Radio-Electronics®

APR. 1982 $1.25

BUYER'S GUIDE
SPECIAL SECTION
YOUR OWN COMPUTER

Build your own
VIDEO TITLER

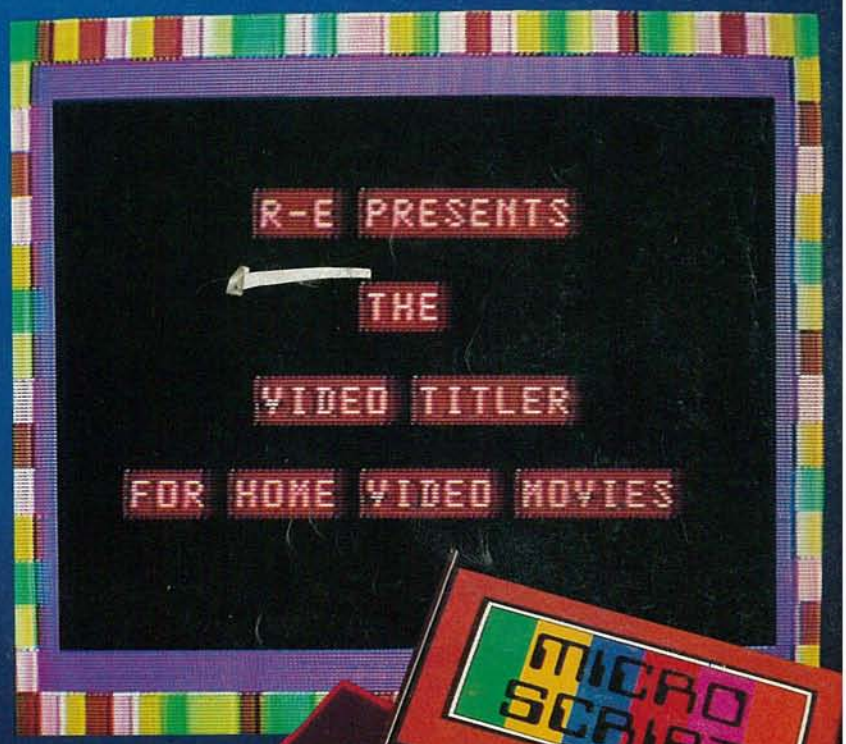
YOUR OWN COMPUTER BUYERS GUIDE

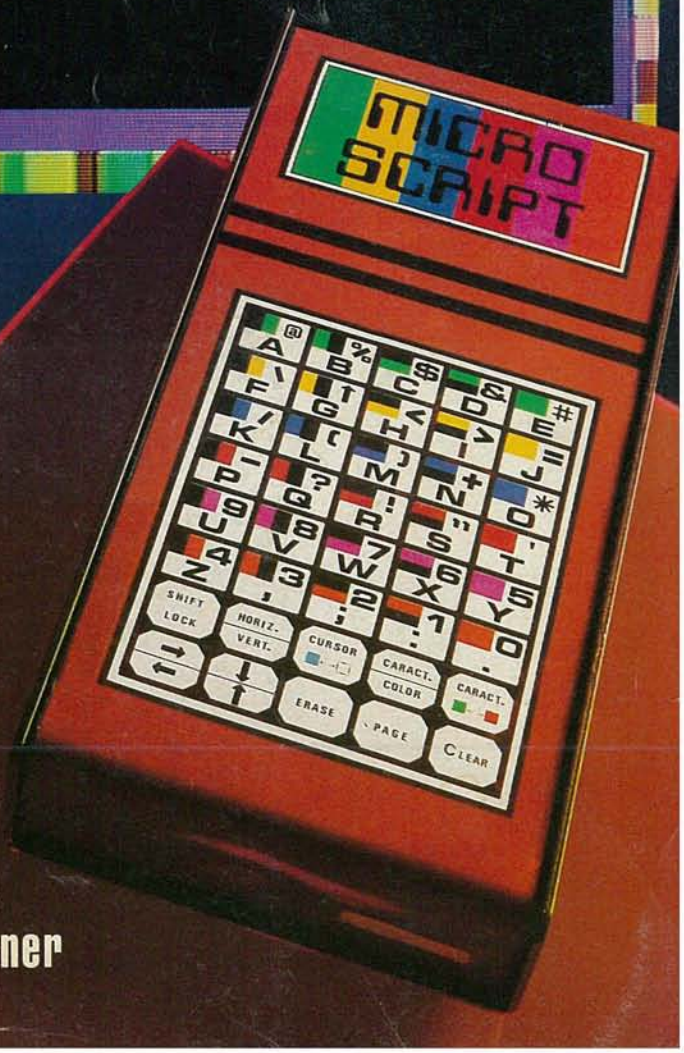
SOFTWARE

- ★Operating Systems
- ★High Level Basic
- ★Word Processing
- ★Data Base Management
- ★Utilities

HARDWARE

- ★Commodore ★IBM
- ★NEC ★Osborne
- ★Sinclair ★Xerox
- ★Pocket Computers





PLUS:

Hobby Corner
Service Clinic
State-Of-Solid-State
Communications Corner
Equipment Reports

04
0 71896 48783

Radio-Electronics

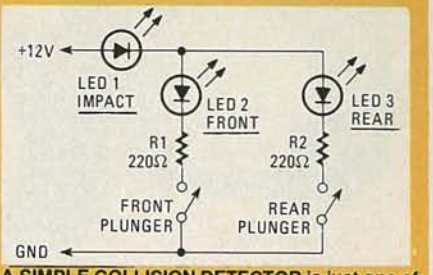
THE MAGAZINE FOR NEW IDEAS IN ELECTRONICS

Electronics publishers since 1908

ARPIL 1982 Vol. 53 No. 4

ON THE COVER

The video titler is a graphics and alphanumeric character generator for the amateur videophile. Connect it to your VCR and video camera setup and use it to add titles to your home video movies. Get started today building this microprocessor-based video accessory. The story starts on page 39.



A SIMPLE COLLISION DETECTOR is just one of six easy-to-build projects for your car that were designed especially for April 1. Turn to page 46.

IBM's PERSONAL COMPUTER is just one of the many hardware systems covered in this month's special section. Both hardware and software are covered. Turn to page 51.

Radio-Electronics, (ISSN 0033-7862) Published monthly by Gernsback Publications, Inc., 200 Park Avenue South, New York, NY 10003. Second-Class Postage Paid at New York, N.Y. and additional mailing offices. One-year subscription rate: U.S.A and U.S. possessions. $13.00, Canada, $16.00. Other countries, $20.50 (cash orders only, payable in U.S.A. currency.) Single copies $1.25.

Subscription Service: Mail all subscription orders, changes, correspondence and Postmaster Notices of undelivered copies (Form 3579) to Radio-Electronics Subscription Service, Box 2520, Boulder, CO 80322.

