far more than you can with the fanciest pocket calculator. You can, for example, do the following:

- Set up templates for particular uses. You might need to calculate mortgage payments frequently, or figure out your return on investments, or record travel expenses. You can make up a spreadsheet with the appropriate formulas in the appropriate cells and simply plug in new figures for each separate instance. You can also buy templates if you don't want to or don't know how to set them up yourself. For example, you might get templates that allow you to do your taxes, figure depreciation, or calculate compound interest.
- Do projections. Based on last month's sales figures, what profit margin can you expect if you project a growth of 3 percent?
- Use sophisticated mathematical functions that are built into the spreadsheet, so that you don't have to set up the formulas for them yourself. A spreadsheet might include the formula for calculating the net present value of an investment, for example, or for taking the square root of a number.
- Translate your figures into graphs. The spreadsheet programs designed specifically for Apple IIGS are likely to have a graphics program built in, to take advantage of Apple IIGS's color graphics ability. With a few commands you can turn a row of numbers into

a pie chart, a bar chart, or a line graph. When you change a number on the spreadsheet, the graph changes, too.

Most people probably use spreadsheets in connection with money: profit and loss statements, budgets, expense records, taxes. But spreadsheets can have nonfinancial uses too. A birdwatcher might use a spreadsheet to record sightings of different kinds of birds. A research firm might use one for recording population figures and projecting growth or decline. A farmer could use one to keep track of the amount of fodder consumed by his dairy cows, and its relationship to the amount of milk they produce. Numbers come into most people's lives in one way or another; if you have a ledger book in your life or use your calculator for more than figuring tips in restaurants, you can probably find a good use for an Apple IIGS spreadsheet.

Data Base Programs

A data base program turns the Apple IIGS into an information organizer. You can think of a data base program as a substitute for a file cabinet, a Rolodex, a stack of index cards, or even a shoe box, if that's where you keep your bits and pieces of information. A database program can take any kind of a list—from a warehouse inventory to the list of everyone you know in Rangoon—and let you alphabetize it, find the item you want in it, or sort out one kind of item from another kind in seconds.

Imagine this situation: You run a small mail-order business out of your house, selling hard-to-find plants and seeds. Over the years you've built up a list of customers, to whom you mail your catalog and from whom you can expect an order almost every year. There are now 738 people on this list. You have the name, address, and phone number of each customer written on an index card, along with that customer's special interests, and three different dollar amounts showing his or her purchases for the last three years.

It's a very useful stack of index cards. You use it for your mailing list; you use it to send out notices of special acquisitions to those who will be most interested; you use it to select your very best customers for special bargain rates. The only trouble with it is that it fills up six shoe boxes, and sorting through it for the information you want has turned into a monster headache.

This is the kind of situation a data base program was made for. Instead of listing your customers on index cards, you list them on the Apple IIGS, on a grid that might look something like the one in Figure 7.4.

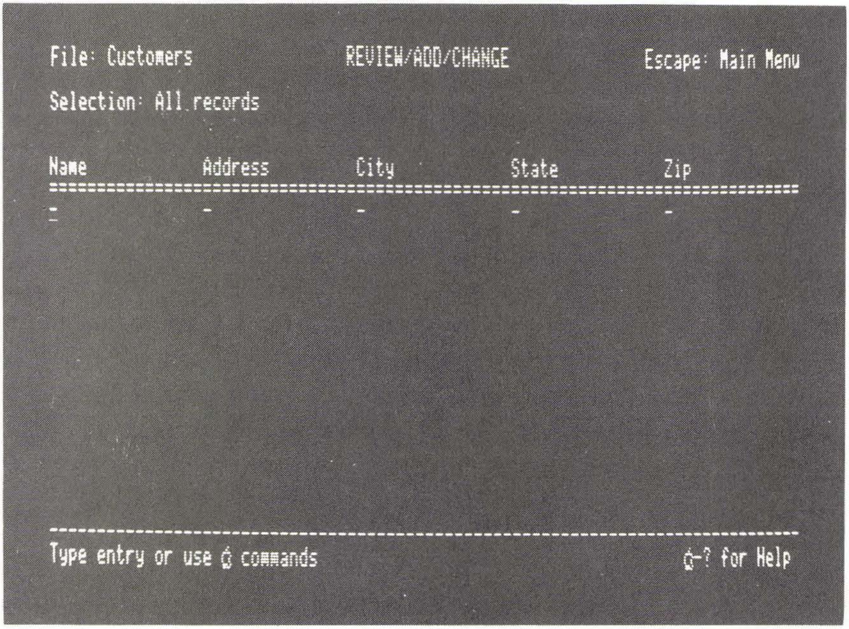
Each row represents one of your index cards; this is called a *record*. Each column represents a different category of information on that card; this is called a *field*. The data base program lets you decide what these categories are, and usually how wide a column you want to put them in.

Now you type in all the information that's on the cards. (See Figure 7.5.)

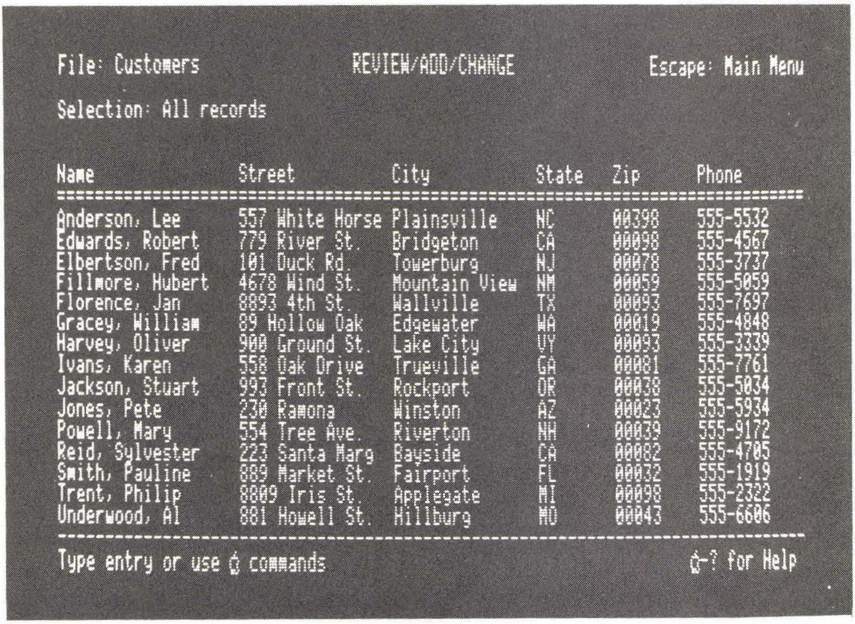
This is a fairly long, tedious job, but the results are more than worth it. Once you have all your customers in the database, you can do any of the following in, quite literally, a few seconds:

- List your customers alphabetically by name.
- List them by state.
- Find all the ones who are interested in hot chilis.
- List them according to the amount of their order in 1985.
- List all the customers who live in Iowa and ordered more than $300 worth of seeds last year.

Figure 7 • 4
Data base grid, with categories filled in



```
File: Customers              REVIEW/ADD/CHANGE             Escape: Main Menu
Selection: All records

Name          Address          City          State          Zip
================================================================================
-             -                -             -              -

--------------------------------------------------------------------------------
Type entry or use  commands                                          -? for Help
```



```
File: Customers              REVIEW/ADD/CHANGE              Escape: Main Menu

Selection: All records

Name               Street          City            State  Zip     Phone
================================================================================
Anderson, Lee      557 White Horse Plainsville     NC     00398   555-5532
Edwards, Robert    779 River St.   Bridgeton       CA     00098   555-4567
Elbertson, Fred    101 Duck Rd.    Towerburg       NJ     00078   555-3737
Fillmore, Hubert   4678 Wind St.   Mountain View   NM     00059   555-5059
Florence, Jan      8893 4th St.    Wallville       TX     00093   555-7697
Gracey, William    89 Hollow Oak   Edgewater       WA     00019   555-4848
Harvey, Oliver     900 Ground St.  Lake City       VY     00093   555-3339
Ivans, Karen       558 Oak Drive   Trueville       GA     00081   555-7761
Jackson, Stuart    993 Front St.   Rockport        OR     00038   555-5034
Jones, Pete        230 Ramona      Winston         AZ     00023   555-5934
Powell, Mary       554 Tree Ave.   Riverton        NH     00039   555-9172
Reid, Sylvester    223 Santa Marg  Bayside         CA     00082   555-4705
Smith, Pauline     889 Market St.  Fairport        FL     00032   555-1919
Trent, Philip      8809 Iris St.   Applegate       MI     00098   555-2322
Underwood, Al      881 Howell St.  Hillburg        MO     00043   555-6606
--------------------------------------------------------------------------------
Type entry or use  commands                                          -? for Help
```

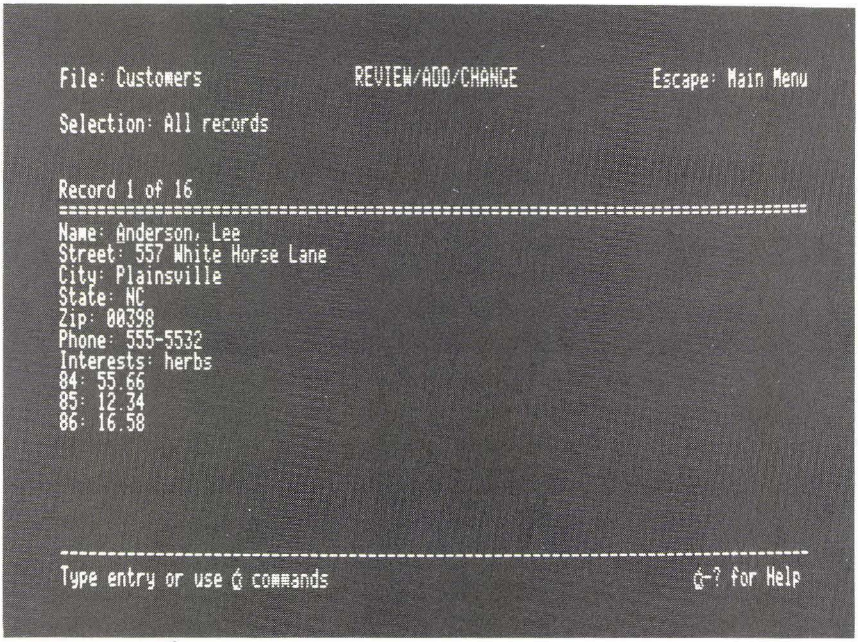
Figure 7 • 5
Data base grid with data filled in

- List all the customers who live in Kansas or Texas and are especially interested in both dwarf herbs and giant squashes.
- Find that one customer who collects unusual kinds of passionflower.
- Find Elmer Oscarson's phone number.

The data base program not only will display the information you want on the screen, it will also print it for you on paper. This ability is especially helpful for lists that include mailing addresses. You can put stick-on mailing labels into the printer and then tell the program to print the names and addresses of your customers, and you've saved yourself the trouble and expense of hiring a mailing service to do the job for you.

Usually a data base program gives you two ways to look at your information: as a list (like the customer list you see above) and as individual records. An individual record from the mail-order plant business might look like Figure 7.6.

Often you can put more categories of information in one record than the Apple IIGS has room to display across the screen in list format. In your customer list, for example, you might want to include a category that tells you whether that person buys vegetables, flowers,



```
File: Customers              REVIEW/ADD/CHANGE              Escape: Main Menu

Selection: All records


Record 1 of 16
===============================================================================
Name: Anderson, Lee
Street: 557 White Horse Lane
City: Plainsville
State: NC
Zip: 00398
Phone: 555-5532
Interests: herbs
84: 55.66
85: 12.34
86: 16.58







-------------------------------------------------------------------------------
Type entry or use  commands                                         -? for Help
```

***Figure* 7 ▪ 6**
Individual record

or both. You might want a category that lets you know if the customer is a business or an individual. The data base program will show you these categories when you look at your list one record at a time; most programs will also let you rearrange your categories in the list format so that you can bring any category you want onto the screen.

Data base programs designed specifically for Apple IIGS are likely to work in a much more flexible way than the kind described above. They'll let you move your information fields around on the screen in the same way you move icons or graphic elements. You'll be able to design your "index cards" exactly the way you want them.

Now maybe you don't run a mail-order plant and seed business out of your house. But, as you can imagine, there are innumerable other uses to which you can put a data base program. You can use it to list . . .

