VOL. 1 No. 1 May 1984

COMPUTERDIGEST

NEW KIND OF MAGAZINE FOR ELECTRONICS PROFESSIONALS

LISA — PC — MACINTOSH — PEANUT

Quick comparison of these four popular computer systems

UPGRADING BUDGET PRINTERS...

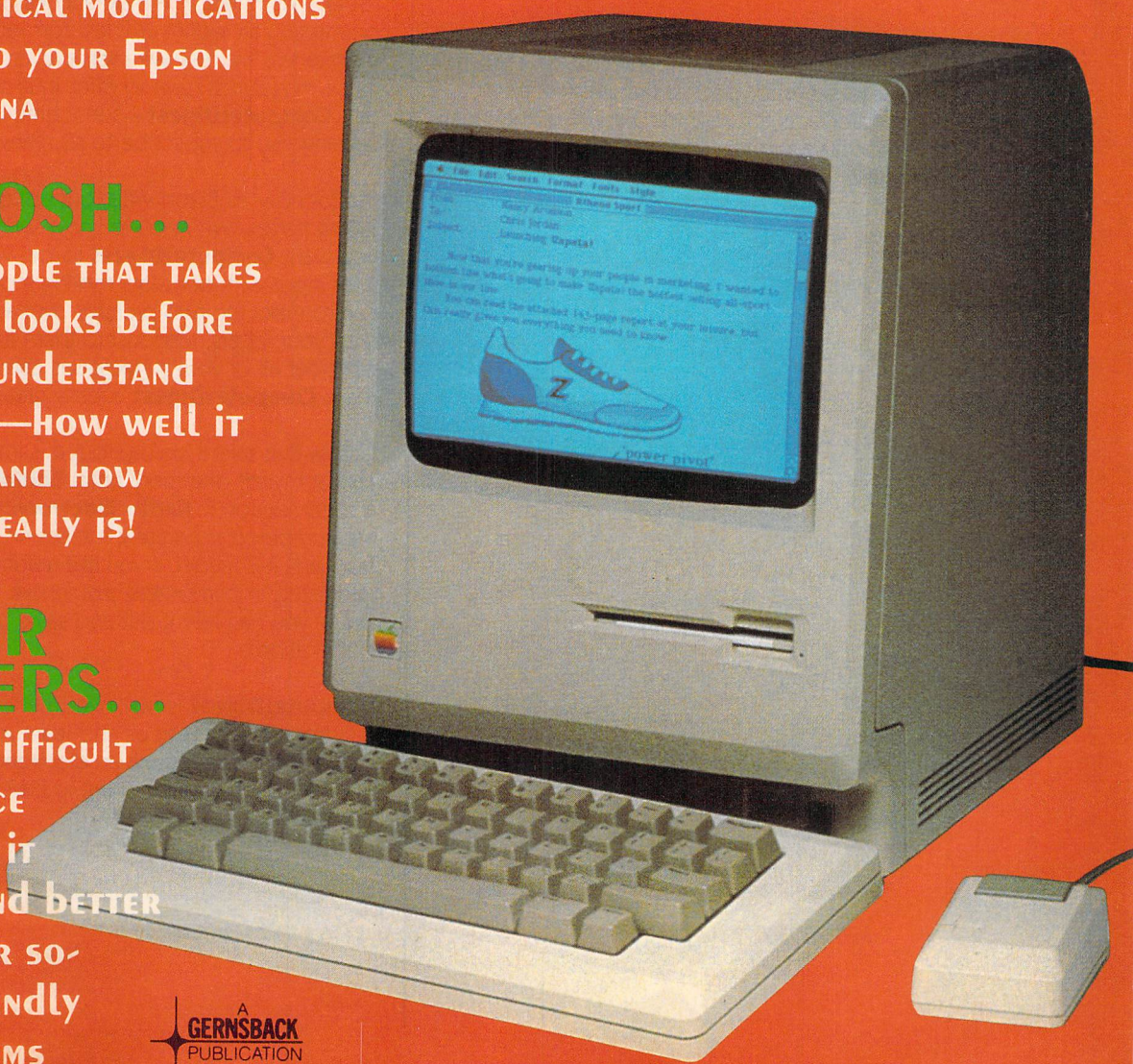
Graphics and tractor feeds are easy and inexpensive to add to your system. Here's a bunch of practical modifications you can make to your Epson or Smith-Corona

MACINTOSH...

A shiny new Apple that takes several careful looks before you can really understand what it can do—how well it can function—and how inexpensive it really is!

CP/M FOR BEGINNERS...

It really isn't difficult to use! And once you know how, it can be easier and better than many other so-called user-friendly operating systems



A GERNSBACK PUBLICATION

CONTENTS

Vol. 1 No. 1 May 1984

4 Lisa, PC, Macintosh, And Peanut
A look at four popular systems and how they stack up against each other. **Byron Wells**

6 Macintosh: A New Variety Of Apple
Apple's new computer has created a bit of excitement. Find out why, and exactly what it is that makes the Macintosh so special in our in-depth review of that new machine. **Marc Stern**

9 CP/M For Beginners
Though it has scared off many a beginner, CP/M is one of the best, and easiest-to-use operating systems available. **Herb Friedman**

12 Upgrading Budget Printers
Just because you own a budget printer doesn't mean that you have to settle for budget features. Here's a look at some products for upgrading some of the most popular inexpensive printers on the market.
Herb Friedman

2 From The Publisher

3 Editorial

FROM THE PUBLISHER

Welcome to **ComputerDigest**. This is Volume 1, Number 1 of what we intend to make a long series of continuing publications. Our purpose is to bring you something that you cannot get elsewhere currently: a broad look at what is happening in computers from the viewpoint of the electronics professional.

That means that **ComputerDigest** is not written for the typical computer operator or hobbyist. It is written for the electronics professional—engineer, technician, or hobbyist—who has his feet solidly placed in the field of electronics. To sum it up—the typical **Radio-Electronics** reader.

Because of that slant, that orientation, we are different! However, we do not replace any other computer magazine. In fact, if we are successful in obtaining permission to reprint articles from other computer magazines, we may well serve as an introduction to what those magazines have to offer. In fact, you may blame us—**ComputerDigest**—some time in the future for causing you to subscribe to so many other publications.

But for now, as Byron Wels, our editor points out on his facing page—we need to know what kind of magazine you want **ComputerDigest** to be. We need to compare your input with our plan, making modifications where we can to make the two pictures overlap. So do write. We'll read your letters, respond to them as quickly as possible, consider all of your suggestions carefully, and go on to make **ComputerDigest** a really great publication.

I know I've said it before, but it is important so I will repeat myself one more time. **ComputerDigest** is an added section in **Radio-Electronics**. Note that the pages are numbered separately. It takes no editorial space away from any R-E reader. If you like reading R-E, but are not interested in computers, just ignore the section. If you do like it, you can tear it out and save it as a separate publication.

ComputerDigest is our investment in the future of electronics. Join with us and learn more about this existing exploding field. You are an electronics professional today, and that means you're going to have to know about computers.

Sincerely,

LARRY STECKLER
PUBLISHER

EDITORIAL

Another Computer Magazine!

No! This one is different. Really different!

Considering the numbers of computer magazines available today, you could go broke trying to read them all. What's more, the amount of time you would use up just going through all of them would keep you from getting anyplace near your computer.

That's why **ComputerDigest** is here.

We do read all the magazines in the field, and we read them very carefully. When we see an article that we feel is important to you, we pull it out and put it aside. Then, once a final selection is made, each of those articles is read again and carefully edited to carve off the fat and leave only that which is necessary to make it readable and informative.

When we can't get permission to reprint an article, we assign one of our own authors to do a story that fully covers the subject in our own style and format. In this issue, every article is original with us. Next month, we hope will start showing carefully selected and edited reprints.

You gain two ways. First you don't have to buy those magazines you don't really need. Second, you get to preview a lot of magazines you may have been considering subscribing to. If you like the kind of articles they publish (after you preview one or two in **ComputerDigest**) you can subscribe, knowing that it is a magazine you will want.

Naturally, we can't operate in a vacuum. We need some input from you too. We'd appreciate your comments, and we hope we'll be able to get a letter column started with the next issue. So do drop us a line and give us the benefit of your thinking. We need and want your help to make **ComputerDigest** a more helpful, valuable publication for you.

We know you'll like what you find in the pages that follow, and that you will be looking forward to our next issue. We feel that we have an excellent opportunity to serve, and we intend to take an aggressive stance to accomplish that purpose.

Thank you for joining us here, and we hope we'll be talking to you again, and often, in the future.