Radio-Electronics
YOUR OWN Computer®
APRIL 1982

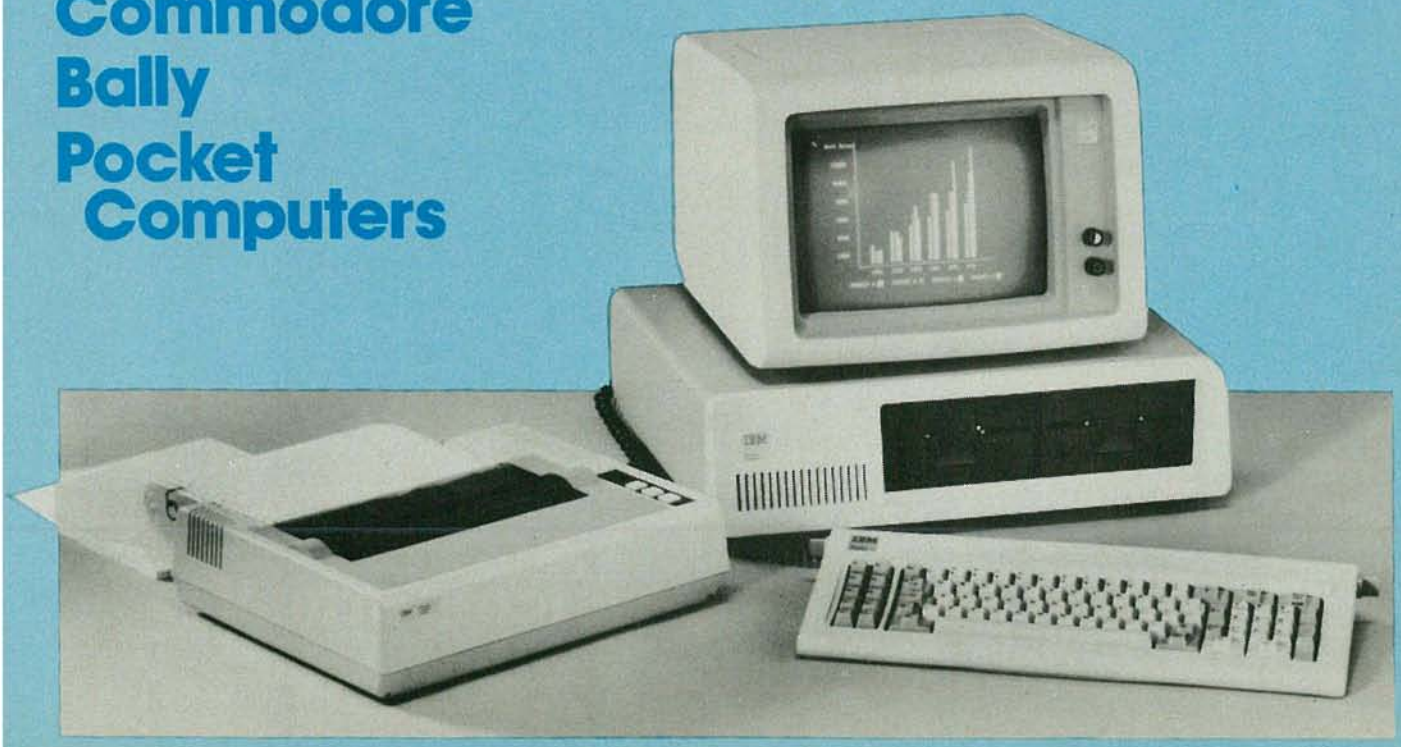
BUYER'S GUIDE TO HOME COMPUTERS

HARDWARE

Xerox
Osborne
NEC
Sinclair
IBM
Commodore
Bally
Pocket Computers

SOFTWARE

Disk Operating Systems
High-Level BASIC
Word Processing
Data Base Management
Utilities



Performance, Quality, Reliability.

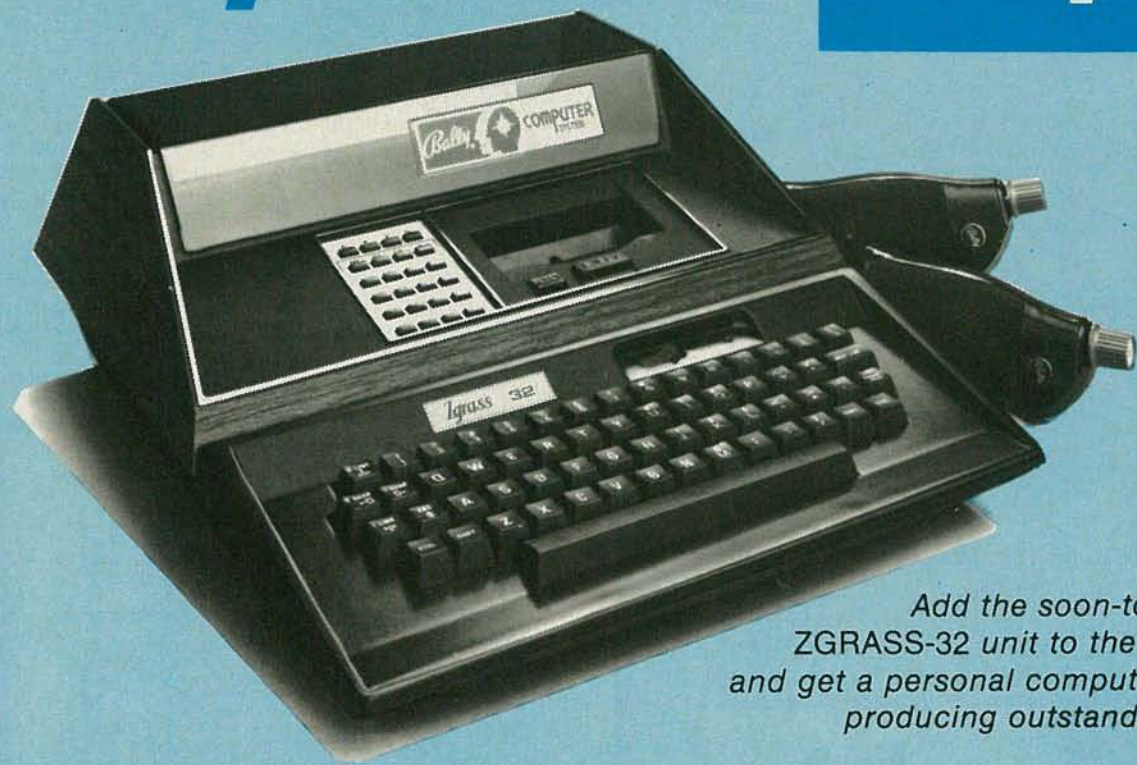
From our super-efficient memories to the innovative **CPU8085/88**—the first, and only board to deliver 8 *and* 16 bit computing on the same bus—**CompuPro** components have proven themselves since 1974. Put them in our heavy-duty enclosure (featuring a power supply people rave about), stack that on top of our DMA controlled disk drives, and you've got the system for the 80's.