- The subscribers to your newsletter, along with their addresses and whether or not they've paid their dues.
- The employees in your business, along with their social security numbers, hire dates, and salaries.
- The houses available from your real estate agency, with categories for asking price, number of bedrooms, and size of lot.

- All the records, tapes, and compact disks in your music collection, including composers' names and code numbers for finding them on your shelves.
- The people to whom you send Christmas cards, including a category that tells you if they sent one to you last year.
- All your valuables, for insurance purposes.
- All the restaurants in the three surrounding counties, with categories for price, quality, and ethnic persuasion

Don't file your recipes on a data base program, though. For some reason, this suggestion always pops up in discussions of data base programs as if it were a good idea. It's not, unless you want to keep your Apple IIGS on the kitchen counter and poke its buttons when your fingers are covered with flour. In this case (and probably a few others), hang onto your index cards. The old way is occasionally the best way.

Graphics Programs

The Apple IIGS stars in graphics. Even if you don't consider yourself an artist, you'll probably want to treat yourself to a graphics program—partly for the fun of it, and partly because once you have one, you're bound to discover plenty of uses for it.

A graphics program draws pictures for you, or helps you draw them. There are two main kinds of graphics programs: business graphics programs, which turn numbers into charts and graphs; and art or entertainment graphics programs, with which you can do diagrams, designs, and illustrations.

Business Graphics: Business graphics programs are often part of spreadsheet programs, or at least made to be compatible with a spreadsheet program, so that you can copy a row of figures from the spreadsheet and paste it into the graphics program instead of having to retype the numbers.

The advantages of graphics over numbers are obvious: with graphs, it's much easier to spot trends, make comparisons, and make certain points clear to other people (bosses, stockholders, and investors, for example). Drawing a graph on your own can be tricky if you aren't much of an artist, and hiring someone to draw it for you can be expensive. A graphics program gives you good results for a minimum of cost and effort.

How do you want your numbers? As a bar graph, a line graph, or a scatter graph? As a pie chart? As a stacked bar graph, or an area graph? Most business graphics programs give you a choice. In fact, they let you try out all the possibilities one after another and decide which suits your purposes best. Once you've decided, you can place titles on your graph and have the program generate a legend (different kinds of information can be distinguished from each other by color or by pattern).

The capabilities of different business graphics programs vary. Unless you're already using one that you're particularly attached to, you'll want to get a program designed to take advantage of the Apple IIGS's ability to do color and pattern.

Art and Entertainment Graphics: These programs are fun to work with and very popular, probably because they give those legions of people who lament that they "can't even draw a straight line" a chance to produce something that actually looks pretty good—as if created by a real graphic artist.

What kind of graphics program you choose depends on what you want to do. Do you want to make a big banner for your bake sale? Design a logo for your business stationery? Draw a picture of a house, a tree, and the sun? Make a diagram that shows how the current flows through a generator? You can do all of these with graphics programs—but not necessarily with the same program.

One kind of program, which requires that you have a mouse connected to your computer (as you will with your Apple IIGS), lets you draw your own pictures, more or less the way you would with a pencil and paper or with paint and a canvas. (These are often called *paint programs*, in fact.) Little symbols of drawing tools are shown on the screen—things like pencils, rulers, spraycans, and paintbrushes—and you use them to draw straight and curved lines, circles, squares, and ovals, squiggles and zigzags. You can fill your shapes with patterns and move them around on the screen until you've arranged them to your satisfaction, and you can add type to your design in a variety of sizes and styles. With programs like this, it's possible to draw everything from doodles to diagrams—although there's usually no provision for drawing to scale, and a mouse doesn't give you the same artistic control as a pen or pencil. This kind of graphics program—you might call it "freestyle" graphics—is great for children to experiment with and can help adults do things like design logos and stationery, make party invitations, draw maps and diagrams, and illustrate children's stories.

A different kind of graphics program is the *draw program* (as opposed to the paint program). This kind of program has a grid and rulers; it allows you to shrink and stretch the objects you draw and to arrange and rearrange them until you have them in exactly the right spots. A draw program is better for diagrams than is a paint program. You'd use one to do an organization chart, a flow chart, or a floorplan. A paint program is better for pictures, though you'd use a paint program to do things that require shading, intricate curlicues, a free or individual style.

A third kind of graphics program does more of the drawing for you. The program stores all kinds of pictures, fancy borders, and interesting type styles on the disk, and all you have to do is choose among them. You can make greeting cards this way, and long banners with large type, and signs advertising your garage sale. *Print Shop* (from Broderbund Software) is a good example of this kind of program, and there's bound to be a new, improved *Print Shop* for the Apple IIGS. Watch for it.

Communications Programs

Communications programs allow your Apple IIGS to send and receive information through the telephone lines. Let's say you work at home, writing abstracts of scientific articles. If you have a communications program, you don't have to drive to the office to deliver your manuscripts; instead you simply send what you've written over the wires to another computer, where it can be read, edited, and printed out. Communications programs have sparked the dream of an electronic cottage industry, composed of workers who stay home and connect to the office through their computers instead of joining the throngs on the freeway.

Using your computer for communications requires a device called a *modem*. It's a small flat box designed to sit unobtrusively under your telephone. One cable connects it to the jack in the back of the phone, and another connects it to the Apple IIGS. The modem translates the electronic signals that make up computer data into sound signals that can travel over the phone lines.

Here's how computer communications works, in a general way:

1. You create a document of some kind—let's say it's a report—and save it on a disk.
2. You start up your communications program, making sure its settings match the settings of the computer you're sending to.

3. You put the disk with your report on it into the computer.
4. You dial a telephone number and hang up the receiver when you hear the connection signal—usually a long, shrill tone.
5. You give your communications program the command to send your document and the computer "address" of the person to send it to.

The person at the other end can tell by looking at his communication program that there's a message waiting. When he gives the appropriate command, your document appears on his screen.

Until recently, communications programs have been fairly unpleasant to work with. They require you to know the answers to questions about baud rate, "handshake," "echo," and other things that can be confusing if they're not presented clearly, and they ask these questions in such a way that you need a tight grip on the manual and a good dose of luck to send or receive a message successfully. A mistyped code or faulty setting can result in a screen full of gobbledygook, and frustration for the baffled user.

Now the increasing use of graphics in all kinds of computer software has started to affect communications programs. An old-style communications program usually gives you a screen with prompts on it, fairly cryptic prompts as a rule, such as "Enter IDcode," or "Set baud rate." A graphics-based communications program, on the other hand, might give you pictures for prompts instead of words, or use a graphic metaphor to help you understand what to do. (See Figure 2.15 in Chapter 2.)

There are several ways to use a communications program. One is to send messages or documents to one or more people at other computers. In some offices the computers are hooked together into a network, so one person can write a memo—announcing a meeting, for instance, or requesting information—and, with one command, send it to everyone's "mailbox." You don't have to be in the same office to send a message via computer, however. By means of the phone lines, you can communicate more or less instantly with people in other states, even across the country.

Another use for a communications program is to connect with an electronic *bulletin board,* which is actually a collection of notices, messages, and pieces of information stored on a central computer. To put your own information on a bulletin board—anything from lonely hearts ads to news about arthritis research to personal hellos, depending on what kind of bulletin board it is—and to look at information put there by other people, you subscribe to a service and pay, usually, a monthly fee plus a rate per minute of connection time. Plugging into these bulletin boards is a little like being a ham radio operator. You can find out what's going on in Schenectady or Tuscaloosa, strike up friendships with people you've never met, and even carry on a game of chess with someone hundreds of miles away.

A third use for communications programs is research. Enormous amounts of information are stored in databases on mainframe computers. Dialog, for example, stores thousands of references to periodicals, and Lexis is a database for legal research. With a modem and a communications program (plus a fee), you can tap into these databases yourself. As more and more information becomes available in this way, home computer users will be able to find out a lot of what they need to know without going to the library. (These data bases are not always a complete substitute for a library, however; most of them list the titles of articles in books and periodicals and provide abstracts of them, but they don't include the actual text of the article. To get this, you must order it from the communication service or go to the library and find it.)

You also hear about how you can use communications programs to do your banking and your shopping and pay your bills through your computer. These conveniences, however, are still not very widespread. A few banks have set up ways for home computer users to make payments from their accounts to a set list of creditors; a few services offer catalogs from which a shopper can choose goods, placing an order and making the payment all from the computer. For the most part, however, these are things of the future. When enough people own computers and modems, and when communications programs are as easy to use as telephones, you can expect this whole field to take off.

Games

There are shoot-'em-up games, logic games, classic games like chess and checkers, educational games that sneak in instruction about economics or math or fourteenth-century Europe, simulation games that make you feel as if you're flying a plane or driving a race car, adventure games that give you a world full of hazards and surprises to find your way through—an amazing variety of computer entertainment is available. The Apple IIGS, with its spectacular color and high speed, is going to spur game developers to new heights.

Some games require an attachment called a *joystick*, named after the stick a pilot uses to guide a plane. It lets you direct an object across the screen—your spaceship or your little green monster. You might also need game paddles. These let two people compete against each other, firing rockets or fighting dragons or racing cars.

Some people have no use for computer games. Others are addicted to them. You'll know which you are as soon as you try out a few.

Other Kinds of Programs

No personal computer has had more software programs written for it than the Apple II (including the II Plus, the IIe, and the IIc). At last count, there were more than 10,000. Not all these programs are masterpieces of clarity and simplicity; their quality varies widely. If you look hard enough and are willing to put up with occasional rough spots, however, you can probably find a program to do just about anything you want.

It's possible (just barely) that there is no program that exactly meets your needs. Maybe you need to calculate the volume of shark bladders in relation to water pressure at different depths, or write folk tales in 43 different East Asian scripts. If you're really determined to have a computer program that helps you with your very particular task, you might just decide to write it yourself. Writing a computer program, as you probably know, is called *programming*.

Programming

First let's get one thing clear: you never *have* to program if you don't want to. Even if you desperately need that program about shark bladders, you can always get someone else to write it for you. A passion for programming seems likeliest to seize people who are fond of problem solving, logic, and secret codes and who are compulsive enough to sit for hours in front of the screen searching through lines of hieroglyphics for the one out-of-place period or missing zero that's causing the whole program to malfunction. If this doesn't describe you, you can move on to the next section without a twinge of guilt.

If it does, or if you think it might, or if you'd like just to see what programming is all about without committing yourself to making a career of it, you can play around with the programming language called

Applesoft BASIC, which is built into your Apple IIGS. An introductory manual (called *A Touch of Applesoft*) comes with the computer; going through its lessons will help you know whether programming interests you or not.

There are advantages and disadvantages to Applesoft BASIC. One advantage is that it's built in; you don't have to go out and buy a special disk to enable the Apple IIGS to understand it. Another advantage is that it's relatively simple to learn. Its commands are enough like English, and its logic is straightforward enough, that you can make sense of it without a lot of study. Here, for example, is a short program written in BASIC. Each line is numbered (usually program lines are numbered by tens), and the computer executes the lines in numerical order. You can probably tell more or less what this program instructs the computer to do.

```
10 PRINT “How many cats do you have?”
20 INPUT A
30 PRINT “How many dogs do you have?”
40 INPUT B
50 PRINT “The number of animals you have is “A + B”
60 GOTO 10
```

Did you figure it out? This program will print on the screen the words between quotation marks, wait for the answers, assign the numbers you type in to a spot in the computer's memory (A or B), and then add the contents of A and B to give you the answer. The instruction in line 60 tells the program to go back to line 10 and start again.

You can write functioning programs in BASIC almost as soon as you begin to study it because it's so easy to learn. For more advanced, more complex programming, though, it has some disadvantages. Other languages, such as Pascal and some other kinds of BASIC, allow you to write your program in a more organized way, indenting certain sections and labeling them according to what they do. Doing this makes it easier to see at a glance what's going on. If you're going to be a serious programmer, you'll probably favor a structured language, such as Pascal or C.

Some programming languages are designed for specific purposes. COBOL is used in programs for business. FORTRAN is a scientific language. Logo is especially good for graphics. It features something called a “turtle,” which you direct around the screen with commands like Right 50 and Forward 10. The turtle leaves a trail behind it and in this way creates a design according to your directions. Children can learn to use Logo very quickly and, in the process of drawing designs, can also pick up some of the principles of geometry.