BYRON G. WELS
EDITOR

COMPUTER DIGEST

Hugo Gernsback (1884-1967)
founder
M. Harvey Gernsback, editor-in-chief
Larry Steckler, CET, publisher
Art Kleiman, editorial director
Byron G. Wels, editor
Brian C. Fenton, technical editor
Carl Laron, associate editor
Robert A. Young, assistant editor
Ruby M. Yee, production manager
Robert A. W. Lowndes, production associate
Dianne Osias, production assistant
Joan Roman, circulation director
Arline R. Fishman, advertising coordinator

Gernsback Publications, Inc.
200 Park Ave. South
New York, NY 10003
Chairman of the Board,
M. Harvey Gernsback
President; Larry Steckler

ADVERTISING SALES 212-777-6400
Larry Steckler
Publisher

EAST/SOUTHEAST
Stanley Levitan
Radio-Electronics
200 Park Ave. South
New York, NY 10003
212-777-6400

MIDWEST/Texas/Arkansas/Okla.
Ralph Bergen
Radio-Electronics
540 Frontage Road—Suite 325
Northfield, Illinois 60093
312-446-1444

PACIFIC COAST
Mountain States
Marvin Green
Radio-Electronics
15335 Morrison St., Suite 227,
Sherman Oaks, CA 91403
818-986-2001

IBM vs

Although there are more than two computer machines
Apple—always seem to dominate the headlines.

When you talk to the experts, all you're going to get is confused. Most people who already own computers compare other computers to those they have. It's something like buying a new car. It takes a bit of getting used to. If you're switching from a stick shift to an automatic you're going to have some adjustment problems. And the additional little gadgets on the new car will probably remain unused until you get accustomed to their being there!

Got a keyboard with the legends printed right on the keys? You might not like having the legends off to the top of the key, on a separate insert. But once you get used to it, who knows? You might even learn to like it.

What we're trying to say, is that while a lot of information is available when it comes to making major decisions, we'd suggest that you do not base an important decision on a superficial factor.

Because most of us like to comparison shop, here's a good jumping-off place for you:

Apple's Macintosh

This personal computer will solve all kinds of problems for anybody from a schoolperson to a business manager. It comes with a mouse, and offers a 32-bit (internal architecture) microprocessor and bit-mapped graphics. You get many of the advantages of the Apple 32 supermicro at an excellent price.

Apple's Lisa

There are three, count 'em, three Apple Lisa's. The Lisa 2, the 2/5 and the 2/10. If you want Apple, and want more Apple than you can get in a MacIntosh, go for the Lisa series. Again, the more Lisa you get, the better off you will be, so we advise investing the most you can afford to make your first expense your last. All the Lisa's come with a mouse, and as you'll see from the specifications, they keep getting more and more powerful...

TABLE 1

Apple Macintosh

Microprocessor	MC68000
Memory	64K ROM 128K RAM
Disk drives	One 3½ inch microfloppy
Disk storage	400K per disk
Keyboard	58-key detached (numeric keypad optional)
Size	9.7 (width) × 10.9 (depth) × 13.5 (height) inches
Weight	16½ pounds (without keyboard or mouse)
Other comments	Mouse, independent clock/ calendar, two RS-232/RS-422 serial ports, built-in video monitor, sound/voice generator.

TABLE 2

Apple Lisa 2

Microprocessor	MC68000
Memory	16K ROM, 496K RAM
Disk drive	One 3½-inch microfloppy
Disk storage	400K per disk
Keyboard	76 keys with numeric pad
Size	18.7 (width) × 15.2 (depth) × 13.8 (height) inches
Weight	48 pounds
Other comments	Three slots, two serial ports, one parallel port, built-in speaker with software-controllable tone generator.

APPLE

manufacturers, the same two—IBM and Let's compare what their most popular offer.

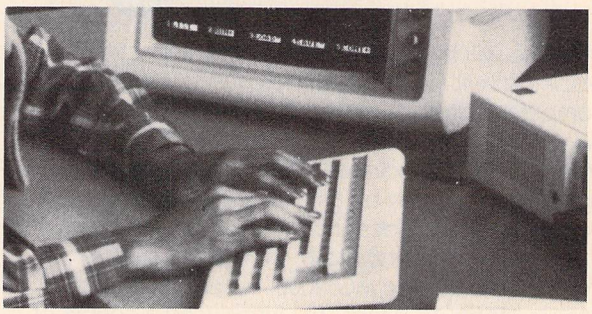
TABLE 3
IBM PC

Microprocessor	8088
Memory	40K ROM, 16K (standard) to 256K RAM
Disk drives	Up to two 5¼-inch diskette drives (optional)
Disk storage	160K per diskette
Keyboard	detached, with 6-foot cable. 83 keys, auto repeat
Size	20 (width) × 16 (depth) × 5.5 (height) inches
Weight	21 pounds (without drives) 28 pounds (with two drives)
Other comments	Cassette I/O, five expansion slots, built-in speaker, BASIC interpreter. RF modulator or video monitor must be purchased separately.

TABLE 4
IBM PCjr

Microprocessor	8088
Memory	64K ROM, 64K RAM
Disk drives	one 5¼-inch double-sided drive (optional)
Disk storage	360K on a double-sided diskette
Keyboard	cordless infrared keyboard with 62 chicklett keys
Size	13.9 (width) × 11.4 (depth) × 3.8 (height) inches
Weight	6 pounds (without drive) 9 pounds (with one drive)
Other comments	Cassette I/O, two joysticks. Cord for keyboard is optional. Display is 40 columns, 80 columns with upgrade that also increases RAM to 128K. (Both that upgrade and disk drive are included with the enhanced version. Connector for TV is optional, direct and composite video are standard.

APPLE's MACINTOSH (above) and IBM's PCjr—two new machines generating a lot of excitement.



IBM's PCjr.

Many have taken to calling this "the Peanut," and it's a truly innovative unit with some features that tickled our fancy—like the cordless keyboard that is coupled to the system unit via infrared signals to provide a degree of freedom heretofore unavailable. It's an excellent, low-priced starting unit, and will do a workmanlike job for the small office or for the home.

IBM's PC

There are two PC's from IBM. The PC and the PC XT. Chances are that when you visit your local dealer, if you ask to see the *jr* it won't be long before he's selling you "up" to a PC or an XT. Both are excellent values, and sufficiently powerful to do any sort of job that you'd call on a desktop unit for. There's a vast array of software available for both, and the specifications speak exceedingly well for these units. ◀CD▶

MACINTOSH: a new variety of apple.

Apple's long-awaited new business computer has arrived. Let's take a look at what it has to offer.

MARC STERN

■No one will know for some time the impact that the *Macintosh* will have on computer users and the computer industry—but we'd be willing to bet that it will have a major effect. The *Macintosh*, with its MC68000 microprocessor, icons, and mouse, (we'll explain shortly) may be an example of how all computers will be designed in the years to come.

To be sure, the *Macintosh* isn't the first small computer to use the MC68000. For example, Radio Shack's *Model 16*, which was introduced a couple of years ago, and Apple's *Lisa* both use it. But the *Macintosh* is the first system to use the power and capability of that microprocessor in a computer that costs under $2500.

Just what do you get for your $2500? You get the main unit, the keyboard, and the mouse. The main unit includes a 9-inch black-on-white display (512 × 342 pixels), the microprocessor and related circuitry, (including 128K RAM and 64K ROM), a 3.5-inch Sony microfloppy-disk drive that can store up to 400 kilobytes; two RS-232/RS-422 serial ports, and built-in sound and speech hardware. That's quite a bit of power in a lightweight (17-pound) unit.

Icons and a mouse

Perhaps the most important features of the *Macintosh* are those that make it easy for almost *anyone* to use it. With the mouse (a desk-top "roller controller") and icons (graphic representations of a command or function), you can do a great many things without knowing the first thing about operating systems or command structures. All you have to do is move the mouse on the desk, and the screen pointer (cursor) moves to a picture (icon) that indicates the action you want to take. Then, you push the single button on the mouse and the system goes to work.

For example, if you want to open a file, you simply point to a file-folder icon, push a button on the mouse, and a file is opened. Or, if you want to create some text (provided the proper application program is in the disk drive) you then point to the appropriate picture or menu command at the top of the screen and you are in the text-processing mode. During this time, the computer is handling all the tasks without requiring you to remember any commands to control its operating system.

That ease of use points the way the personal computer world is likely to go. It seems almost inevitable that the personal computer of the future will be easy to use yet powerful. It's likely the personal computer of the future will take only a few hours or so to learn adequately so you can use it for whatever purpose you have in mind.

The *Macintosh* is a logical progression of the small computer-system. In the last decade, since the first, "primitive" microcomputers became available, the industry has advanced a step at a time from 4-bit to 8-bit to 16-bit and to 32-bit architecture. Each step in this progression has brought with it an increase in power and capability.

Eight-bit microprocessors, with their 16 address lines, can directly address a maximum of 64K (actually 2^{16} or 65,536) memory locations. However, that figure was the upper limit of its capability. Some 8-bit-based systems do claim the ability to address more memory, but they are using fancy memory and timing techniques (bank selecting) to achieve that affect.