The finest computing systems available.

For business, scientific, and industrial computing solutions, contact your nearest CompuPro systems center.

CompuPro division Godbout Electronics, Oakland Airport, CA 94614-0355 (415) 562-0636

Bally Arcade



MARC STERN

Add the soon-to-be-released ZGRASS-32 *unit to the Bally* Arcade *and get a personal computer capable of producing outstanding graphics.*

IN MANY INSTANCES, THE DECISION TO BUY A HOME COMPUTER is difficult. That is true especially if the ultimate use for a home computer is somewhat uncertain. Very often, the potential customer is interested in using the computer to play elaborate video games and is unwilling to spend the necessary money for a full-blown computer system. In response to that need, several manufacturers have developed programmable video games that can be expanded into a computer system at a later time. Most notable is Matell's *Intellivision*, although the computer expansion is still only in the test-marketing stage. Another system along those same lines is the Bally *Arcade*.

The name "Bally" has long been associated with the coin-operated video arcade-game field. Bally's coin-operated video games have featured high-level graphic capability with imaginative animation. However, the name "Bally" hasn't been associated with the computer field at all; in fact, it would normally seem out of place in a computer buying guide. But that isn't the case any more. "Bally" is now linked with the personal computer world, thanks to this brand-new computer offering.

Actually, it is a Bally unit in name only; while the unit was developed by Bally, it was later sold to Astrovision Inc. (6460 Busch Blvd., Suite 215, Columbus, Ohio 43229). Astrovision is currently marketing the unit, which is correctly called Astrovision's *Bally Arcade*.

The base unit, shown in Fig. 1, is a video game. It is designed to connect to the antenna terminals of a standard TV set, and comes complete with four *joysticks*. However, the joysticks are unconventional in that they are not operated by moving the joystick. Instead, there is a knob on top of the joystick that is operated using your fingers. The joystick also contains a *trigger*. The front-panel of the *Bally Arcade* contains a keypad with 24 keys and a slot for the pre-programmed cartridges. The *Bally Arcade* is capable of producing 256 colors on your TV screen, however, only four can be displayed at a time.

Internally, the *Bally Arcade* has a Z80 microprocessor that operates at 1.8 MHz. There are also two custom LSI IC's, and a video processor that operates at 7 MHz and handles all color manipulation and animation effects. The video processor provides NTSC video to the TV set. The second custom IC is the I/O processor that handles up to four joysticks, four analog-to-digital converters, and the 24-key keypad. The I/O processor also creates the music and sound effects. Three separate sound synthesizers provide both AM and FM noise over a frequency range of 2 Hz to 100 kHz. An 8K internal ROM contains the software routines for color and sound effects that are used with the plug-in cartridges. In addition to the ROM, the unit comes with 4K of RAM. Of that RAM, however, only 1800 bytes are user available; the rest of the memory is used by the video display. In its basic form, the *Arcade's* memory can not be increased.