All the languages mentioned so far are *high-level languages;* that is, they bear a recognizable relationship to the kind of language that human beings speak. They have English words in them, and their syntax is more or less like that of a sentence: PRINT "Hello," for example, or "Do Procedure Triangle." The language that computers actually speak, however, is several steps removed from these high-level languages. Special programs built into the computer have to interpret languages like BASIC and Pascal before the computer can deal with them, because all that a computer really "understands" is two signals: on and off. The native language of computers is called *machine language*, and it consists entirely of symbols for "on" (meaning a pulse of electricity is flowing through the circuits) and "off" (meaning it isn't). The symbol for "on" is 1, and the symbol for "off" is 0, so that a machine language program looks something like this:

11111010100010010010001001010110111001010101010001011110010,

which shows you rather dramatically why high-level languages were invented in the first place.

To discover whether you like programming, take *A Touch of Applesoft*, or any other beginning programming book and go through the exercises systematically. (If you don't like being systematic, you won't like programming.) You may find that programming is drudgery—worse than figuring income taxes or doing long division. Or, like a surprising number of people of all ages and backgrounds, you may—perhaps to your own astonishment—get hooked.

How Information Gets "Processed"

So you've chosen (or maybe written) the application programs you need. You put your word processing program (or your spreadsheet or database program) into the disk drive and start up the Apple IIGS, and pretty soon you see the "blank page" that is the application's starting point. You type your letter (or your budget, or your list of clients). The words appear on the screen.

But what's really going on? Something devious, you suspect. Something complicated and mysterious, unlike the simple, straightforward operation of pressing a typewriter key and watching it fly up and hit the paper. The words go into the computer and somehow end up getting written on the disk and spit out of the printer. To feel comfortable with your Apple IIGS—in fact, to use it at all—you need to understand how it works.

It's useful to imagine that the computer is filled with little boxes or cells—rather like a beehive. (This is not literally true, but the computer *acts* as if it is.) Each box holds a piece of information, and taken together, the boxes are the computer's memory.

ROM—Permanent Memory

Some boxes contain the programs that tell the computer how to do its basic functions: power up, accept data from the keyboard, display characters on the screen, and so on. These programs are built into the computer; they reside permanently in their part of the memory, which is called *Read Only Memory*, or ROM. You can't put any of your own information into these boxes. The information is there only for the computer to read—hence the name. Beyond knowing that ROM exists and takes up some of the memory space inside your Apple IIGS, you don't have to think about it at all.

RAM—Temporary Memory

Other parts of the computer's memory are like empty boxes (as opposed to the permanently filled boxes of ROM), which wait to have information put into them by the user. When you start up your word processing program, the instructions that tell the computer how to be a word processor go into some of these boxes. The characters of the report or letter or memo you type go into others. This part of the computer's memory is called RAM, for *Random Access Memory* (because the computer has access to any of these boxes at random and lets you put things in and take things out).

The amount of Random Access Memory a computer has is measured in kilobytes, abbreviated as K. A *byte* is a unit of information (you can think of it, for convenience, as the amount of information that goes into one box), and a *kilobyte* is about a thousand bytes. Thus if your computer has 256K of memory, that means it has storage space enough to hold 256,000 units of information. When you use your computer on Monday, you might fill part of that space with a spreadsheet program and part of it with the budget figures you type into the spreadsheet. When you use your computer on Tuesday, you might fill part of it with a word processing program and part of it with the words you type in Chapter 23 of your novel. The computer holds onto the information you give it for as long as you keep the power on. When you turn the power off, the information in RAM goes away.

If you've spent five and a half hours writing Chapter 23 (a chapter especially rich in well-crafted passages and subtle insights), you will naturally be disappointed when it all vanishes at the flick of a switch. Fortunately, there's a way to save your work.

Getting Your Work from the Computer to the Disk

Imagine that you could take the boxes that contain Chapter 23, make a duplicate of their contents, and put the duplicate somewhere outside the computer, where it wouldn't matter whether the power were off or on. This is more or less what you do when you "save" your work onto a disk.

Let's say you've written most of Chapter 23—all but the part where Clive tells Beatrice that she must not go into the locked room in the east wing. You're ready to stop working for the day. When you give your word processing program the command to "save," the computer gets ready to make a copy of all the information that's currently in memory and "write" this copy on the disk. Before it can do this, however, you have to provide a name for what you've written so the computer can label it on the disk and find it again. You type a name—"Chapter23" is a good one—and the computer transfers a copy of your work to the disk, accompanied by little whirring or grunting noises from the disk drive. Now when you quit your word processing program and turn off the computer, it won't matter that Chapter 23 disappears from memory, because an exact copy of it (called a *document*, and sometimes a *file*) is safely stored on the disk.

The next day, when you want to finish up Chapter 23, you have to reverse the process. You have to transfer a copy of what's written on the disk into the memory of the computer, so you can work on it. This is called "getting a document," or, in older computerese, "loading a file."

Getting Your Work from the Disk to the Computer

Depending on what kind of word processing program you're using, there are different ways of telling the Apple IIGS to get "Chapter23" from the disk. We won't go into those ways here. All you need to know is that from within any word processing program (or any other kind of application program) you can give a command that says, "Look on the disk that's in drive 2 (or wherever it is), find the document called 'Chapter23,' and put it in memory." When you give this command, the disk drives chugs a little bit and in a moment Chapter 23 appears on the screen. What's happened is that the computer has made a *copy* of what's on the disk and put that into its memory; what you've saved onto the disk is still there.

Revising Your Work and Saving New Versions of It

Now you get to work on Chapter 23 again. You write the scene between Clive and Beatrice, filling in the ominous undertones, and in an hour or so you're finished.

The situation is now as follows: What's in the memory of the computer is Chapter 23 in its finished form. What's on the disk is Chapter 23 in its unfinished form—the version you saved yesterday. If the power went out right now, everything in memory would disappear—including the suspenseful ending you just wrote—but you'd still have yesterday's version of Chapter 23, because what's on the disk isn't affected by whether the power is on or off.

Most likely, you want to save the new version of Chapter 23. When you give the command to save, the computer makes a copy of what's in its memory, starts to transfer it to the disk, and realizes there's already something called "Chapter23" there. This is important because no two documents on a disk can have the same name. If you save what's in memory now as "Chapter23," that text (what you copied from the disk, with any changes and additions you made) will *replace* what's presently on the disk. The program (in one way or another) gives you a chance to decide if that's really what you want to do. In this case, it is. In some cases, though, it might not be. Suppose on Wednesday you write a version of Chapter 24 in which Beatrice worries all day about the locked room. Then on Thursday you begin to think Beatrice is being unattractively timid. You bring up Chapter 24 and rewrite it to show her stealing the key out of Clive's cigar box and heading determinedly down the corridor.

At this point, once again, there are two versions of your chapter in existence: the one you saved on the disk yesterday as "Chapter24," and the one that's in memory right now. Perhaps you aren't sure which one is better. For the time being, you'd like to keep both of them. Therefore, when you tell the computer to save your work, you *give it a different name*—say, "Chapter24A." The new version is saved on the disk with this new name, and the old version remains intact. On Friday, when you have a little distance on the subject, you can get both of them back and decide which to go with.

An Important Note

Picture this scenario: You start in on Chapter 25 at eight o'clock in the morning. By nine, you've typed four pages. By ten, you're up to page 11. At noon, you're on page 20. At twelve fifteen, there's a sudden power failure, your Apple IIGS goes dead, and four hours of work blinks into oblivion.

To avoid this kind of catastrophe, *save your work as you go along*. Every fifteen minutes or so, give the "save" command, which will write everything that's in memory at that point on the disk. That way, even if there's a power failure, or your three-year-old flips the computer switch, or you make some awful mistake, you haven't lost more than fifteen minutes of work.

To Sum Up

Once you've used your Apple IIGS for a while, all the above will become familiar. In the meantime, you can get along fine if you remember the following:

1. When you type, your words (or numbers, or drawings) go into the computer's temporary memory. (Information does *not* go onto the disk as you type it in.)
2. Everything that's in the computer's temporary memory disappears when you turn the power off.
3. To keep your work from disappearing, you save it onto a disk, giving it a name as you do so. (How you save it depends on the application program you're using.) This creates a "document" (or "file") on your disk.
4. You can bring that document back into the computer's memory any time you want to work on it some more.
5. If you give the work you save the same name as a document that's already on the disk, what's in the computer's memory at the time will *replace* that document. If you save it with a different name, you'll create a different document on the disk, and the original one will be unaffected.
6. It's a good idea to save your work often as you go along.

You Are Now an Expert

Well, maybe not quite. You can never feel like an expert until you've spent a lot of time actually using your computer, trying things out, making mistakes, getting things done. But at this point you should have a good grasp of the basics, and with that tucked under your arm, you can turn back to Chapter 1 and start tackling the specifics of the Apple IIGS.

Appendix—Resources

Users Groups

The ultimate resource is a more experienced Apple user. You can meet a lot of them by joining an Apple Users Group. For information on users groups in your area call the Apple User Group Hotline:

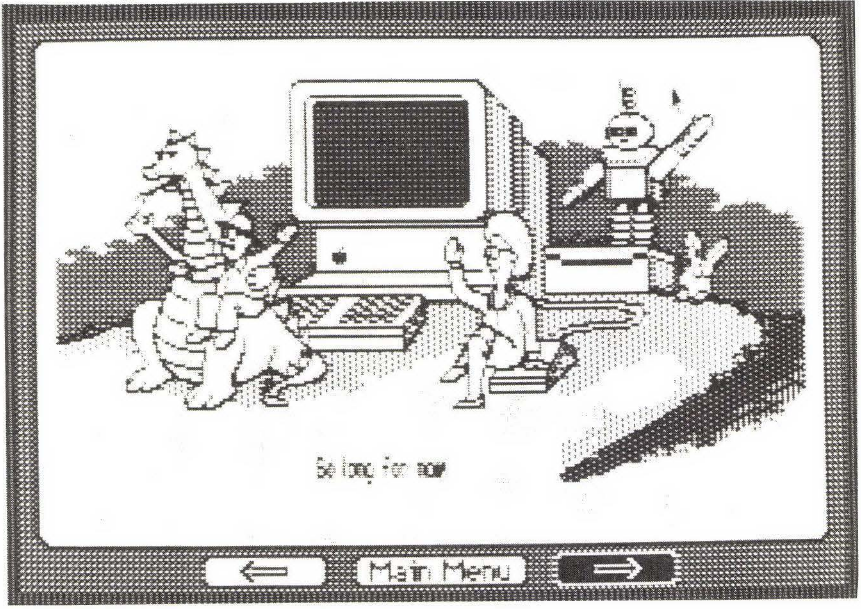
1 (800) 538-9696 EXT. 500

Reference Manuals

The Apple IIGS comes with a setup guide, an owner's guide, a guide to the system disk, and an introduction to Applesoft BASIC. It also comes with a disk-based training course that lets you learn about the Apple IIGS as you use it.

The training disk and the books that come with the Apple IIGS are designed for users of application programs and beginning programmers. The following technical manuals on the Apple IIGS and

Training disk screen



the Apple II family of computers are available separately from your Apple dealer or your local bookstore.

Apple IIGS Technical Introduction
A Programmer's Introduction to the Apple IIGS
Apple IIGS Hardware Reference
Apple IIGS Firmware Reference
Apple IIGS Tools Reference (Part 1 and 2)
Apple IIGS Programmer's Workshop
Apple IIGS Programmer's Workshop Assembler Reference
Apple IIGS Programmer's Workshop C Reference
ProDOS 8 Reference
Apple IIGS ProDOS 16 Reference
Apple Human Interface Guidelines
Applesoft BASIC Programmer's Reference
Applesoft BASIC Programming with ProDOS
Apple IIe Technical Reference

Programming the 65816 by David Eyes and Ron Lichty. New York: Brady Communications (a division of Simon & Schuster), 1986.

The C Programming Language by Brian W. Kernigham and Dennis M. Ritchie. Englewood Cliffs, NJ: Prentice-Hall, 1978.

Software Catalogs

Addison-Wesley Book of Apple Software
Addison-Wesley Publishing Company
Reading, MA 01867
(617) 944-3700

The Book of Apple Software
11223 South Hindry Avenue
Los Angeles, CA 90045
(213) 410-9466

Whole Earth Software Catalog
27 Gate Five Road
Sausalito, CA 94965
(415) 332-4335

Magazines

A+
Box 2965
Boulder, CO 80322
(800) 525-0643

The Apple II Review (quarterly)
Redgate Communications Corp.
3381 Ocean Drive
Vero Beach, FL 32963
(305) 231-6904

BYTE
Box 597
Martinsville, NJ 08836-9956

Family Computing
Box 2508
Boulder, CO 80321

inCider
CW Communications/
Peterborough
Box 911
Farmingdale, NY 11737

Some of the more popular magazines on personal computing.