For its time and for most purposes, that was more than adequate for many computer users. It was a fairly simple setup, requiring only 16 address lines, eight data lines, and various control-bus lines. But that simplicity had its drawbacks, especially for systems and program developers, who just love more and more computing power. The 64K limit of memory addresses and the relatively primitive level of instructions available limited them in what they could do.

Fortunately, the 16-bit microprocessor began to make its appearance. Here is a device which can directly address more than a megabyte of memory (2^{20} bytes). Of course, it is a bigger, more powerful device. It uses 20 address lines plus a 16-line data bus for its 16 internal data registers, and lines for timing and other control functions. System designers and programmers can do a great deal more with it because not only can it support far more memory, it also has a richer instruction set.

Although it was the type of device which could do cartwheels in the eyes of developers, it was also a device which required more support circuitry than an 8-bit microprocessor. So, of necessity, the motherboard in a 16-bit computer became much more complex.

And that brings us to the *Macintosh* and its 32-bit internal architecture, the next step in the microcomputer revolution. Unlike its predecessors (such 8-bit devices as the Z80 or 8080 and such 16-bit

microprocessors as the 8086/8088), the MC68000 is packaged in a 64-pin flat pack with 24 address lines, 16 data lines, and 20 control lines. It has a 32-bit internal architecture (its has seventeen 32-bit data and address registers) and it has the capability of directly addressing 16 megabytes with its 24 address lines.

Macintosh's only limitation at the moment is the amount of RAM available: 128K. Unless a program designer wants to move applications routines in and out of memory with constant use of the single disk drive (a second optional drive is available), most routines will actually be single tasks. When Apple upgrades the *Macintosh* to a promised 512K of RAM in the fall, then multitasking and true concurrency will be available.

Still, Apple's unique way of handling input and output makes it seem as if *Macintosh* is capable of concurrency. You are able to call up two concurrent applications under the main program and have them resident on screen, with one of the executing. Much of this is due to the fact that a large part of the *Macintosh*'s operating program is stored in read-only memory and is called for by the disk-based applications program. Because it is, very little RAM is wasted holding the operating system and thus the entire 128K is available for use.

But this is getting away from the point here which is that *Macintosh*'s microprocessor points the way to the future and it is a logical step in the microcomputer revolution.

16 or 32 Bits?

You might have noticed that whenever we said "32-bit," the word "architecture" always followed. Although Apple markets the *Macintosh* as a 32-bit computer, there are some who would debate that terminology, preferring to call it, instead, a 16-bit computer. The reason is the number of data lines. The 68000 uses 16 data lines (rather than 32) to bring data to its 32 registers. The result is that the device must make two 16-bit fetches, which slows things a bit. However, once the data are inside the processing unit, they are processed in a 32-bit manner. So what kind of computer is the *Macintosh*? It is a 16/32-bit machine. The MC68000 uses special timing routines for data input, so it works nearly as fast as a true 32-bit microprocessor—although Motorola (the microprocessor's manufacturer) calls it a 16-bit device. For convenience, though, we'll continue calling the *Macintosh* a 32-bit machine. Just remember, we're referring to the microprocessor's internal architecture.

Macintosh comes with 192K of internal memory, which consists of the 64K ROM that stores many of the operating system primitives and device drivers, as well as other system and device calls, and 128K of RAM. Mass storage is via a 3.5-inch microfloppy diskette, which holds 400K of information on a single-sided double-density disk.

When you open the system box you will find all of the *Macintosh*'s memory and processing power is on one motherboard running along one side of the box. This board contains the CPU, RAM and ROM. It also contains the connectors for the mouse controller, as well as the connector for the optional external floppy disk drive. Further, it also contains dual serial ports to interface a printer and modem. The system also contains a built-in, battery-driven clock/calendar and the sound/speech generation chip.

With the exception of the 68000 CPU and microfloppy diskette drive, the basic *Macintosh* outline is pretty similar to many other small computer systems on the market. But Apple has done something that is unique. For a better understanding of this, it's only necessary to take a look at the architecture of the system, as well as the user interface to see why *Macintosh* is innovative.

Unique architecture

At the heart of this system is the 68000 CPU and its memory. In the *Macintosh*, the RAM data-output lines are connected to a different bus from that used by the rest of the system, increasing operating speed. The RAM has three different entry ways, each of which give the CPU, screen display and sound-generation hardware separate periodic access to the data and address buses so that sound, video and the current task that the 68000 is performing appear concurrently executing to the user.

The ROM, which contains low-level graphics primitives, operating system routines, and user-interface

THE *MACINTOSH* WILL NOT take up too much room on your desktop. Its "footprint" (not including the keyboard or mouse) is 10×10 inches.

A VARIETY OF ACCESSORIES is available for the *Macintosh*, including an outboard drive, a numeric keypad, printer, modem, and carrying case.

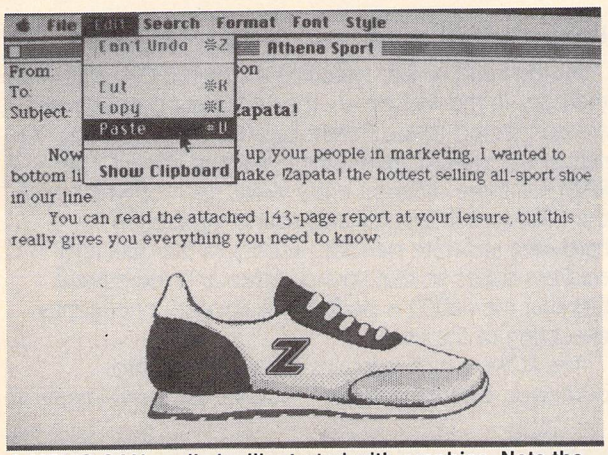
routines, is directly connected to the system bus and is used only by the CPU. Because of the way its software is encoded and stored—in low memory locations—the ROM-based subroutines function as extensions of the 68000 instruction set and operate at system speed. Therefore, the ROM is always accessed at 8 MHz.

Video memory appears as a linear array of 10,944 16-bit words of data. Because of the arrangement of video memory with the rest of the system, RAM access time slows to about 3.5 MHz during the horizontal video refresh period. That prevents video glitches.

A key to the *Macintosh* system, though, is the serial bus. It is through this bus that most peripherals will interface with *Macintosh*. It can run in two modes, the first using an external clock which allows data transfer at up to one megabit-per-second; the second mode uses an internal clock and allows data to be transferred at up to 230 kilobits-per-second. That is the speed at which most peripherals will transfer their data.

It is through this latter feature that *Macintosh* implements "virtual" slots. Rather than relying on conventional slots as has been Apple's policy with its tremendously successful *Apple II+* and *IIe* lines, the company interfaces each piece of peripheral equipment in a daisychain manner off the serial bus. Each peripheral has its own power supply and will use the 230-kilobit data-transfer rate.

While a closed bus might seem to be a disadvantage, virtual slots do have some advantages that become apparent. For starters, it means the system box can remain closed, thus preventing radio-



A MEMO CAN easily be illustrated with graphics. Note the window near the top corner of the sereen.

frequency interference (RFI) problems that have troubled many other manufacturers. Further, since there are fewer mechanical connections in a serial interface than on a typcial add-on computer card, there are fewer places where mechanical troubles can occur. And, since each peripheral will use its own power supply, the power requirements of the *Macintosh* can be kept low and stable, and the company doesn't have to worry about beefing up the supply to handle future peripherals. In that way, costs are kept down. Finally, for software developers, the virtual-slot system means that the memory map will always remain the same—there will be no peripheral cards inserted into memory that will change the mapping itself. Therefore, developers can be assured the programs they write will run, without worrying about whether company X's peripheral card will change things.

One last feature of the system architecture is its simplicity. Using programmable logic-arrays, large-scale and very large-scale integration techniques, Apple has been able to keep the IC package count to about 50. That not only helps to keep the computer's size and cost down, it helps to ensure reliability.

The user interface

The second key to *Macintosh*'s uniqueness is the user interface. It is simple to learn and easy to use. Rather than relying on the traditional operating system approach with its rather cryptic A> (or whatever) prompt, followed by the need for a command or series of commands, Apple has chosen the mouse and screen graphics and menus as the way someone will be able to access and use any of the applications which he may need.

The windowing capability allows several documents to be displayed on screen simultaneously. The windows can be moved, expanded or shrunk. This means that numbers, words and pictures can easily be cut and moved around and repasted elsewhere.

Since *Macintosh* uses an unchanging memory and access format, all programs—will use the same command structure in the same way. No longer is it necessary to learn and relearn new commands as new programs are acquired. This means that software designers are forced into standardization that is seldom seen in the microcomputer world.

With the *Macintosh*, there's no more worry about learning long, complicated command strings or the like. And, rather than taking 20 to 40 hours to master the machine, Apple estimates it will take 10 or less. Also, the software allows the system to share files across several programs and it gives software developers direct access to its many features through a "developer's toolbox."