A video game is only as good as the games available, and there are some nice ones for this unit. *Space Fortress*, shown in Fig. 2, is much like the coin-operated *Space Zap* game. *Munchie*, shown in Fig. 3, is a variant of the popular *Pac Man* game. *Coloring Book with Light Pen* (Fig. 4) lets you create your own "art work," making full use of the system's graphic capabilities.

The *Bally Arcade* supports *Astro BASIC*, which is an enhanced version of *Bally BASIC*—one of the many versions of BASIC that has come into existance. *Astro BASIC* is available as a plug-in cartridge that includes an audio-cassette interface for program storage. Programming is accomplished through the 24-key keypad.

What we've seen so far gives us little reason to classify the unit as little more than an advanced video-game. What qualifies this unit is an add-on that will be available sometime this year. That add-on is called *ZGRASS-32*. Among that unit's features are a full-size ASCII keyboard, 32K of ROM, and 16K of RAM. Resident in the ROM is ZGRASS, a powerful graphics-oriented language. With the *ZGRASS-32* add-on, all programming is done in ZGRASS.

The *Z GRASS 32* add-on makes the new system unique.

FIG. 1—THE BALLY *ARCADE* video game is shown here in it's basic form. The games shown are only a few of those available.

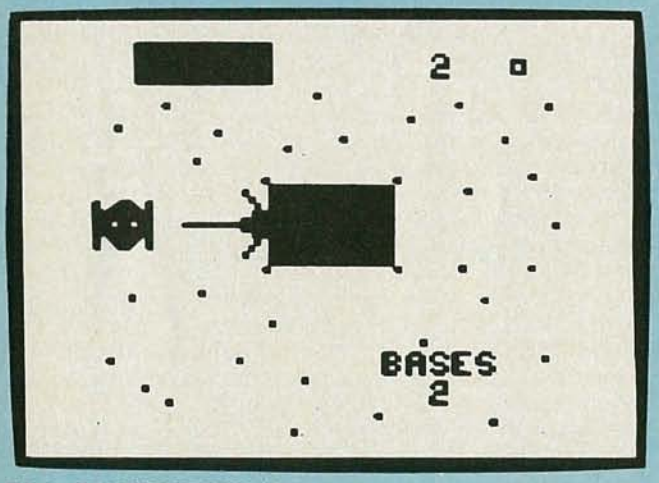


FIG. 2—SPACE FORTRESS video game.

With it a user has full graphics capability. Commands such as CIRCLE, BOX or LINE can be used for direct creation of graphics. A further indication of the potential of ZGRASS, which is also the operating system for the unit, can be shown by having a user draw a figure running across the screen using a peripheral light-pen. The user first draws the picture he wants with the pen and then indicates movement by moving the pen across the monitor screen. The figure will then run across. By using the SNAPSHOT command, the figure can be stored in memory for later recall. That indicates that there is a high degree of memory mapping in the 160 by 100 display. With this high degree of display memory-mapping, formatting displays for video games becomes easy.

The *Bally Arcade* has an interesting variation in its memory mapping. Rather than using the upper left-hand corner for 0-0 coordinates, the *Arcade* uses the center of the screen for its 0-0 location, and everything is determined from there. That makes it easier for a user to create and store graphics, rather than beginning in the upper left, which can complicate things. The graphic display is stored in memory under a macro (user-created) name.

Another indication of the power of the graphics system is typeface creation. A user has the ability to create an infinite variety of typefaces.

The *ZGRASS-32* includes two RS-232 serial input/output ports for data exchange. Data storage can be increased by adding up to two cassette recorders. Unlike other slow-speed cassette program-load systems, this one operates at a high 2000-baud rate. When a particular file is found, a menu of that particular file is displayed.

Apparently, Astrovision has plans to make this unit a full-capability personal computer. The reason is for suspecting that is that it will also interface with disk-storage systems, ranging from mini-floppies up to Winchester drives. The smallest Winchester drives provide a 5-megabyte storage capability.