CLASSROOM COMPUTER LEARNING
ARTIFICIAL INTELLIGENCE: What's In It For Educators?
THE COMPUTING TEACHER
TEACHING and computers
A+ Apple II/Macintosh
ELECTRONIC LEARNING
Computer
THE APPLE II MAGAZINE
inCider
SUCCESS WITH APPLEWORKS
Making Money with Word Processing
★ More Original Programs for your Apple //! ★
Inside — A complete, new Apple language!
nibble
THE REFERENCE FOR APPLE COMPUTING
Turtle Basic
Logo power from Applesoft
PERSONAL FINANCE ISSUE
FOR APPLE II USERS
II COMPUTING
Computerized Banking
Investment Interview: Charles Schwab
Roundup of Financial Software
Convert Hi-Res to Print Shop
Mountain Skiing II

InfoWorld (weekly)
Box 1018
Southeastern, PA 19398-9982
(800) 227-8365

nibble
45 Winthrop Street
Concord, MA 01742
(617) 371-1660

Personal Computing
Box 2941
Boulder, CO 80321
(800) 525-0643

Reading List

Some great books have been written about personal computers and programmers. Most of the oldtimers you'll read about in these books are only in their 30's today and still pushing the personal computer revolution forward. Here are some books to get you started:

Hackers: Heroes of the Computer Revolution
Steven Levy
Anchor Press/Doubleday
1984

Fire in the Valley
Paul Freiberger, Michael Swaine
Osborne/McGraw-Hill
1984

Little Kingdom: The Private Story of Apple Computer
Michael Moritz
Morrow and Company
1984

Programmers at Work: Interviews
Susan Lammers
Microsoft Press
1986

Psychology of Computer Programming
Gerald M. Weinberg
Van Nostrand
1971

Soul of a New Machine
Tracy Kidder
Little, Brown
1981

Software Publishers

Here are the addresses and phone numbers of the software publishers mentioned in this book:

Advanced Ideas, Inc.
(Wizard of Words)
2902 San Pablo Avenue
Berkeley, CA 94702
(415) 526-9100

Apple Computer, Inc.
(AppleWorks, Access II, AppleWriter, Apple Logo, Backup II, Instant Pascal, MousePaint, QuickFile II)
20525 Mariani Avenue
Cupertino, CA 95014
(800) 538-9696

Arktronics Corp.
(Jane)
520 East Liberty
Ann Arbor, MI 48104
(313) 769-7253

BPI Systems
(BPI Accounting Series)
3001 Bee Cave Road
Austin, TX 78746
(512) 328-5434 (800) 531-5252

Brøderbund Software, Inc.
(Arcade Machine, Print Shop, DazzleDraw)
17 Paul Drive
San Rafael, CA 94903
(415) 479-1170

CBS Software
(Mastering the SAT, Charles Goren: Learning Bridge)
One Fawcett Place
Greenwich, CT 06836
(203) 622-2500

CompuServe Inc.
(CompuServe)
Box L-477
Columbus, OH 43260
(614) 457-0802 (800) 848-8199

Datamost Inc
(Cavern Creatures)
21040 Nordhoff Street
Chatsworth, CA 91311
(818) 709-1202 (800) 692-1649

Davidson and Associates
(Spell It! Word Attack, Math Blaster, Speed Reader II)
3135 Kashiwa Street
Torrance, CA 90505
(213) 534-4070 (800) 556-6141

DesignWare, Inc.
(CryptoCube)
185 Berry Street
San Francisco, CA 94107
(415) 546-1866 (800) 572-2272

Developmental Learning Materials (DLM)
(Alien Action, Alligator Alley, Dragon Mix)
200 East Bethany
Allen, TX 75002
(214) 727-3346 (800) 520-4747

Earthware Computer Services
(Star Search, Volcanoes)
Box 30039
Eugene, OR 97403
(503) 344-3383

Educational Activities, Inc.

(Comp-U-Solve)
Box 392
Freeport, NY 11520
(516) 223-4666 (800) 645-3739

Electronic Arts
(One on One, Pinball Construction Set, Deluxe Series, Music Construction Set)
1820 Gateway Drive
San Mateo, CA 94404
(415) 571-7171 (800) 448-8822

Field Publications (formerly Xerox Educational Publications)
(Stickybear Series)
245 Long Hill Road
Middletown, CT 06457
(203) 638-2400 (800) 852-5000

Harcourt Brace Jovanovich, Inc.
(Computer SAT Preparation)
1250 Sixth Avenue
San Diego, CA 92101
(619) 231-6616

Information Appliance
(Swyftcard)
1014 Hamilton Court
Palo Alto, CA 94301
(415) 328-5160

International Solutions
(MouseDesk)
910 West Maude Avenue
Sunnyvale, CA 94086
(408) 773-0443

Koala Technologies Corp.
(Koala Painter)
2065 Junction Avenue
San Jose, CA 95131
(408) 946-4483 (408) 986-8866

The Learning Company
(Rocky's Boots, Gertrude's Secrets, Moptown, Magic Spells, Reader Rabbit, Bumble Games)
545 Middlefield Road #170
Menlo Park, CA 94025
(415) 328-5410

Manzanita Software Systems, Inc.
(BusinessWorks)
1 Sierra Gate Plaza 200A
Roseville, CA 95678
(916) 781-3880

MegaHaus Corp.
(MegaWorks, ReportWorks)
5703 Oberlin Drive
San Diego, CA 92121
(619) 450-1230

Microsoft Corp.
(Multiplan)
10700 Northup Way
Bellevue, WA 98004
(206) 882-8080

PBI Software
(Jeeves)
1111 Triton Drive Suite 201
Foster City, CA 94404
(415) 349-8765 (800) 843-5722

Pinpoint Publishing
(Pinpoint)
Box 13323
Oakland, CA 94661
(415) 654-3050 (800) 633-2252

Quark, Inc.
(Word Juggler II, Catalyst 3.0)
2525 West Evans, Suite 220
Denver, CO 80219
(303) 934-2211 (800) 543-7711

Scarborough Systems, Inc.
(Mastertype)
55 South Broadway
Tarrytown, NY 10591
(914) 332-4545

Scholastic Software, Inc.
(Story Tree)
730 Broadway
New York, NY 10003
(212) 505-3000 (800) 325-6149

Simon & Schuster, Inc.,
(Typing Tutor III)
1230 Avenue of the Americas
New York, NY 10020

Software Publishing Corp.
(PFS:File, PFS:Write,
PFS:Report, PFS:Plan,
PFS:Graph)
1901 Landings Drive
Box 7210
Mountain View, CA 94043
(415) 962-8910

Source Telecomputing Corp.
(The Source)
1616 Anderson Road
McLean, VA 22102
(800) 336-3366

Spinnaker Software Corp.
(Alphabet Zoo, Kidwriter)
One Kendall Square
Cambridge, MA 02139
(617) 494-1200

SubLOGIC Corp.
(Flight Simulator II)
713 Edgebrook Drive
Champaign, IL 61820
(217) 359-8482 (800) 637-4983

Terrapin, Inc.
(Terrapin Logo)
222 Third Street
Cambridge, MA 02142
(617) 492-8816

TruTec
(Person-to-Person)
1700 Solano Avenue
Berkeley, CA 94707
(415) 525-4901

United Software Industries
(ASCII Express, MouseTalk)
10651 East Bethany Drive
Aurora, CO 80014
(303) 671-6655

Vernier Software
(Projectiles II)
2920 S.W. 89th Street
Portland, OR 97225
(503) 297-5317

VIP Technologies Corp.
(VIP Professional)
132 Aero Camino
Goleta, CA 93117
(805) 968-4045

Wadsworth Electronic
Publishing Co.
(Algebra Arcade)
8 Davis Drive
Belmont, Ca 94002
(415) 595-2350

Glossary

ACIA: A chip on the Super Serial Card.

active window: Some applications let you have several windows open at one time so you can see more than one document or directory. However, you can work only with the contents of the foremost window. That window where action occurs is the active window.

American Simplified Keyboard: See *Dvorak*.

analog RGB color monitor: RGB stands for "red green blue" and it describes a technology for generating color video. Analog is one of two methods of generating RGB video (the other is digital). The word *analog* describes a device that can accept measurable voltages or frequencies as input, in contrast to a *digital* device, which accepts only discrete numbers as input.

Apple Desktop Bus: A port for connecting Apple Desktop Bus devices like the Apple Desktop Bus keyboard and the Apple Desktop Bus mouse. It's called a bus rather than a port, because several devices can use the same information transportation system to get input to the computer—that is, they all daisy-chain off the same port. "Desktop" is used to describe the devices that use this port, because they don't share much in common besides their proximity on the desktop.

Apple key: The key to the left of the space bar. It is used in combination with other keys to send special instructions to the computer. It's also called the *Command key* and is sometimes represented with a propeller icon—the other symbol on the key. On earlier models of the Apple II, the Apple key was called the Open Apple key to distinguish it from the Solid Apple key (a black Apple). On the Apple IIGS, the Solid Apple icon has been replaced with the word Option.

Apple II Plus: A member of the Apple II family, no longer manufactured by Apple. It has uppercase-only keyboard; it was ahead of its time in the 1970's.

Apple IIc: A compact model of the Apple II with 128K, a choice of 40- or 80-column display, a built-in disk drive, and support for standard devices like a printer, a modem, another disk drive, and a mouse. Unlike other Apple II's, it is a closed system. You can't open the lid and add other devices by plugging in interface cards.

Apple IIe: An Apple II with 64K of memory expandable to 128K and a 40-column display adaptable to 80 columns. It improved on its predecessor, the Apple II Plus, by featuring an uppercase and lowercase keyboard and a new operating system that could work with larger storage devices.

Apple IIGS: A computer with a 16-bit processor, 256K RAM, 128K ROM, color, Apple II family compatibility, super-high resolution graphics (640 by 200 pixels), tools that support a Macintosh-like interface, a built-in clock, a 15-voice sound chip, volume control and selectable pitch (so you can choose the musical note that will accompany error messages), support for two serial devices, up to four disk drives (two 3.5-inch drives and two 5.25-inch drives), RGB and NTSC color monitors, a mouse, and for a joystick or hand controls, slots for connecting a memory card or interface cards for devices that have special interface requirements.

AppleLink: Apple Computer's internal communication network. Dealers can access AppleLink for answers to technical questions and for product compatibility information.

Applesoft BASIC: A dialect of BASIC that's built into the Apple IIGS and all other models of the Apple II. Because it's built in, you can write programs without first loading the programming language from a disk, as you have to do with other dialects of BASIC and other languages like Logo and Pascal.

AppleTalk network: A means of connecting Apple computers and peripherals by cable to permit many users to exchange messages and share peripherals.

AppleWorks: A set of integrated applications developed by Rupert Lissner just about the time the Apple IIc was introduced. Notable for the consistency of the user interface, jargon-free prompts, and on-line instruction.

application program: A program that lets you do something useful or interesting with your computer: type things, keep records, plan budgets, draw pictures, and so on.

arrow keys: Keys in the lower right corner of the keyboard that control the movement of the cursor.

ASCII (American Standard Code for Information Interchange): A code that defines every letter (uppercase and lowercase), number, and punctuation mark as a distinct string of seven 0's and 1's. (There are also ASCII codes for carriage returns and line feeds.) It's one of the few standards that all computer makers adhere to, so it's the form you convert a document to when you send it over the phone lines to another computer.

assembly language: A programming language that is similar to the language of 0's and 1's that the computer understands without a translator. Because they require less translation, assembly language programs run faster and are more efficient than programs written in languages that more closely resemble human language, such as BASIC and Pascal.

auto repeat: Something that happens over and over again once it's set in motion. The keys on the keyboard are described as auto repeat keys because the character you press will keep repeating on the screen as long as you hold it down.

backspace: To move the cursor (insertion point) to the left, usually over letters you've already typed. Backspacing with the Delete key usually erases the character to the left. Backspacing with the left arrow usually moves the cursor over the character without deleting it.

baud rate: The speed at which devices exchange information. The two devices must agree on the baud rate. By using the Control Panel Program, you can change the rate at which information is sent to printers and modems connected to a serial port.