Future trends

What is in store for the future of *Macintosh* is still up in the air and several questions remain with it. Like it has done before, Apple has introduced a system that is totally incompatible with the systems it already has on the market—although it will "talk" with the *Lisa* (it is *upward compatible*—meaning that the *Lisa* can run *Macintosh* software, but not the other way around).

Further, since there is no compatibility with the near industry-standard MS-DOS operating system and since there is likely to be none, there have been questions raised about its future in the business market. Also, it comes totally without software—not even BASIC is included when you buy it, so you must start adding to the system's price at the start. (Apple, however, will throw in word-processing and graphics software if you buy the computer within 100 days after its introduction date of January 24, 1984.)

In conclusion, the future of the *Macintosh* is still unclear. Yes, many people and the company are ecstatic about the machine, but there is still some uncertainty around. Whatever the outcome, one thing is for sure, *Macintosh*'s impact on the small computer-systems market is likely to be felt for years.◀CD▶

YOUR KEY TO CP/M

HERB FRIEDMAN

Here's a CP/M primer that's perfect for beginners others who need to know about that operating system on a non-technical level.

■It often appears that CP/M is the bogeyman of personal computing. Some colleges short on funds take in substantial extra income with rather expensive "How to use CP/M" adult education courses. Highly paid consultants will come to your home or office any hour of the day or evening to give you personal training in CP/M. Magazine authors suggest you consult the local "CP/M expert" to work out your problems. And computer stores offer as an excuse to users who wiped out all traces of their software from a hard disk that "it's your fault if you misuse CP/M."

Why is CP/M so difficult to use? In truth, it's not difficult at all; it's often easier to use than most of the so-called user-friendly operating systems.

Most difficulties in using CP/M arise because the system has remained essentially the same while the hardware, the program software, and the type of person using the software has changed dramatically. Many of the original CP/M concepts no longer pertain to how the modern personal computer is generally used. But once you understand CP/M in terms of modern hardware and software and your own technical level, you will find that it becomes a delight to use.

What's CP/M?

CP/M, an acronym for *C*ontrol *P*rogram for *M*icrocomputers, is only a disk operating system with a fancy name. It's primary purpose is to create disk files and then permit the user to use the files in any desired manner. For example, the output from a disk file could be directed to a printer, or another disk drive to make copies, or another computer, or a modem, or a paper punch—anything at all.

On the opposite side of the coin, CP/M allows almost anything to be used to create the disk file. The input could come from a keyboard, a teletype machine, a CRT terminal, a card reader, paper tape—again, anything.

Some of those devices and uses might sound strange because you have not heard of them or your computer has no provisions for them; and that's a good part of the reason some users have difficulty using CP/M. CP/M was never intended for the modern personal-computer or the typical person using it today. The fact that it remains the leading 8-bit computer operating system is a testimonial to how well it was initially thought out, touching bases that hadn't even been thought of in the late 1970's.

CP/M was intended for a "personal computer" that, like the mythical kingdom of Camelot, lasted for one brief instant in time. It was written for a central computer to which other equipment—called "peripherals"—were connected. Depending on the particular installation the input to the computer could be from any of several devices such as a teletypewriter, a video terminal consisting of a keyboard and CRT display, batch processor, a telephone line, a paper-tape reader, virtually anything. Similarly, the computer could feed any number of devices, among them a high-speed line printer, slow speed letter-quality-printer, several disk drives, paper-tape punches, teleprinters, even another computer. Hence, CP/M was designed to accommodate all those various peripherals, and numerous software routines commonly used by "data processing" technicians were built into CP/M so the user could easily use the peripherals.

Today, however, the typical personal computer rarely connects to that much hardware. Nor does the user need the software functions that go with the hardware; yet the functions remain in CP/M. If you call a function not implemented for your particular computer, either nothing happens, your information gets directed to places unknown to you, or the keyboard locks up. If you just don't use the features there won't be any problems. We'll cover what not to use a little later.

Also, the documentation was intended for the computer scientist/engineer/technician, and even some of them had difficulty understanding it. For many applications-oriented users of personal computers, inconsequential hassles become insurmountable problems because the user doesn't understand the technically-oriented documentation.

To make life easier and simpler for those of you who aren't computer experts, and don't care to become experts, we're going to cover CP/M from a strictly non-technical applications-oriented viewpoint. Some of you will undoubtedly gnash your teeth and shout to the world "that isn't right." Gnash away! This short course in CP/M is intended strictly for those who want

to use their computer, not write the next multi-megabyte integrated word processor/spreadsheet/graphics program.

Note: To avoid confusing computer keyboard commands, the punctuation—the periods, commas, colons, etc.—which are by normal convention located within parentheses will be shown outside the parentheses.

Drives and user areas

As stated earlier, CP/M is a disk operating system. It will write and read disk files on any of up to 16 individual or "dual identity" disk drives which are alpha labeled as "A:", "B:", "C:", etc. The colon (":") is part of the disk drive hardware identification; without it CP/M has no idea what you are talking about, and the lack of the colon is the reason why what should be an ordinary copy command can result in inaccessible disk files. While a disk drive can be assigned any letter (within reason), because of the way modern personal-computer software is written, the main drive, the so-called primary default drive, is always A: and the second drive is always B: You could have your second drive designated C: but a lot of commercial software wouldn't have the vaguest idea what to do. Unless you modify the software that is the way it must be done; otherwise, you will end up with software that won't run, or software that simply erases it's own records. (Self-erasure is very common to hard disks where A: and B: are the same disk mechanism.)

Take note that some programs permit the user to eliminate the colon when specifying disk drives, and some so-called "utility" software permit the user to use CP/M without the colon. Using a colon at one time and not at other times causes more problems than the convenience is worth because omitting a colon at the wrong time can "crash" (ruin) a file. Try to do it CP/M's way at all times—use the colon even if you don't have to.

The basic CP/M system that handles the filing and retrieval of disk data has several inherent command functions. The most commonly used command is DIR (for "directory"), which will cause the display device to show a directory of the disk files. The directory can be "keyed" so that only a desired group of files appears at any one time.

One of the attractive features of CP/M is that it can create up to 16 individual "user areas" on a single disk that are keyed to individual directories for each user area. "User area" is essentially an invisible tag applied to one or more disk files, and the computer always "comes up" in user 0 when CP/M is first booted up.

You enter a user area from the command line—when the "A>" shows on the screen—by typing USER 1, USER 5, etc. Since the computer automatically comes up in user 0, the command USER 0 need be entered only when leaving some other user area.

User areas work this way: Assume your disk drive will record 200K bytes of data. Any program written to the disk is placed in the user-0 area and a DIR command will display a listing for all the disk files. If all the files were simply written to the disk, the directory would appear as shown in Fig. 1. But let's assume you don't want the directory listing cluttered with a string of

```
    DIR
A: CARDFILE COM : MBASIC   COM : LABEL2   BAS : MAILPAC  BAS
A: FAMILY   BAS : BOOKEEP  BAS : CHECKS   BAS : CHECKS82 BAS
A: INSURE84 BAS : PIP      COM : CARDFILE PRM : CARDFILE DAT
A: STAT     COM : XDIR     COM : COMHEX   COM : MITE     COM
A: FILES    DOC : FLS      COM
A>
```

FIG. 1—THIS TYPICAL DIRECTORY display shows the CP/M utilities, MBASIC and its files, and a *Cardfile* database with its files.

unrelated files; you would like them arranged into homogeneous groups. For example, you would like only the CP/M utility programs—PIP, STAT and XDIR—to be available on boot up, with MBASIC and all its related programs with the .BAS extension listed independent of both the CP/M utilities and the database program with its individual data files and prompt screens. Figure 2 shows how only the CP/M utilities would be displayed.

```
  DIR
A: PIP      COM : STAT     COM : XDIR     COM
A>
```

FIG. 2.—THIS IS HOW we would like the computer to "boot up," with a DIR showing only the utility files.

What you can do to avoid the clutter is to tag each individual file by assigning it a user area tag. As shown in Fig. 3, we could place the MBASIC file and its programs in user-2 area by first typing USER 2. Anything done from here on will have the USER 2 tag, so if we now move (write) MBASIC and its related programs to the disk they will all have the USER 2 tag. All programs created by MBASIC will also automatically get the USER 2 tag.

```
  USER 2
A>DIR
A: MBASIC   COM : LABEL2   BAS : MAILPAC  BAS : FAMILY   BAS
A: BOOKEEP  BAS : CHECKS   BAS : CHECKS82 BAS : INSURE84 BAS
A>
```

FIG. 3—MBASIC AND ITS FILES should have their own exclusive tagged directory.

To keep our database files separate we might give them a USER 5 tag. From the "A>" command prompt we type USER 5. Everything we do will now get the USER 5 tag. If we write our database, in this instance *Cardfile*, to the disk, the program and all its generated prompts and files will be tagged USER 5 (see Fig. 4).

```
  USER 5
A>DIR
A: PIP      COM : CARDFILE COM : CARDFILE PRM : CARDFILE DAT
A>
```

FIG. 4—A DATABASE PROGRAM, such as *cardfile*, and its files could similarly have its own tagged directory.