As we mentioned, the *ZGRASS-32* system includes 32K of ROM and 16K of RAM. That RAM can be increased to a full 64K, allowing the use of the optional *CP/M* operating system. With *CP/M*, the *Arcade* with the *ZGRASS-32* add-on can become a versatile small business computer as it would be possible to use many of the most popular business programs.

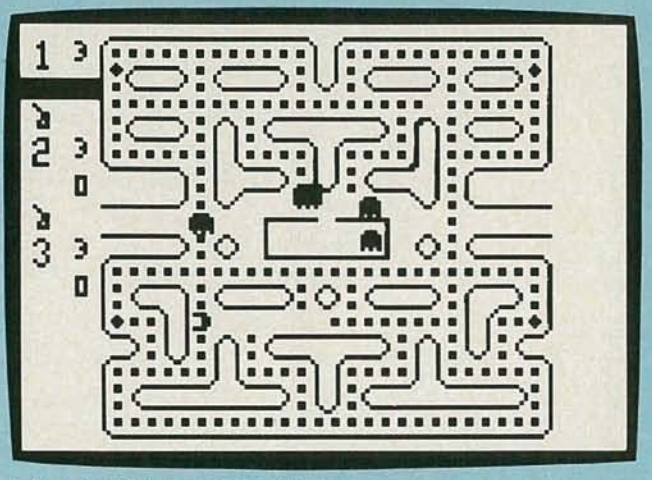


FIG. 3—MUNCHIE, a version of Pac Man.

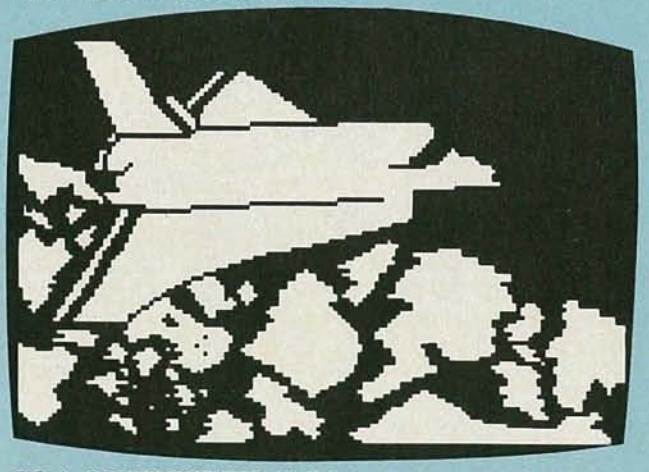


FIG. 4—CREATE PICTURES with Coloring Book with Light Pen.

Software that runs under *CP/M* ranges from word processing to accounting to database management.

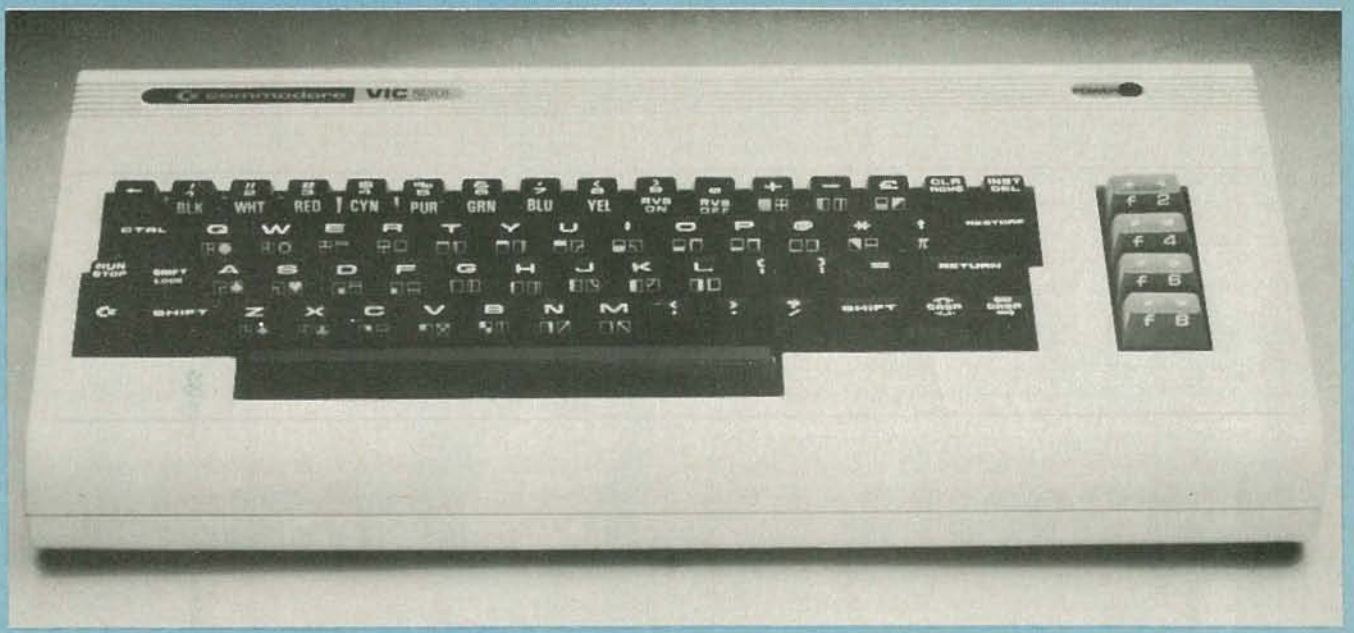
No video-game-based personal computer would be complete without some type of voice-synthesis feature, and Astrovision's *Bally Arcade* is not an exception. Included in the *ZGRASS-32* add-on is a Votrax voice-synthesis IC.