BASIC: Acronym for Beginner's All-purpose Symbolic Instruction Code. A programming language designed for first-time programmers that lets you write instructions to the computer in English-like words and phrases.

binary: A numbering system with just two digits: 0 and 1. Larger numbers are formed by combining 0's and 1's. Computers use the binary system for calculations.

bit: A contraction of the words *binary* and *digit*. As the name suggests, a binary digit is a 0 or a 1. The computer uses 0's and 1's to do calculations and also uses strings of 0's and 1's to represent letters, numbers, and punctuation marks. See *ASCII*.

bits per second: The number of bits transmitted from one device to another every second. Bits per second, abbreviated bps, is used interchangeably with baud rate.

boot: To start up the computer. Starting up is called booting because when you turn on the computer's power switch, the computer looks inside itself (in ROM) for startup instructions and then carries them out. That is, it pulls itself up by its own *boot*straps rather than relying on an outside program for instructions. Apple is trying to phase out the term *boot* in favor of the more descriptive term *startup,* but many users like the expression *boot* so well that they apply it to all sorts of startup situations—booting the car, booting the VCR . . .

buffer: A place in memory where information is stored until the computer or a peripheral device is ready to use it. The keyboard buffer stores keypresses when the microprocessor is busy and isn't ready to process them. A printer buffer, which can be on memory chips built into a printer or on an interface card, stores text until the printer is ready for it. Storing text in the print buffer instead of in the memory of the computer allows you to use the computer for other things while your document is being printed.

bug: A problem. Something that doesn't work the way it's supposed to.

bulletin board: A computer that posts information and allows you to download (receive over the phone lines) copies of uncopyrighted software. Access to most bulletin boards is free. You can find phone numbers of local bulletin board systems from your local computer dealer or users group.

business graphics: Programs that create different kinds of charts and graphs from spreadsheet figures or from numbers you enter using the graphics program.

button: An option in a dialog box. You click anywhere inside a button to select it. Button could also refer to the pad on top of the mouse that you press to select and drag.

byte: The set of eight bits required to form a character.

C: A programming language. It is designed to be easy to port from one brand of computer to another because it avoids addressing the idiosyncracies of any particular machine.

Caps Lock: A key you can press when you want every letter you type to be capitalized. When Caps Lock is down, you don't need to press Shift to get capital letters. Caps Lock affects only letter keys, not number keys or keys with punctuation marks. (In other words, to get the dollar sign over the 4, you have to press Shift, even when Caps Lock is down.)

card: See *interface card.*

carriage return: (CR) An ASCII character that tells the computer or printer to put the next character on the left margin. It does not say to go the next line. That instruction comes from the linefeed ASCII character.

CD ROM: Acronym for Compact Disk Read Only Memory. Very large-capacity disks containing data recorded on them through laser technology. They're called ROM because a CD player connected to a computer can read what's on them but can't write anything to them.

cell: The intersection of a row and a column in a spreadsheet. You can put words, numbers, formulas, or functions in a cell.

character: A generic term for a letter, number, or punctuation mark. Any symbol that appears on the screen or on a printed page.

character generator: The program in ROM that determines what symbol to put on the screen and in memory in response to a keypress.

character set: The symbols you can generate by pressing every key on the keyboard. The Apple IIGS can generate the character sets for eight different international keyboards as well as for the Dvorak keyboard layout. You choose the character set you want to use with the Control Panel Program.

chip: Jargon for integrated circuit. A sheet of silicon with circuits etched on top and pins on the bottom that connect to electronic pathways on the circuit board so information can get from one chip to others on the circuit board. The pins make the chips look like mechanical centipedes.

choose: To pick a command, usually from a pull-down menu. In mouse-based programs, you first select what you want to work on, then you choose what you want to do with it (for example, print it).

circuit board: See *main circuit board.*

classic application: An application developed for an earlier model of the Apple II. An application that runs on the Apple IIGS but doesn't take advantage of its new features.

clicking: Pressing and releasing the button on top of the mouse. Clicking on a word or icon selects it for some action.

clip art: Illustrations (usually by a professional artist) that you can cut and paste into your documents.

clipboard: A special part of memory that holds what you just cut until you're ready to paste it. The clipboard holds only one clipping at a time. If you cut a second item before pasting the first, the first thing you cut is lost.

clock rate: The speed of the processor.

close box: The square in the upper left corner of a window. You can close a document or a directory by clicking on the close box. Doing this has the same effect as choosing the Close command from the File menu.

COBOL: A programming language prevalent in the business world.

column: A vertical bar. A column is one character wide when it refers to the number of characters that can fit on the screen. It usually has an adjustable width when it refers to a vertical lane in a spreadsheet.

commands: Your instructions to the application program. For example, you give the print command when you want the application to print your document. You give the save command when you want the application to save your document. The manual that comes with the application will tell you what commands the program responds to. In mouse-based programs you issue commands by choosing from pull-down menus. In keyboard-based applications, you issue commands by typing the numbers that are in front of menu items, or by pressing the Apple, Option, or Control key in combination with one of the other keys on the keyboard.

communications software: Applications that make it more convenient, useful, or economical to access other personal computers and information services over the phone lines.

compatibility: Harmony among the peripherals and applications connected to your computer.

composite: See *NTSC composite monitor.*

computer: A machine that can manipulate the numbers 0 and 1 very quickly. By expressing everything in terms of 0's and 1's, the computer can process numbers, words, sounds, and pictures to increase your productivity and creativity.

computer system: The computer and all the peripherals attached to it.

configuration: The way you have things set up. It could apply to the way you have your computer set up or to the way you have your serial port set up to send and receive information.

connect time: The amount of time you spend accessing another computer over the phone lines. The elapsed time between when you log on and when you log off. The time you're billed for.

construction set: An application that gives you the tools and raw materials to build something. A music construction set, for example, gives you notes and electronic music paper for building a musical score. A pinball construction set gives you the levers, bells, and bumpers to build a pinball game.

Control: A key that you press in combination with another key to send an instruction to an application program. It's called the Control key because you use it to control the operation of the program.

Control Panel Program: A built-in program that lets you set the time and date of the built-in clock and calendar; control the pitch and volume of the built-in speaker; configure the serial ports for different printers and information services; control the color or shade of text, background, and screen border in text-based applications; choose between a 40- and 80-column display; and more. Control Panel settings are stored in a special kind of battery-powered RAM that saves your preferences even after the power has been turned off. This means you can change the settings anytime you want, but you don't have to set your preferences more than once.

controller card: See *disk controller card.*

co-processor: A processor that works along with or instead of the processor that's built into your computer. It comes on a card. See *Z80 card.*

copy protect: Something programmers do to make it difficult to duplicate their software.

CP/M: Stands for Control Program/Monitor. CP/M was one of the first operating systems for personal computers, consequently there are a lot of CP/M-based programs floating around, many of them public-domain (free) programs. To use CP/M programs on an Apple IIGS, you need a Z80 card.

crash: Just what you'd expect it to mean—a problem that brings the computer system to its knees. It usually requires restarting the application and losing whatever you were working on.

cursor: A symbol that marks your place on the screen and shows where your next action will occur. In some programs, the cursor is a flashing box or underline, but it could be anything. In Logo, it's a turtle (actually a triangle that's called a turtle).

cut: To remove from a document and place on the clipboard, usually with the intention of pasting it somewhere else.

cut and paste: To remove from a document and insert somewhere else in that or another document.

daisywheel printer: See *letter-quality printer.*

data: Information. It usually refers to information that hasn't been processed yet, like the information that sits in a database waiting for your instructions on how to sort and arrange it.

data bits: When information is sent to serial devices, each character is sent as a string of seven or eight bits. These are called data bits.

data base: An application that gives you different ways to arrange and sort lists of information. Regardless of how you enter your records initially, you can have the computer sort your records alphabetically (A to Z, or Z to A) or numerically (low to high, or high to low). Besides arranging your data in different ways, a database application can search for all the records that meet a certain criteria—for example, all the restaurants in Baton Rouge that serve Cajun food and accept the American Express Card.

DCD handshake: Something devices do in preparation for sending and receiving information. DCD stands for Data Carrier Detect. You can choose to use this handshaking protocol or not by using the Control Panel Program.

default: A setting or response that an application will use if you don't supply an alternative. Apple is trying to phase out this term in favor of *preset* or *standard response.*

Delete: A key that you press to erase the last character you typed—the character just to the left of the insertion point. There was no Delete key on early Apple II's, so programs designed for other models of the Apple II won't erase when you press Delete. They'll tell you some other way to fix mistakes.

Desk Accessories: Miniature application programs that you can use without leaving your main application. With a calculator as a desktop accessory, for example, you can do a quick calculation and paste the result into a word processing document.

desktop: A metaphor that is synonymous with memory in a program like AppleWorks or with the screen in mouse-based applications.

developer: Someone who writes software or designs peripherals for a living.

device: See *peripheral device.*

dialog box: A rectangle with words inside. It is displayed on top of whatever you're working on when the program needs to have a word with you. A dialog box usually asks you to make choices by clicking in labeled buttons inside the box.

digital: A digit is a number. Digital refers to something that is expressed in numbers. Compare *analog.*

digitizer: A machine that converts analog data—a photograph or a sound—to numbers that the computer can manipulate.

directory: A catalog or list of the documents on a disk. The directory shows the name, the size in K's, the location on the disk, and the date the document was created or last revised. If you save documents in subdirectories, only the subdirectory names appear in the directory. To see the documents in a subdirectory, you ask the program for a subdirectory listing the same way you asked for a directory listing. Applications use directories to locate documents you tell them to load. You can use the directory to see what's on a particular disk, to find out the exact name of a document you know is on the disk, or to find out the date you last revised a document.

disk: A sheet of plastic coated with magnetic oxide (the same stuff that's on cassette tape) and covered with a plastic shell in the case of 3.5-inch disks or cardboard in the case of 5.25-inch disks. You use them to save programs you write and documents you create with application programs.

disk controller card: An interface card that tells the computer how to exchange information with up to two disk drives. You don't need to use a disk controller card with the Apple IIGS because you can attach up to four disk drives (two 3.5-inch and two 5.25-inch) to the disk drive port—the information that controls the drives is in ROM.

disk drive: A device that (a) reads information recorded on disks, copies it, and sends it to the memory of the computer and (b) writes (records) information on disks so you can get it later. There are three types of disk drives: 3.5-inch drives, 5.25-inch drives, and hard disk drives. How information is written to disks and where it is copied into memory are determined by the disk operating system used by the application.

disk operating system: See *operating system.*

display: What you see on the screen.

document: The body of information you create with an application. It could be a song, a drawing, a list, a chart, or the chapter of a book. Also called a file.

DOS 3.3: DOS, an acronym for Disk Operating System, was the first Apple II operating system. Version 3.3 is the most advanced version of that operating system. It's a perfectly good operating system and there are a lot of applications that use it. Its only limitation, and the reason that Apple developed ProDOS as a replacement for it, is that DOS 3.3 works only with 5.25-inch disks. It can't manage large-capacity disks like 3.5-inch and hard disks.

dot matrix: A printer that forms characters and pictures by using patterns of dots. Some dot matrix printers can print in color when you use color ribbons and software that supports color printing.

double-click: To press and release the mouse button twice in rapid succession.

double high resolution: A graphics mode that displays information on the screen by assigning different values to the pixels in a 560 by 192 dot grid. It's the graphics equivalent of 80 column mode. (In 80-column text mode, characters are formed from a 7 by 8 dot matrix. 7 dots times 80 = 560.)

Down Arrow: One of four cursor control keys. Pressing Down Arrow moves the cursor one line down on the display.

download: To have a document sent to your computer from another computer over the phone lines.

drag: To move the mouse while holding the mouse button down. You can drag across a block of text to select it for deleting, or copying, or some other reason. Or you can select an icon and drag it to a trash icon to discard it, to a disk icon to copy it, or just to a new place on the screen for the esthetics of it.

drive: See *disk drive.*

DSR: Stands for Data Set Ready. DSR is a handshaking signal—a way for devices to regulate the flow of information between them.

DTR: Stands for Data Terminal Ready. DTR is a handshaking signal—a way for devices to regulate the flow of information between them.

duplex: See *full duplex* and *half duplex.*

Dvorak: The name of an efficient keyboard layout. Named after its inventor, August Dvorak. On the Dvorak keyboard, the most-often used keys are located on the home row. The Dvorak keyboard is also called the American Simplified Keyboard.

echo: A copy of a transmitted document that appears on the transmittting computer's screen as well as on the receiving computer's screen. The echo can be sent by the receiving computer as a way of confirming that the message arrived, or by the sending computer to make sure it's being sent correctly.