If you now go back to the "boot up" condition by typing USER 0, you would see only the CP/M utilities, as shown in Fig. 2.

In Fig. 2, the computer has been "booted" so it "comes up" automatically in the the user 0 area and a DIR command produces the response NO FILE. In Fig. 3 the computer has "come up," we have first entered USER 2 and then DIR. Notice we have a directory listing of MBASIC and its files. In Fig. 4 we have entered USER 5, then DIR, and we get only a directory of the *Cardfile* files.

There is normally no way a single DIR command will list files from other user areas unless special software has been provided, and this is not usually done except for the computers with hard-disk drives such the *Kaypro 10*. Kaypro provides a special directory program

that will list all files regardless of their user area. (There are available special utility programs that simplify getting directories and moving from user area to user area, but they are a subject for another time.) In fact, because hard-disk drives contain so many files it's often difficult to locate a desired file unless they are organized in homogeneous user area. In fact, a full directory listing for the software that comes bundled with the *Kaypro 10* would take almost four full screens to display. That does not include any user-generated files. Using user areas simplifies loading programs—it's certainly a lot easier to locate and use the desired files from a user area than trying to dig them out from the "full directory."

While the user areas are often a decided convenience you must keep in mind that the user area on the default drive, the control drive determines the user area on all other drives. If you copy a program from drive A: to a disk on drive B:, or C:, or whatever, it will be copied to the same user area from which it came. Same thing if B: is the default drive. Often, the user will be in a user area, say user 4, and copy a program to a backup disk in drive B:. Sometime later, forgetting that a user area was used during the backup, the user calls for a DIR on the backup disk and the screen shows NO FILE, because DIR checks the 0 area of B: and the files have the USER 4 tag.

In Part 2 of this series we'll show how to move programs between user areas. For now keep this point in mind: It is easy to move files from any user area into the one being used. It is difficult, often almost impossible for a non-programmer to move a file from the assigned user area into any other because the original CP/M made no provision to do it easily.

Utilities

Because CP/M is a "universal" program it is always specifically enhanced by the computer manufacturer for a particular computer. That is done through special utility programs that are supplied with CP/M, as well as by utilities supplied by Digital Research, the outfit that wrote CP/M. Some are invaluable for day to day work; others can cause the total destruction of the disk files if used by a non-programmer—they shouldn't even be on the average user's work disk because they were intended for computer scientists, technicians, and programmers. The TV technician using a personal computer for business, the dentist keeping patient records, the service shop using a spreadsheet program for financial projections, and the wheeler and dealer doing stock transactions from home have rare or no need for some of the supplied utilities; again, all they can do is damage the disk data or confuse the user.

The Digital Research utilities are: ASM, DDT, ED, LOAD, DUMP, STAT, EXSUB, SUBMIT, SYSGEN, and PIP. ASM is an assembler for those who write assembly programs; you probably don't need it so erase it from your working disk (not from the master or backup copy of CP/M). DDT is also for software experts and is used primarily to modify program code. Normally, it can also be erased from the working disk; but even if you haven't the vaguest idea of what DDT is all about, it is required one single time if you want to use user areas. In Part 2 we'll give you a short seven-statement routine for DDT that you don't have to understand at all, but it will open up user areas to you—and then you can forget all about it.

ED is a rudimentary line-oriented editor that wasn't even good in "the good old days;" kill that one also. LOAD and DUMP are also for computer science types; you will probably never use them so kill them from the work disk. SUBMIT is a way to automatically link commands, as is XSUB. Many software houses supply their programs in SUBMIT or XSUB format under a different name. For beginners and application-oriented users it's best to avoid those programs until it's you who are the local CP/M expert.

SYSGEN is the program that is used to place the CP/M system on other disks. Leave it on the work disk. STAT means "statistics." It will tell you almost anything you want to know about a disk or the programs stored on the disk, such how much free space is available on the disk; how much disk space a program takes up (or needs if you're making a copy); the devices connected to the computer; the condition (setup characteristics) of the devices. STAT is important, and you should hands-on experiment with it until you can use all its "bells and whistles."

PIP—an acronym for *P*eripheral *I*nterchange *P*rogram—is the utility that moves files from place to place and it is the keystone of CP/M. Unfortunately, PIP works on two levels: very-easy-beginner and difficult-to-understand mumbo-jumbo *technese*. (Technese is a language whose purpose is to insure that outsiders don't understand what's going on. For example, social workers speak technese, educators speak technese, etc.) In Part 2 of this series we will cover PIP strictly on the beginner level. Be prepared to follow along "hands on" because you'll be the local "CP/M applications-software expert" when you're finished.

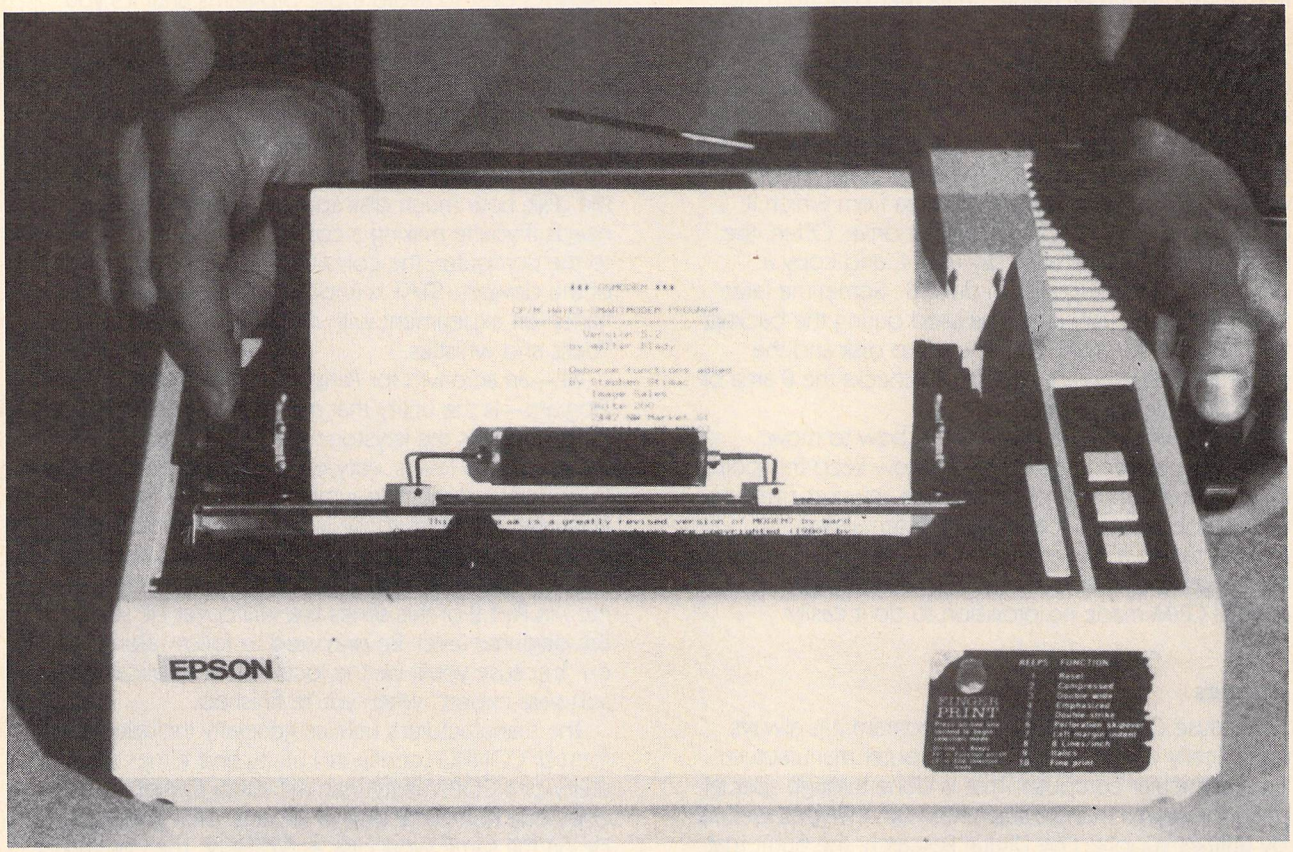
The manufacturer's utilities generally include some form of CONFIG (configure) utility, and either a FORMAT utility or a COPY utility that will either format a disk or copy one complete disk to another disk. That should be on the same work disk as SYSGEN. The command FORMAT (or FORMAT through the COPY program) places invisible magnetic "tracks" and other information on blank (new) diskettes so they can be used to store disk files. You cannot store information until a disk is formatted. If the disk isn't formatted there is no personal computer using any operating system that will know the disk exists. That is a point often forgotten, if ever known, by many "toy store" computer salespersons.