Since the *ZGRASS-32* unit has not yet appeared on the market, the pricing and distribution plans are not finalized. However, the add-on keyboard for the video unit is projected to carry a suggested retail list of $599.95. The add-on unit includes the extra ROM and the keyboard. It was unknown at press time whether there would be a printer, or what type of disk system would be available. **R-E**

Commodore VIC20

Looking for a full-featured, expandable computer system at a reasonable price? The Commodore VIC-20 may be right for you.

MARC STERN



COMMODORE BUSINESS MACHINES (681 MOORE ROAD, KING OF PRUSSIA, PA 19406) has always been a leader in the personal-computer field. For example, that company has been credited with helping to start the personal-computer revolution by developing the first system that you could simply uncrate and plug in. Now, Commodore is continuing its reputation as a leader with the introduction of the *VIC 20*, a full-featured, expandable computer at an affordable price.

Don't be fooled by the suggested list price of $299.95—this is a sophisticated machine. The heart of the unit is a 6502 eight-bit microprocessor. It comes with 5K of user-available RAM; that can be expanded to 32K using plug-in cartridges. The *VIC 20* has an extended BASIC built into its operating system. That version of BASIC is a powerful language; it allows for features such as animated graphics and string manipulation, and offers such high-level commands as PEEK and POKE. Other interesting features of the extended BASIC are tape file-handling commands such as OPEN, INPUT#, PRINT#, and CLOSE. (We'll look at the *VIC 20's* cassette-tape system shortly.)

The *VIC 20* has a 65-key typewriter-like keyboard, which is set up to simplify the generation of both text and graphics. Four keys, located on the right-hand side of the keyboard, are programmable-function keys. They are unused when the computer is first turned on, but any BASIC command or instruction set can be assigned to them; that is done under program control. The keyboard also features special function keys. They include: CONTROL; RUN/STOP; RVS ON and RVS OFF (a user can reverse characters on the display from white on black to black on white); CLR/HOME (which either returns the cursor to its upper left position or clears the display); INST/DEL (that key simplifies editing); RESTORE (reset) and keys that move the cursor.

The INST/DEL key is one of the more interesting ones on the keyboard. When using that key it is possible to insert or delete a single character, rather than having to re-enter an entire line. That is a significant feature that makes editing or correcting your program much easier.

The computer is designed to get a beginner up and programming within a relatively short time. To help reach that goal, a step-by-step instruction manual and programming guide is supplied with each unit. The operating system also generates a series of English-language prompts and error messages that help make the machine even easier to use.

As with other systems, a cassette-tape recorder can be used to load and save programs. For that purpose, Commodore sells a high-quality cassette recorder called a *Datassette*. The *Datassette*, and several other peripherals for the system, are shown in Fig. 2. Incidentally, you pretty much have to use their recorder, because a special plug is used to connect the recorder to the computer.