Edit menu: A menu in mouse-based programs that has commands for cutting, pasting, and copying text.

8-bit processor: A processor that processes information 8 bits at a time. Compare 16-bit processor.

80-column card: A card that made it possible for the Apple IIe to send characters to the screen to fit in a 80- by 24-line grid instead of in the standard 40- by 24- line grid. (In 80-column mode, you can fit twice as many characters per line as in 40-column mode.) You don't need an 80-column card with the Apple IIGS, because 80-column support is built in.

80-column display: Information that is displayed on the screen in an 80-column by 24-line grid. Compare *40-column display.*

eject: To remove a disk from a disk drive.

electronic mail: Messages that are sent over the phone lines using communications software and a modem.

e-mail: See *electronic mail.*

error message: Words on the screen put there to alert you to some problem. Either the application can't carry out your last instruction, or it doesn't understand what you asked it to do.

even parity: A way of checking to see if a transmitted message got through intact. The transmitting device adds up the seven 0's and 1's in each character and adds either a 0 or 1 to the data string to make the total number of bits add up to an even number. If the total still adds up to an even number on the receiving end, the message got through okay.

extended 80-column card: An 80-column card that also adds 64K of RAM to the 64K built into the Apple IIe. You don't need such a card in the Apple IIGS, because you have 80-column support built in and 256K to start with.

Fast Key Speed: A Control Panel option that lets you double the rate at which a pressed key repeats on the display when you hold the key down while pressing Control.

fast RAM: In Apple IIGS, memory that is accessed at 2.8 megahertz. Compare *slow RAM.*

field: One category of information in a database. If the database is an auto parts inventory, one field might be "part number" and another field might be "supplier."

file: A body of information that is written to a disk. Files can be either documents or programs. When you save a file on a disk, you give it a filename. When you want to retrieve that information, you ask for it by its filename.

File menu: A menu in mouse-based applications that lets you open, close, save, and print documents.

filename: The name of a file. Compare *pathname.*

Finder: A mouse-based program selector and utilities application designed especially for the Apple IIGS.

firmware: Programs which are stored in ROM.

firmware entry point: A memory location that is designated as the "legal" way for developers to access the routines in ROM from their programs.

5.25-inch disks: Disks designed to be used with 5.25-inch drives. The disks are coated with magnetic oxide (the same sort of material used for cassette tape) and permanently sealed inside protective cardboard jackets. You can see the disk itself through an oval cutout in the jacket, and that's the opening through which the drive accesses the disk. 5.25-inch disks have a storage capacity of 143K (about 70 pages).

5.25-inch drive: A drive that reads from and writes to 5.25-inch disks. This is the first kind of drive that was available for Apple II computers.

folder: A subdirectory. Something you can use to keep related documents together in a directory so you can find what you need quickly.

font: A type style such as Helvetica. Many graphics-based applications give you a choice of fonts.

footer: Text that appears at the bottom of every page or every other page of a document.

format: Something that has to be done to blank disks before information can be saved on them. There is a command for formatting disks on the Apple IIGS System Disk. Different disk operating systems require different formats.

formula: An equation that you assign to a cell in a spreadsheet. By using spreadsheet locations in formulas, you can change numbers in those other locations without revising your formulas.

FORTRAN: A programming language used for scientific purposes.

40-column display: Text that fits in a 40-column by 24-line grid.

full duplex: Describes a modem that echoes messages back to the sending device.

function: Formula that is built into a spreadsheet application. There are typically functions that calculate square root, average, and the number of items in a row or column.

function keys: Keys that have a special purpose in a given application. Some applications use the keys on the numeric keypad as function keys.

graphics: A way of displaying information on the screen. In graphics mode, images are sent to the screen as patterns of dots. Different graphics modes use different concentrations of dots. See *super high resolution, double high resolution, high resolution,* and *low resolution.*

graphics applications: Programs for creating charts and graphs, posters and personalized stationery, or freestyle illustrations. See *business graphics* and *clip art.*

graphics mode: A way of displaying information on the screen. In graphics mode, the screen is made up of pixels, each of which can be assigned a different color value.

graphics tablet: A peripheral that, with software designed for the purpose, captures what you draw on a special pad with a special pen and displays it on the screen.

half duplex: Describes a modem that doesn't echo messages back to the sending device.

hand controls: A pair of peripherals with knobs that you turn to control the movement of a cursor, and with one or more buttons that you press for special effects. In games, the "cursor" that responds to the hand controls is a character or object in the game, and the button fires rockets, shoots basketballs, and the like.

handshake signals: Information exchanged by a peripheral and a computer to start, stop, and generally regulate the flow of information between them.

hard disk: A very-large-capacity disk that is permanently sealed inside a hard disk drive. By copying your applications to a hard disk, you can change applications without changing disks. Also, by saving documents on a hard disk, you can access more information without doing any disk-swapping. Applications that work with large bodies of information (for example, accounting programs and sophisticated database applications) sometimes require a hard disk.

hard disk drive: The mechanism that reads information from and writes information to a hard disk. The terms "hard disk" and "hard disk drive" are used interchangeably because, with most hard disk drives, you can't remove the disk from the drive.

hardware: Computer parts and peripherals—the stuff you can see and touch. Compare *software*.

header: Text that appears at the top of every page or every other page of a document.

hertz: Frequency unit equal to a cycle per second. See megahertz.

high resolution: A graphics mode in which there are 280 by 192 dots that can be assigned 6 different color values.

High Speed Mouse: A Control Panel option that allows you to double the distance the pointer moves for a given movement of the mouse on your desk.

highlight: To select something for some action.

home control devices: Peripherals that you can attach to electrical outlets and appliances so you can control your lamps, coffeemaker, alarms, etc. using your computer.

I-beam: The cursor that is used for editing text.

icon: A picture that represents something. Mouse-based programs use icons so you can point to pictures of what you want to work on and what you want to do rather than remembering commands and typing them at the keyboard.

information service: A computerized database that you can subscribe to and access using your communications program and a modem. You can get the latest news and stock quotes and can exchange messages and mail with other subscribers.

initialize: See *format.*

input: Information that travels into the computer (resulting from the keys you press and the moves you make with the mouse).

insertion point: A vertical line on the screen that shows where text will appear when you type. Compare *cursor.*

integrated applications: Applications that, ideally, have a common interface, making it easy for you to move information from a document created with one type of application to a document created with another. The most commonly integrated applications are word processing, database, spreadsheet, graphics, and communications.

integrated circuit: See *chip.*

interface: A translator. Compare *user interface, serial interface,* and *parallel interface.*

interface card: A small circuit board containing the instructions that tell the computer how to communicate (interface) with a particular peripheral. (The instructions are stored on chips on the card, just as the instructions that tell the computer how to communicate with devices attached to ports are stored on chips on the main circuit board of the Apple IIGS.) Some devices come with interface cards (a hard disk drive, for example). Other devices require a particular kind of card (a parallel printer, for example, requires a parallel interface card). Compare *port.*

interrupt: Something that demands the processor's attention when it's carrying out some other activity.

inverse characters: If the normal video display mode is white characters on a black background, inverse characters are black on a white background.

I/O: input/output. The process of getting information into or out of the computer.

joystick: A device modeled after the joystick in an airplane. Moving the joystick forward, back, left, and right makes the cursor move a corresponding way on the screen. In game programs, the "cursor" that responds to joystick moves is a character or object in the game.

K: See *kilobyte.*

keyboard: A peripheral that looks like a typewriter keyboard. It gives you a way of creating documents that involve text. Before the mouse came along, the keyboard was also the main way of sending instructions to a program.

keyboard buffer: Part of memory where key presses are stored when the computer is busy and can't process them.

keycap extractor: Special tweezers that can be used to remove keycaps from stems.

keypad: See *numeric keypad.*

kilobyte: 1024 bytes.

laser printer: A printer that burns images on paper using laser technology.

launch: To start up a program by using the Finder.

Left Arrow: One of four cursor-control keys. Pressing Left Arrow moves the cursor one character to the left. In some applications, pressing Left Arrow erases as it backspaces.

letter-quality printer: A printer that hammers metal letters on a printhead into a ribbon and onto paper like a typewriter. The printhead often looks like a daisy, so this kind of printer is also called a daisywheel printer. LQP's, as they are often abbreviated, are good for correspondence, but not good at all for printing graphics.

light pen: A peripheral that you use with special software to point to choices displayed on the screen.

line number: A number in front of a BASIC program line. Line numbers tell the computer the order in which to carry out the instructions in a program.

linefeed: An ASCII character that tells the computer to move the cursor (and the printer to move the printhead) to the next line.

Logo: A programming language especially suited to children because it allows them to learn good programming habits and skills by writing programs that draw pictures.

low resolution: A graphics mode in which there are 40 by 48 blocks to which you can assign 16 different color values.

Macintosh: A personal computer that popularized the mouse interface—the point, click, and drag approach to controlling applications.

mail merge: An application that takes information from a database—names and addresses, for example—and inserts them at specified locations in a word processing document. Lets you create personalized form letters.

main circuit board: The plastic square that covers the bottom of the computer case and holds the integrated circuits (chips) that process and retain information as well as the wires that interconnect the chips. The main circuit board is also called a *motherboard* because it supports smaller circuit boards called interface cards.

main menu: The top-level menu in a keyboard-based application.

mass storage device: A peripheral, like a hard disk drive, that can store lots and lots of data.

megabyte: A million bytes. The capacity of hard disks and the memory you can add to the Apple IIGS with a memory expansion card are measured in megabytes. See *byte.*

megahertz: Frequency unit abbreviated MHz equals a million cycles per second. The speed of the Apple IIGS microprocessor is measured in megahertz. It can operate at either 1 or 2.8 megahertz. Compare *hertz.*

memory: Chips that store programs and documents electronically. Because the information is stored electronically, it is lost when you turn off the power. Before you can use the application and document again, you have to load them from a disk

memory expansion card: A card you can put in the Apple IIGS's memory expansion slot that adds 256K, 512K, or 1 megabyte of RAM to the 256K that's built into the Apple IIGS. New programs use the additional memory as an extension of the built-in RAM. With older programs you can use the memory on the card as a RAM disk.

menu: A list of choices.

menu bar: A horizontal listing of menu titles at the top of the screen in mouse-based programs.

menu title: The name of a pull-down menu in a mouse-based program. To see what's on a given menu, press and hold down the mouse button when the pointer is on the menu title.

MHz: See *megahertz.*

microprocessor: Usually referred to as Central Processing Unit. The Apple IIGS uses the 65816 microprocessor.

mode: An environment in which specific rules prevail—where key presses, for example, have different consequences than they do in other environments. Some programs have one mode for entering text and another mode for editing text: in the text-entering mode, pressing D makes a D appear on the screen, and in editing mode, pressing D might delete a character. Programs that use modes will tell you how to get into them, and out of them, and how to behave when you're in one.

modem: A device that changes digital computer signals (0's and 1's) into audio signals that can be sent over the phone lines to another computer. A modem on the receiving end changes the audio signals back to digital signals the computer can process. See *communications software, information services,* and *e-mail.*

monitor: A device that displays instructions for using an application program and displays the document you create with the program.

monochrome monitor: A monitor that displays information in one color against a black background—the monitor equivalent of a black and white TV, except that you can choose white on black, green on black, or amber on black.

mouse: A device that some applications use as a way for users to control application programs. In mouse-based applications, when you move the mouse across your desk, a pointer moves in a corresponding way across the screen. You make choices in mouse-based applications by pointing to words and pictures representing your selections and pressing the button on top of the mouse.

mouse-based: Describes applications that let you use the mouse to control them.

mouse button: A rectangle on the front of the mouse that you press when you want to select in mouse-based programs.

MouseText: Special characters, like check marks, an apple symbol, and the like, that are used in text-based mouse applications.

MS-DOS: The operating system used by IBM personal computers.

music synthesizer: Hardware and software that you can use with your computer to generate the sounds of a variety of musical instruments. It makes it possible to play back digitized recordings or your own electronic compositions to see how they will sound when they are performed by a band or orchestra.

network: Two or more computers linked together and able to exchange information and share peripherals.