The CONFIG program is a means whereby the user can assign different technical attributes to various "ports"—the connections for a printers, modem, or whatever. It is generally not needed on the work disk since once CP/M is configured, meaning "set up" for particular hardware or software there is rarely any need to change things.

If the manufacturer provides other utilities they are generally intended for his particular computer, or specific software, and aren't really part of CP/M.

Next time, we'll PIP and STAT through several disks and user areas. Be prepared by having a few SYSGEN'ed disks available. ◀◉▶

Upgrade Your Budget Printer



Many low-priced printers can be easily upgraded to give them new features and greater flexibility. Here's a look at some of the upgrade kits currently available for two of the most popular of those printers.

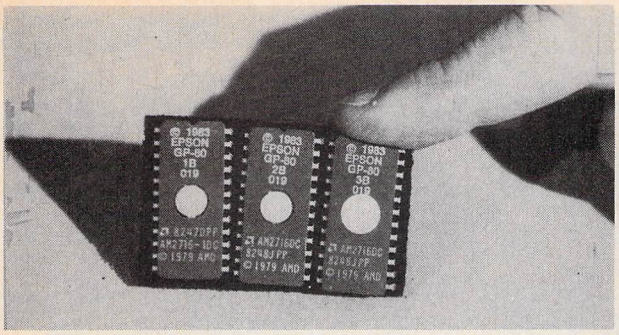
HERB FRIEDMAN

■If you have been into personal computing from its "hobbyist" days, or have built your system on a tight budget, it's odds-on that you have either an Epson *MX-80* matrix printer or Smith-Corona *TP-1* (or *TP-2*) daisy printer. Either, or both, have probably given you years of trouble-free service, but lately you find that more and more software can't be used with your printer, or you want to upgrade the computer but find your printer can't quite hack it as an up-to-date printer with the new computer hardware.

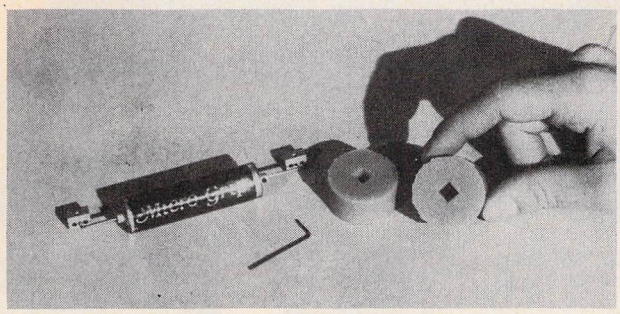
Essentially, what you have is still functional equipment that's been made obsolete by modern software, or by modern uses for a personal computer.

The Epson *MX-80* printer, which could be used with any computer having a Centronics-compatible parallel printer output, was intended primarily for the Radio Shack *TRS-80 Model 1*, the most popular computer of its day, hence the *MX-80* featured a Radio Shack graphics mode. Much of the modern software, however, is written for the graphics mode of the newer *MX-80* printers, which has the graphics capability for the present most popular 8-bit computers, the Apple *II* and *IIe*. Even if the software is for other than Apple computers the graphics will most likely be intended for the graphics capability of the most recent version of the *MX-80*.

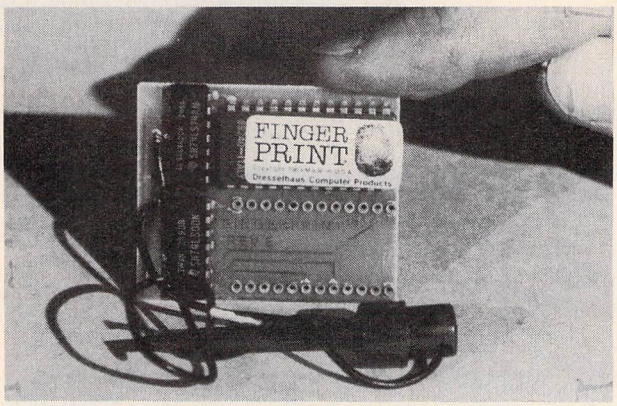
The *MX-80* (and its clones such as the Texas Instruments and IBM printers) accommodates only tractor-feed paper. If you want to print on single sheets, such as letterhead, the only way to do it is to use a special plastic tractor/pin feed "carrier," some of which are prone to damage the printhead when pushed



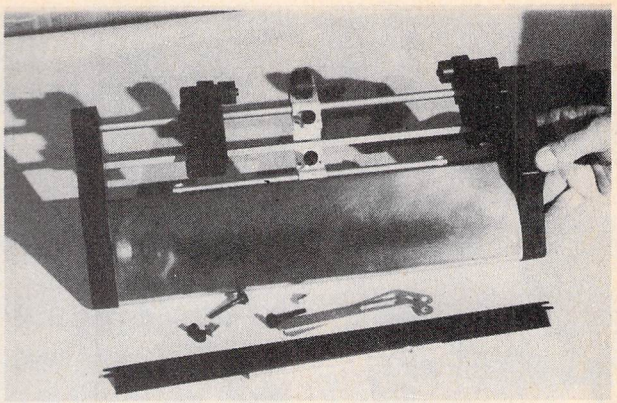
EPSON'S OWN *Graftrax-Plus* upgrade consists of three ROM's supplied on a strip of conductive foam



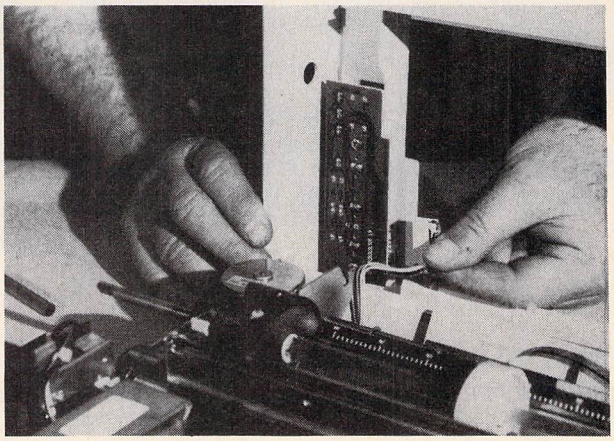
THE *MICRO-GRIP* single sheet upgrade kit. Its use does not interfere with tractor operation.



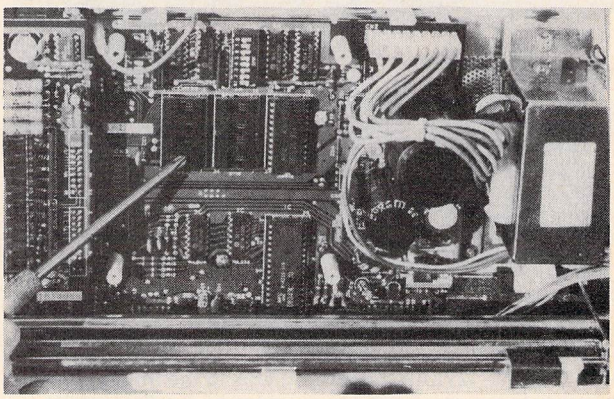
THE *FINGERPRINT* module plugs directly into an Epson ROM socket. The existing ROM is moved to the empty socket on the board.



THE PASSIVE TRACTOR FEED upgrade kit for the Smith-Corona printers. The main section is supplied as a complete assembly.



EPSON TELLS YOU to work on the printer with the cover attached, but don't do it—"pull the plug" instead.

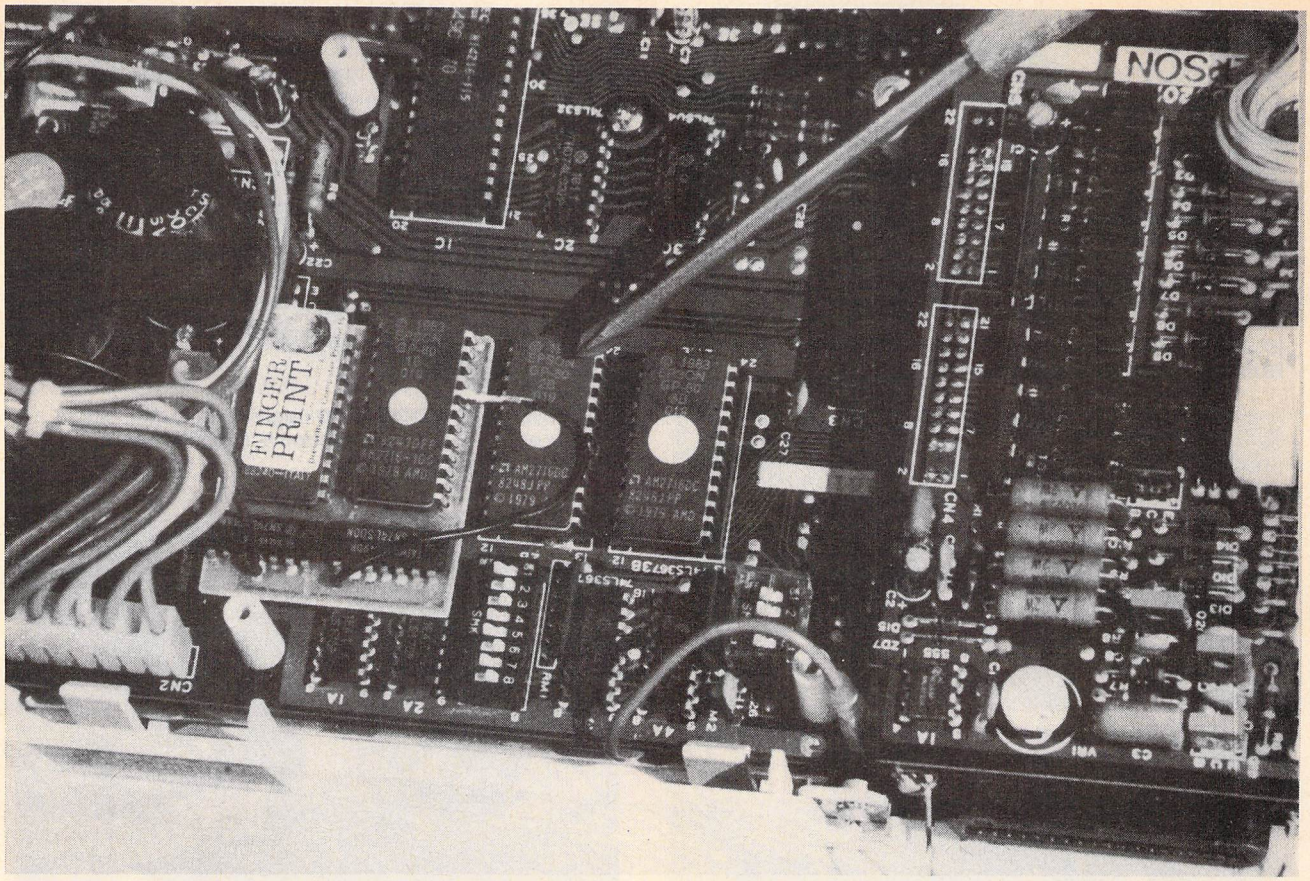


THE THREE ROM sockets located in the base of the *MX-80* printer. If you have an original *MX-80* two of those are empty and one is filled.

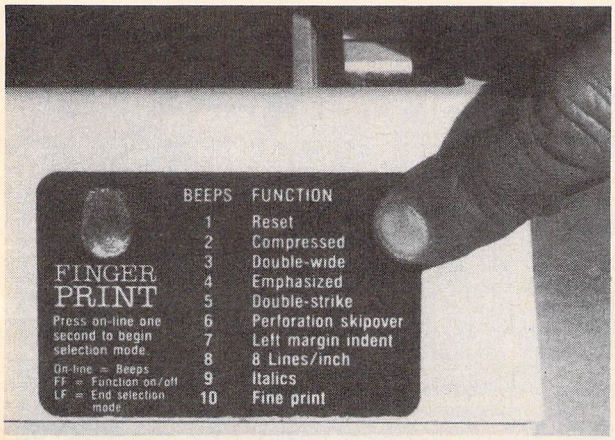
through the printer mechanism. While the original *MX-80* has a host of features such as compressed or enhanced printing it lacks a backspace, which precludes underscoring from some of the less expensive (but otherwise excellent) word processors that are available.

The "letter quality" Smith-Corona daisy printer TP family, which consists of the *TP-1* and the newer *TP-2*, is the "buy of buys" when it comes to letter-quality printers for home-and-family and small businesses. The *TP-1* was the first under-$1000 daisy printer, which by early 1984 was selling for as little as $250. Thousands upon thousands of personal computer users who could not otherwise possibly afford a letter quality daisy printer struck gold in the *TP-1*/*TP-2*—the price might very well be the reason you decided to get a letter quality printer even though you already had a matrix printer. (If a type 251 ribbon is used—which was not mentioned in the early documentation—the *TP-1* can produce "camera ready" print quality the equal of machines costing well over $1000 because it is essentially the printer end of a Smith-Corona electronic typewriter with an accessory interface for computer output.)

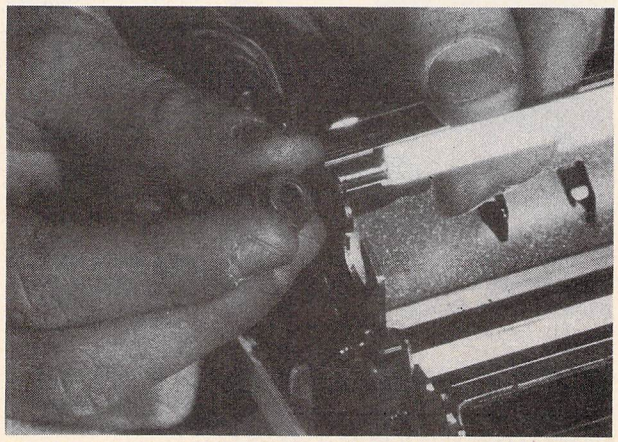
The problem with the *TP-1* and the *TP-2* is that they were intended for single-sheet documents such as business letters, etc., which is logical because it is really the printer mechanism from a typewriter. But today, much modern software is intended for continuous forms printing, such as checks, labels, IRS



THE *FINGERPRINT* assembly snaps into ROM socket 1B, while the ROM that normally goes in that socket is installed on the *Fingerprint* assembly.



AN ADHESIVE MYLAR LABEL you affix to the front of the printer lists the new functions provided by the *Fingerprint*.



TO ADD the *Micro-Grip* single sheet feed to the *MX-80* you simply remove the collar securing the drive bar on the left side.

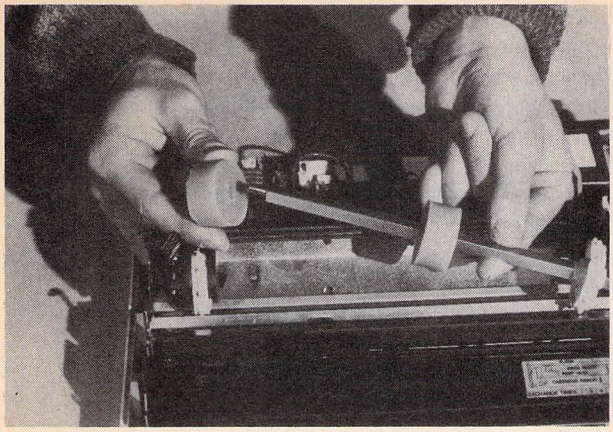
schedules, even business stationary—which includes letterheads and envelopes. To insure precise alignment of the printing, continuous forms must be tractor or pin fed.

So what we have is a popular tractor-feed matrix printer that needs at least new graphics capability and the ability to feed single sheets, and a daisy printer that needs a tractor feed in order to accommodate continuous forms. Let's see what we can do to satisfy those important needs.

Upgrading the printers

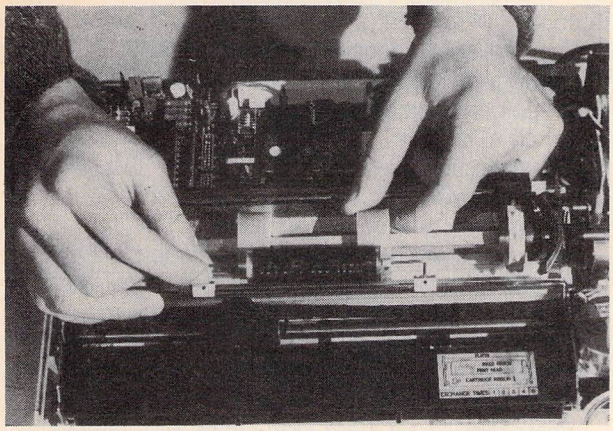
Regardless of why you purchased the *MX-80* or *TP-1* in the first place, user-installed retrofit kits are available that will upgrade either printer to accommodate modern needs. While some of the retrofits are better or more convenient than others, we know those discussed here will really work exactly as promised because we actually tried them out.

The first of the user-installed *MX-80* retrofits was Epson's own *Graftrax*, which consisted of three ROM's, two of which plugged into empty sockets in the base of the printer while the third replaced an existing ROM. Among other things, the *Graftrax* upgrade provided for dot-addressable graphics, itallics printing, and even more important, backspacing. It was terrible

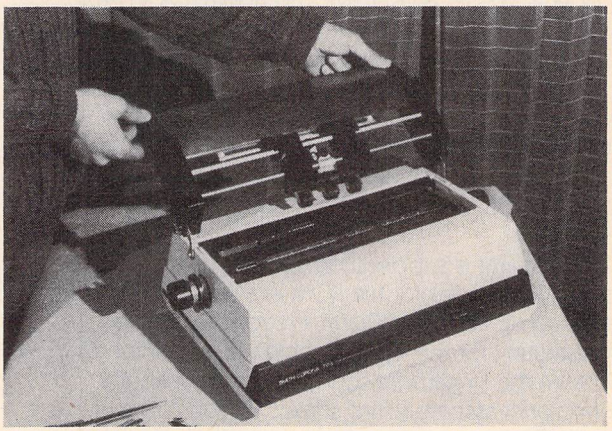


SLIDE THE ORIGINAL plastic paper guide roller from the driver bar and slide the two Micro-Grip rubber rollers on the bar. Then replace the collar.