NTSC composite monitor: NTSC stands for National TV Standards Committee. An NTSC çomposite monitor is one that is set up to receive video signals according to the standards set by the NTSC. Composite color monitors receive color signals over a single wire. Compare *analog RGB color monitor.*

numeric keypad: The number keys on the right side of the IIGS's detached keyboard. The keys are arranged like the keys on an adding machine to make it easier to enter numerical information in spreadsheets and databases.

odd parity: A way of checking to see if a transmitted message got through intact. The transmitting device adds up the seven 0's and 1's in each character and adds either a 0 or a 1 to the data string to make the total number of bits add up to an odd number. If the total still adds up to an odd number on the receiving end, the message got through okay.

offline: The line, in this case, is the cable or telephone wire that connects two computers. When you're offline, you're not in a position to exchange information with the computer on the other end of the cable. Being online usually costs money, so it's generally a good idea to compose messages offline and use your online time for transmitting and receiving them.

online help: Built-in instructions for using an online program.

Open Apple: The key with the Apple. Some applications refer to it as the Open Apple key, because there are two apple keys on earlier models of the Apple II: an outline of an Apple and a colored-in or "Solid Apple." The Apple key is used in combination with other keys to initiate some action. For example, you might be told to press Open Apple and P to print your document.

open architecture: The concept of expandability. A computer that gives you easy access to the main circuit board so you can add memory and interface cards and basically draw on the computer's horsepower in any way you want.

operating system: Instructions that tell the computer how to save documents on disks, retrieve documents from disks, and put them into memory where the microprocessor can rearrange them according to your instructions. Programmers don't have to write their own operating system instructions. They can use one of the three operating systems developed for the Apple II: ProDOS, Pascal, or DOS 3.3. The preferred operating system for the Apple IIGS is ProDOS because it takes the best advantage of the Apple IIGS's features. It is also the most efficient manager of the space on mass storage devices like 3.5-inch disks and hard disks.

Option key: A key that you press in combination with another key for some special effect. On earlier models of the Apple II this key was labeled with a black Apple symbol.

output: Information that travels out of the computer to peripheral devices like monitors and printers.

palette: A board that artists use for mixing paint. With the Apple IIGS you can have 16 colors on your electronic palette at one time and you can select those 16 from 4096 possible colors.

parallel interface: One way that the computer can exchange information with a peripheral device. Devices that use a parallel interface are connected to the computer by parallel wires inside the cable. The data bits that make up each character travel side by side down the wires, in contrast to serial interface, where they travel single file down one wire.

parity: A form of error checking. Transmitting and receiving devices can agree to use odd parity, even parity, or no parity to ensure that the bits that make up a message don't get garbled during transmission.

Pascal: A high-level programming language which can also function as an operating system.

paste: To insert the contents of the clipboard at the spot marked by the cursor.

pathname: The complete name of a document, starting with the name of the disk it's on, the subdirectory it's in (if it's in one), and the name of the document. It begins with a slash, and each component of the name is separated with a slash. If you use subdirectories as a way of organizing documents on a disk, you must use a pathname to tell the computer where to find documents you want loaded and where to put documents you want saved. The pathname /SPAIN/MADRID/CHEAP.HOTELS, tells the computer to save the document on the disk named SPAIN in the subdirectory named MADRID and to name the document CHEAP.HOTELS.

peripheral device: Something you attach to the computer, like a printer or a modem. Monitors and disk drives are called peripherals too, even though they aren't exactly peripheral to operating the computer.

pixel: A loose contraction of the words "picture element." A pixel is one of the dots that makes up the image on the display.

plotter: A device that reproduces computer documents on paper with colored felt-tip pens. Plotters produce paper copies of pictures and patterns created with Logo programs, as well as charts and graphs created with business graphics programs, architectural drawings, and documents created with other graphics-oriented applications.

point: To position the pointer on something by using the mouse.

pointer: An arrow or other symbol that moves on the screen in response to the way you move the mouse on your desktop.

port: A place to plug in a peripheral device. The instructions that tell the computer how to interact with devices for which there are ports built into the computer. (Compare *slot*.)

prefix: The first part of a pathname. By setting a prefix, you don't have to type that part of the pathname.

print: To make a paper copy of a document created with the computer.

printer: A device that turns electronic documents into paper documents. It's one of the least peripheral of the peripherals you can get for a computer, because while it's possible to operate a computer without a printer, it's not practical to mail and distribute all your documents on disks. Paper still has its place in the electronic age.

printer options: Codes you can insert in a document that instruct the printer to underline a particular section or put it in bold or italics. Unless a program is graphics-based, you can't get these special effects on the screen, but by embedding codes in your document, you can get them from your printer.

ProDOS: An acronym for Professional Disk Operating System. It is the primary Apple II operating system. It supports subdirectories and large-capacity storage devices like 3.5-inch disks and hard disks.

program: Instructions that control the operation of the computer.

program selector: An application that lets you move from one application to another without restarting the computer.

programmer: Someone who writes instructions for the computer. If you make money at it, they start calling you a developer.

prompt: An instruction on the screen that tells you what to do next, or a character that asks you to type something.

public-domain software: Programs that have not been copyrighted. Many programmers write for the fun and challenge of it and distribute their efforts online through computerized bulletin boards and users groups.

pull-down menu: A list of choices in a mouse-based program. It's called a pull-down menu because you don't see it until you pull it down by pointing to the menu title and pressing the mouse button.

RAM disk: Available memory on the memory expansion card. It's called a RAM disk because the computer treats it like a disk in a disk drive. You can save documents on it, copy programs to it, and so on. But it's like other RAM in that what you save on the RAM disk is lost when you turn off the power.

read: What the disk drive does when it gets information from a disk and copies it into the memory of the computer.

record: All of the information about one person, place, or thing in your database. Compare *field.*

report: Something you generate from the data in a data base. It can be everything in the database, but usually it's a subset of the records or the fields. You can generate hundreds of different reports without affecting the database as a whole.

Reset: A key you press in combination with other keys to restart the computer. On the detachable Apple IIGS keyboard, the Reset key is marked with a triangle. On the Apple IIe keyboard, it's labeled Reset.

resolution: The degree of detail on the display. In low resolution, images are drawn out of relatively few large blocks. In super high resolution they are drawn out of many tiny dots.

Return: A key that you press in many applications to confirm a choice or to tell an application that you're ready to proceed. In word processing applications, pressing Return inserts a return character (abbreviated CR for carriage return) that tells the cursor (or the printer) to end that line and go to the left margin. (It doesn't tell the cursor or printer to start a new line. That instruction comes from a linefeed character.)

RF modulator: A device that converts computer signals into a form that can be displayed on a regular TV set. The resulting picture isn't as sharp as you get with a computer monitor, but it's adequate for programs displayed in 40-column mode.

RGB monitor: A type of color monitor that receives video signals over three wires (one for red, one for green, one for blue) instead of receiving red, green, and blue signals scrambled onto one wire.

Right Arrow: One of four cursor-control keys. Pressing it makes the cursor move one character position to the right.

Rocky's Boots: A popular educational program for the Apple II from The Learning Company.

ROM disk: Card for the memory expansion slot that contains applications permanently recorded in ROM, just as the programs that control the basic operation of the computer are permanently recorded in ROM on the main circuit board. If you have such a card installed, you can designate it as your startup device using the Control Panel program.

run: To use an application. What a processor does when it starts carrying out the instructions in a program.

save: To store information temporarily, in the memory of the computer, or permanently on a disk.

SCC: A chip in the Apple IIGS that handles serial interface and supports the AppleTalk network.

screen: The monitor's display area. The place where information from the program is projected.

screen dump: Getting a printed copy of what's on the screen.

scroll: To move a different part of a document or directory into the window on the screen so you can see it or work on it.

scroll arrow: The arrow at either end of the scroll bar. By clicking on the arrow, you cause the document to scroll one line or column at a time in the direction in which the arrow is pointing.

scroll bar: The "elevator shaft" on the right side of the screen in mouse-based applications that represents the length of your document. By sliding the scroll box along the scroll bar, you can move different parts of a document into view. Spreadsheets and other wide documents also have a scroll bar along the bottom of the screen so you can move left and right through your document.

scroll box: A box that you drag along the scroll bar to move different parts of your document into view. When the scroll box is at the top of the scroll bar, you see the first part of your document. When it's in the middle you see the middle of your document.

SCSI: An acronym for Small Computer System Interface, pronounced "scuzzy." You can use a SCSI interface card to connect one or more hard disks or other SCSI devices to the Apple IIGS.

search and replace: To find one or every occurrence of a word or phrase in a document and exchange it for another.

select: To pick a document, a folder, a disk, or a block of text in order to do something with it. In mouse-based applications, you select something, then you choose what you want to do with it.

serial interface: One way that the computer and peripheral devices can exchange information. Devices that use serial interface send information one bit at a time along a single wire. Compare *parallel interface.*

serial port: A port for connecting serial devices (devices that use a serial interface) to the computer. The Apple IIGS has two serial ports. One is marked with a printer icon; the other is marked with a phone icon.

shadowing: Copying certain parts of classic applications from "fast RAM" into "slow RAM," a scheme that enables old programs to run faster on the Apple IIGS than on earlier models of the Apple II without affecting compatibility.

simulation: An imitation of a real-life situation. A flight simulator, for example, gives you the simulated experience of flying a plane. Some training programs simulate applications. That is, they give you a feel for using the actual application but provide coaching and remediation, and protect you from making mistakes you might make in the actual program.

16-bit processor: A processor that processes information 16 bits at a time. The Apple IIGS uses the 65C816, a 16-bit processor that can directly address 16 megabytes of memory.

65C816: The 16-bit processor in the Apple IIGS that is able to emulate the 6502 processor in other Apple II's so you can run programs developed for earlier models of the

Apple II as well as programs that take advantage of the special features of the 65C816.

size box: A box in the lower right corner of a window that lets you shrink and expand the display area. Drag the box up and left to shrink the window; down and right to expand it.

slot: Long, narrow connector on the Apple IIGS's main circuit board for connecting interface cards. Computers with slots are more adaptable than computers that let you connect devices only by using ports. The instructions for using various devices can be stored on the card, so there's almost no limit to the variety of devices you can connect to the computer through slots. There are seven general-purpose slots on the Apple IIGS's main circuit board and one special slot for adding a memory expansion card. Compare *port.*

slow RAM: Memory that is accessed at a frequency rate of 1 megahertz, the system speed of earlier models of the Apple II. Compare *fast RAM.*

smart port: The disk drive port when it's being used with an intelligent device like a 3.5-inch drive—devices that use the "smart port interface" that's built into the computer.

software: Instructions that tell the computer what to do. Another word for *application* or *program.*

software catalog: A book that describes, reviews, and ranks applications. Most software catalogs are devoted to a particular brand of computer and they are usually organized by category of application: games, word processors, spreadsheets, and so on.

speech synthesizer: Software and hardware that allows the Apple IIGS to imitate the human voice.

spreadsheet: An application that lets you manipulate numbers and words on an electronic grid. The intersection of a row and a column on the grid is called a *cell.* You can define one cell as the sum of certain other cells. In this way it is possible to change values in the grid and let the application handle the recalculation for you.

startup: What the computer does when it loads an application into memory and starts running it.

stop bits: Bits that mark the end of a string of data bits in serial data exchange.

structured language: A programming language that requires that a program be written in orderly, well-defined modules.

subdirectory: An electronic folder that you create as a way of keeping related documents together. (They aren't really kept together on the disk—they could be stored from one end to the other—but they are listed together when you ask to see what's in the subdirectory.) By using subdirectories, you don't have to look through every document on a disk searching for the one you want. You can open the relevant subdirectory and look through a much shorter list.

super high resolution: A graphics mode that displays information on the screen by assigning different values to the pixels in a 640 by 200 dot grid for 4-color programs or in a 320 by 200 dot grid for 16-color programs.

Super Serial Card: A serial interface card available from Apple Computer, Inc.

system: Short for computer system. It refers to your computer and everything you have connected to it.

system configuration: The way you have your computer system set up.

system disk: A disk that acts both as a program selector for launching applications, and as a utility disk for copying, deleting, renaming, and otherwise managing the documents stored on disks. The Apple IIGS System Disk utilities work with documents that use any of the three Apple II operating systems, but it's effective as a program selector only with ProDOS-based applications.

System Speed: A Control Panel option that lets you choose to operate the Apple IIGS at its top speed (up to 2.8 megahertz) or at 1 megahertz, the top speed possible on other models of the Apple II. Most programs run best at the top speed, but the timing of certain animated programs could be thrown off by the faster speed. If that's the case, you can use the Control Panel to set the speed back to 1 megahertz.