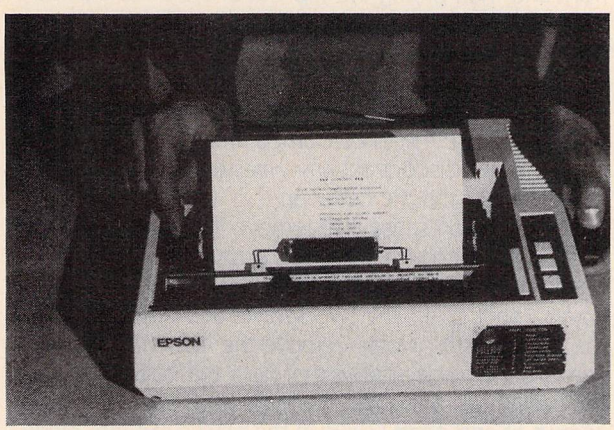
WHEN WORKING ON the Smith-Corona *TP-1*, you cannot separate the casing top because the leads to the fan are too short.



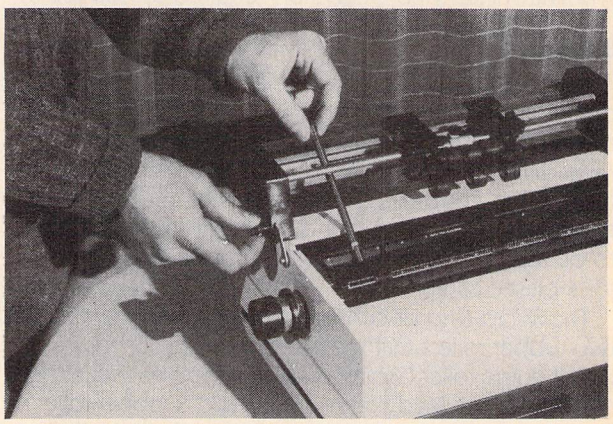
ATTACH THE PRESSURE roller assembly to the paper bail and the single sheet retrofit is finished.



IT APPEARS TO BE permanently installed, but the tractor mechanism goes on and off in seconds.



A SINGLE SHEET being printed by the *MX-80*. Be certain the pin-feed mechanisms are against the sides of the paper it feeds straight.



SLIDE THE SUPPLIED GAUGE—actually a notched rod—on the drive bar, tighten two screws, and the tractor assembly is adjusted for use.

backspacing because the printhead returned to the extreme left and then advanced for each backspace, but it was adequate for general word processing: underscore, kerning, etc. Unfortunately, a graphic printout would cycle the head continuously until the drive motor could be heard slowing to a crawl.

The latest retrofit available from Epson dealers is *Graftrax-Plus*, which gives your old model *MX-80* the advanced features of the latest model. Among the *Graftrax-Plus* highlights are the Apple-compatible graphics (no more Radio Shack *Model 1* graphics); a true backspace that puts graphic printouts in high gear and relieves the strain on the printer's printhead-drive motor; a "fine print" that can be used for superscripts and subscripts; automatic perforation skipover (no more program listings printing on the perforations); and a continuous underscore that can be turned on and off from within a program or word processor. (A continuous underscore is formed simultaneously with the character; the head does not backspace for the underscore.)

The *Graftrax-Plus* retrofit kit consists of three ROM's

and the latest Epson-with-Graftrax printer manual—which explains all the "bells and whistles." The *Graftrax-Plus* retrofit has the same kind of installation as for the original *Graftrax* upgrade—just plug the ROM's into the correct sockets. One "caveat" however: The Epson instructions leave the top of the printer attached when you open the printer by separating the top and bottom; it has you delicately balancing the cover on its end. If you look closely you'll see the wires to the switches in the cover are attached through a connector. Mark the orientation of the connector with a pencil or pen and then separate the connector and put the cover in a safe place until you're finished.

Pushbutton control

As long as you have the printer open consider a retrofit called *Fingerprint* (Dresselhaus Computer Products, 837 E. Alosta Ave., Glendora, CA 91740), which allows the three printer control pushbuttons to also program ten operating modes: compressed print; double-wide; emphasized; double-strike; perforation skipover; left margin indent; 8 lines/inch spacing; itallics, and fine print. For example, just touch the printer's ON-LINE button twice and the printer shifts to compressed type without any commands from the computer.

Fingerprint is supplied on a small printed-circuit board that swaps for the ROM in a socket 1B, the one that's replaced in the *Graftrax* retrofit. Instead of substituting for the ROM in socket 1B, install the *Graftrax* ROM in the *Fingerprint* socket and then snap the *Fingerprint* assembly into socket 1B. You clip two attached *Fingerprint* leads where indicated and you now have both the *Graftrax* and the *Fingerprint* retrofits, and with virtually no extra effort required on your part.

Single sheets

If you quit now you'll have one heck of a matrix printer, but you can go one step farther and add a real single-sheet feed by installing a *Micro-Grip Friction Feed* (Bill Cole Enterprises, Box 609, Wollaston, MA 02170). The *Micro-Grip* retrofit does not interfere with the tractor operation, but it does permit single sheets to be fed directly through the printer without the need for a plastic carrier.

The *Micro-Grip* kit consists of three components: two rubber rollers that replace the existing Epson paper guide roller (which is sandwiched between the two tractor pin feed mechanisms), and a rubber-roller pressure assembly that clamps to printer's paper bail—the bar that holds the paper down for printing.

To install the *Micro-Grip* retrofit it is necessary to very slightly dismantle the printer's feed mechanism so the rubber rollers can be fitted to the tractor-drive bar (instructions are provided in the kit). It is, however, a minor disassembly, and the whole installation shouldn't take more than 10 or 15 minutes. Just be certain you don't push or bend anything while you are doing the work—do everything very, very gently.

When you see the *Micro-Grip* installed you won't believe it will work, but it does. The only problem is the paper tends to skew, and the instructions on how to "fine tune" the paper feed really don't do much of anything. You'll spend more time fussing with the "fine tuning" than printing. Just ignore the "fine tuning" instructions. Instead, gently slide the tractor pin-feed devices against the sides of the paper and apply their locks, thereby locking them in position against each side of the paper. The pin feed assemblies will guide the paper so it rolls through nice and straight. The adjustment of the pin-feed mechanisms are "permanent" as long as you keep using single sheets. To use continuous form tractor paper you simply slide the two rubber rollers to the side and reset the tractors, or leave the rollers where they are and pull the paper bail (with its roller) away from the paper; the Epson will print on continuous tractor feed forms just as well without the bail.

Installing a tractor feed

The tractor feed upgrade for the Smith-Corona *TP-1/TP-2* isn't really a retrofit because it does not really become a permanent part of the printer, it can be easily removed in seconds. The tractor feed mechanism, which is available from some (not all) Smith-Corona dealers and the Smith-Corona service centers is a "passive" tractor feed, meaning it's really a guide that insures precise registration even though the paper is really driven by the normal platten mechanism—just like a single sheet.

While installing the tractor feed upgrade can be a user-performed task, Smith-Corona does not provide the first three pages of the documentation with the kit; the pages that show how to do it yourself. Smith-Corona service centers will do the installation for $10 if you deliver the printer to the center.

What's missing are these instructions. First, remove the casing top by loosening two screws at the top front and then prying the top out of the three clamps in the base, thereby separating the casing top from the base. The three clamps are across the rear of the base; you'll need to use a large screwdriver as a pry bar and you'll swear you're breaking the case but that's what it takes to release the three clamps at the back of the casing. Don't disconnect any wires even though they are short; just flip the cover up.

Use common sense and remove the combination dust cover and paper rest, and install the supplied black metal strip (which is called an "electronics cover") so it spans the two screws that originally held the paper rest's pivots—remove the rest with its pivots. Remove the hinge-screws that hold the paper rest to the casing top. Drive out the hinge pins into which the screws fit and gently drive in the new, supplied, hinge pins with a small soft-face hammer—or the back end of a screwdriver. That's it—reassemble the cabinet.

The supplied documentation shows how to install and adjust the tractor mechanism itself, all of a 10 minute job at the very worst. A special gauge is provided in the kit for alignment of the tractor mechanism. The whole project looks much more difficult than it is. Actually, separating the back of the casing top from the base will be the most difficult part of the upgrade.

With the tractor-mechanism upgrade the *TP-1/TP-2* can accommodate either single sheets, tractor-fed sheets, or tractor-fed forms and labels. As with the Epson retrofits, you lose nothing; you only gain. ◀●▶