System Utilities: A disk containing ProDOS, DOS 3.3, and Pascal utilities.

tape backup: One method of duplicating the information on hard disks. It works like a cassette recorder, but faster. Tape storage isn't practical as a primary storage method because access is sequential rather than random, but it's a cheap, fast, and reliable backup system.

telecommunications: Exchanging information over the phone lines.

template: An electronic overlay that makes it easier to use spreadsheets and database applications. Instead of making up your own formulas and categories, you fill in the blanks in a template.

text generator: A program in ROM that tells the computer hôw to display characters on the screen.

text mode: One of two ways to display information on the screen. (The other way is *graphics mode.*) In text mode, an application uses the computer's built-in text generator to send characters to the screen. In 40-column text mode, the characters fit in a 40-column by 24-line grid. In 80-column text mode, the characters fit in an 80-column by 24-line grid.

thermal transfer printer: A printer that uses heat to get characters onto paper.

3.5-inch disks: Diskettes coated with magnetic oxide with a radius of 3.5 inches. The 3.5-inch disks are protected by a hard plastic square casing. A metal shutter on the casing slides away when the drive needs to access the disk.

3.5-inch drive: A drive for reading from and writing to 3.5-inch disks.

title bar: A strip along the top of a window that shows the name of the disk, folder, or document. You move windows by dragging them with the pointer anywhere in the title bar. If the title bar is highlighted, that means the window is active.

trash: An icon that represents electronic oblivion. You drag a document to the trash when you want to erase it from a disk.

troubleshooting: Figuring out what's wrong so you can fix it.

Up Arrow: One of four cursor-control keys. Pressing it moves the cursor up one line.

user interface: The way a computer program communicates with a human.

users group: An organization of computer owners or enthusiasts who get together to share or show off programs they've written and to discuss the relative merits of new peripherals and applications available for their particular brand of computer. You can find out about Apple users groups in your area by calling 1 (800) 538-9696 ext.500.

utilities: Programs for copying, deleting, and renaming documents and disks and for moving documents from one disk to another. The Apple IIGS System Disk has utilities especially designed for working with Apple IIGS disks and documents, but there are many other utility disks you can use. Be aware that some utilities are designed to work only with disks that use a certain operating system (DOS 3.3, Pascal, or ProDOS), and others, like the System Disk, are designed to work with disks using any operating system.

voice input device: A peripheral that, when used with particular applications, lets you send information to the computer by speaking instead of by typing.

volume: Used interchangeably with the word *disk,* but it's not that media-specific. (You can have more than one volume on a hard disk.) It's a place for storing files that is identified by name (volume name) rather than by its location in a particular disk drive.

window: A rectangular area on the screen that frames a document or directory.

word processing application: An application for writing. Allows you to delete, insert, move, and otherwise manipulate text. Lets you make changes to a document without retyping the parts that haven't changed.

word wrap: When you reach the end of a line and you're in the middle of a word, applications that have a word wrap feature move the cursor and the half-completed word to the next line for you.

write: What a disk drive does when it copies information from the memory of the computer onto a disk.

WYSIWYG: An acronym that stands for What You See Is What You Get. When a word processing program displays on the screen the actual fonts, margins, and spacing you'll get when you print out the document, it's described as a WYSIWYG word processor.

XOFF: An ASCII character that tells the transmitting device to stop sending information.

XON: An ASCII character that tells the transmitting device to resume sending information.

Z80 card: An interface card containing the Z80 microprocessor. The Apple IIGS uses a 65C816 microprocessor. By putting a Z80 card in one of the slots in the Apple IIGS, you can use programs developed for computers that use the Z80 microprocessor. See *CP/M.* (CP/M is the operating system used by computers with a Z80 microprocessor.)

zoom box: A box on the right side of the title bar on some windows. Clicking once in the zoom box enlarges the window to its maximum size. Clicking again reduces it to its original size.

Index

About the Authors

Jeanne DuPrau is a senior writer in Apple's User Education Department. She has written demonstration and training disks for the Apple IIe, the Apple IIc, the Apple III, and the Apple IIGS, as well as for Lisa and Macintosh. She has written for textbook companies, newspapers, and magazines, and has published a book about adoption.

Molly Tyson is a senior writer in Apple's User Education Department. She has written 35 Apple manuals, including the Apple IIc Owner's Guide and the Apple IIGS Owner's Guide. Her writing has also appeared in *The New York Times* and in *Runner, Self, Highlights for Children, Popular Computing,* and *Women's Sports* magazines.

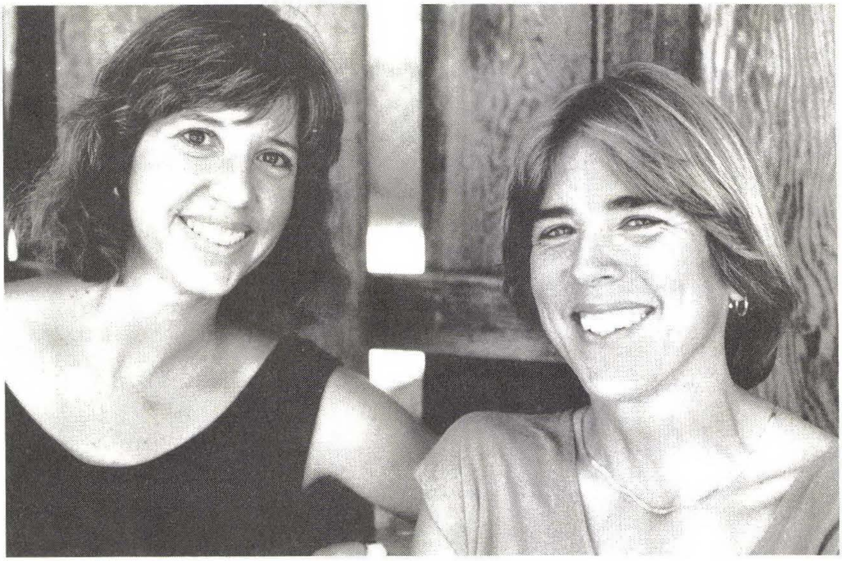


Photo by Sande Low.

Trademark Information

Apple, the Apple logo, Lisa, ProDOS, and Quick File are registered trademarks.

Apple Desktop Bus, AppleLink, AppleTalk, AppleWorks, Mac, MacPaint, MacWrite, AppleWriter, ImageWriter, LaserWriter, MousePaint, Super Serial Card, ProFile, and Apple II Forever are trademarks of Apple Computer, Inc.

Macintosh is a trademark of Macintosh Laboratory, Inc., licensed to Apple Computer, Inc.

Algebra Arcade is a trademark of Wadsworth Electronic Publishing Co.

Alphabet Zoo is a trademark of Spinnaker Software Corp.

American Express is a registered trademark.

Apropos is a trademark of Apropos Software, Inc.

Atari is a registered trademark of Atari Corporation.

Bank Street Writer is a trademark of Bank Street College/ Broderbund, Inc.

Bumble Games is a trademark of The Learning Company.

BusinessWorks is a trademark of Manzanita Software Sytems, Inc.

Catalyst is a trademark of Quark Incorporated.

Coca-Cola is a registered trademark of The Coca-Cola Company.

Commodore is a trademark of Commodore Electronics, Ltd.

CompuServe is a registerd trademark of CompuServe Corporation, an H & R Block Company.

Cypto Cube is a trademark of DesignWare.

Dazzle Draw is a trademark of Broderbund, Inc.

Deluxe Music Construction Set, DeluxePaint, DeluxePrint, and DeluxeVideo are trademarks of Electronic Arts.

DesignWare is a registered trademark.

Electronic Arts is a trademark of Electronic Arts.

Ford is a registered trademark of Ford Motor Company.

Fortune is a registered trademark of Time Inc.

Gertrude's Secrets is a trademark of The Learning Company.

Gumby and Pokey are registered trademarks.

Hayes is a trademark of Hayes Microcomputer Products, Inc.

Hewlett Packard is a trademark of Hewlett Packard Company.

IBM is a registered trademark of International Business Machines Corporation.

Instant Pascal is a registered trademark of Think Technologies.

Jane is a trademark of Arktronics.

Jeeves is a trademark of PBI Software.

KidWriter is a trademark of Spinnaker Software Corp.

Koala is a trademark of Koala Technologies Corp.

The Learning Company is a trademark of The Learning Company.

Lotus is a registered trademark of Lotus Development Corporation.

Magic Spells is a trademark of The Learning Company.

Mastertype is a trademark of Scarborough Systems, Inc.

Mastering the SAT is a trademark of Lightning Software.

Math Blaster is a trademark of Davidson and Associates, Inc.

MegaWorks is a trademark of Megahaus.

Microsoft is a registered trademark of Microsoft Corporation.

Mirage is a trademark of Ensoniq.

MoneyWorks is a trademark of Apropos Software, Inc.

Moptown is a trademark of The Learning Company.

MS-DOS is a registered trademark of Microsoft Corporation.

Multiplan is a registered trademark of Microsoft Corporation.

Music Construction Set is a trademark of Electronic Arts.

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

Pepsi Cola is a registered trademark of the Pepsi Cola Company.

Person to Person is a trademark of TruTec Software.

PFS: is a registered trademark of Software Publishing Corporation.

Pinpoint is a trademark of Pinpoint Publishing.

Print Shop is a trademark of Broderbund, Inc.

Reader Rabbit is a trademark of The Learning Company.

Reportworks is a trademark of Megahaus.

Rocky's Boots is a trademark of The Learning Company.

Rolodex is a registered trademark of Rolodex Corporation.

Speed Reader II is a trademark of Davidson and Associates, Inc.

Spinnaker is a registered trademark of Spinnaker Software Corp.

Stickybear is a trademark of Optimum Resource, Inc.

Story Tree is a trademark of Scholastic, Inc.

THE SOURCE is a servicemark of Source Telecomputing Corporation, a subsidiary of Reader's Digest Association, Inc.

ST is a trademark of Atari Corp.

Swyftcard is a trademark of Information Appliance, Inc.

Terrapin Logo is a registered trademark of Terrapin, Inc.

TRS-80 is a registered trademark of Tandy Corporation.

Tru-Tec is a trademark of Tru-Tec Software.

Typing Tutor III is a trademark of Kriya Systems, Inc.

Visicalc is a registered trademark of Software Arts Products, Inc.

Windows is a registered trademark of Microsoft Corporation.

Word Attack! is a trademark of Davidson and Associates, Inc.

Word Juggler is a trademark of Quark Incorporated.

WordStar is a registered trademark of MicroPro International Corporation.

XEROX is a registered trademark.

Zaxxon is a trademark of Saga Enterprises, Inc.

Z80 is a registered trademark of Zilog, Inc.

Produced with DeluxePaint. Photo courtesy Electronic Arts.

The 16-Bit APPLE II!

Spectacular color and sound, high-speed processing, a mouse/desktop user interface and compatibility with more than 10,000 programs already developed for its older Apple II siblings—that's the beauty of the new Apple IIGS!

And here's the book that gives both novice and experienced users the complete inside story on all aspects of this revolutionary machine—its conception and development, its unique features, its broad capabilities and its startling potential.

Written by Apple insiders, **THE APPLE IIGS BOOK** reveals intimate details of the machine's development and design. Interviews with Steve Wozniak, Dan Hillman and other members of the development team, as well as with third-party software producers, provide initial insight into why certain features were favored over others. The book then provides a detailed user's reference to the components that make the IIGS so unique and powerful. After a look at basic computer concepts and operational procedures, you'll examine the IIGS's new 16-bit environment, which opens the door to faster, more sophisticated software. You'll learn how to make the most of the IIGS's dramatically improved screen resolution and sound definition. You'll also take a detailed look at the mouse user's interface and the Finder, features first introduced on the Macintosh.

This book explains:

- History—How the IIGS came to be.
- Design Considerations—The decision to use the powerful new 16-bit 65816 CPU
- DOS 3.3, ProDOS and Pascal—Three different operating systems for the IIGS.
- The Mouse—The point and click device that speeds data access.
- The Finder—A Macintosh-like utility that promotes user friendliness.
- Graphics and Sound—The super hi-res and the super hi-fi.
- Software—What programs are available.
- All this and much more.

Jeanne DuPrau and Molly Tyson are senior writers in Apple's User Education department. Both also write for national publications.

34359

0 76783 01895

N 0-553-34359-9>1895

34359-9 ■ IN U.S. $18.95 (IN CANADA $22.95) ■ BANTAM COMPUTER BOOKS