One notable feature of the cassette system is the VERIFY command. That command allows you to check whether a program has been saved correctly on the tape. That feature is helpful because on some other systems the only way to check a program is to load it into the computer. If the recording is defective due to bad spots on the tape, low recorder voltage, etc., loading it into the computer will wipe out the good program in the computer's memory and the program will have to be re-enterd manually from scratch. If that's ever happened

FIG. 1—AMONG THE PERIPHERALS for the *VIC 20* are an 80-column printer, the *Datassette* cassette tape recorder, and a plug-in modem for use with any modular-type telephone.

to you, you know how frustrating and time-consuming it can be. The VERIFY command saves a user from those headaches; the original program is not erased.

When a program is stored on tape, it is given a file name. When you want to retrieve that program, the *VIC-20* searches the tape for that particular file and ignores all the others. A rather "human" touch is that the computer will "talk" to you while it is searching for a file. It will let you know it is searching for a file, will give you a list of the files on the tape, and finally announce when it has found the file.

One other feature of the cassette system is a rather unique method of recording data. Rather than recording it just once at a seemingly high 1,000-baud rate, the system actually records everything twice to insure the reliability of the data file when it reads back. When the tape is read, both versions are read, which effectively slows the data rate to 500 baud.

If you prefer using a disk-based system for saving your programs, Commodore offers the *VIC 1540* single disk drive, shown in Fig. 2. That disk drive allows you to store up to 170K on a standard 5½-inch floppy disk. The disk system is read/write compatible with Commodore's *PET* and *CBM* systems.

As we said earlier, the system's 5K of memory can be expanded to as much as 32K with the appropriate RAM car-

FIG. 2—FOR THOSE WHO PREFER a disk-based mass storage system, Commodore manufactures the *VIC 1540* single 5¼-inch floppy disk drive.

tridges. Cartridges are available with 3K, 8K, or 16K or RAM. A single cartridge can be plugged directly into the unit. For further expansion you'll need to purchase an expansion module. Up to six devices can be plugged into that module. Those devices include memory, program, and interface cartridges. Among the interfaces available are an RS-232 interface for modems, printers, and other devices, and an IEEE-488 interface for attaching *PET* and *CBM* peripherals.

Commodore has many more peripherals and program cartridges besides the ones we've already looked at. Among the peripherals are an 80-column dot-matrix printer, game controllers, and a modem that can be used with any modular-type phone. The modem, called, naturally enough, the *VICMODEM*, plugs right into the user port of the computer. It operates at 300 baud and features originate/answer and full and half-duplex capabilities.

Among the program cartridges are games, personal finance, and education programs, and a word processor. Several other program cartridges are of particular interest. One of those is the *Super Expander* cartridge. Commodore has put a great deal into that package. It includes 3K of RAM memory expansion; high resolution graphics; plotting, color, and sound commands, and graphic, text, multicolor, and music modes. It also offers better resolution than the normal 176 by 184 pixels. The programmable function keys that we looked at earlier can be used with the cartridge to make editing easier. The cartridge comes with a tutorial instruction book.

Another interesting cartridge is a *Programmer's Aid*. That cartridge offers 20 new BASIC commands that will help renumber, trace, and edit BASIC programs. It will trace any program line-by-line and permits easy editing. A special KEY command lets you redefine those special functon keys for BASIC commands, subroutines, or new commands.

The last cartridge we'll look at is for those that want to program in assembly language. It is the *VICMon* machine-

FIG. 3—POPULAR SOFTWARE. Commodore provides a wide variety of popular software on cassette tape. A few of the titles are shown here.

language monitor; it helps programmers write fast, efficient 6502 assembly-language programs and includes a line-at-a-time assembler/disassembler.

There is also a wide variety of software on tape; some of the more popular titles are shown in Fig. 3. Among the tapes offered is one called *VICTerm I;* that program is a terminal emulator that converts the computer into a terminal for use with a telephone modem.

The *VIC 20* user has at his disposal several manuals to help him make the best use of his computer. One of them, the *VIC Programmers Reference Guide*, is intended for both novices and more experienced programmers. It's divided into four sections. The first is a dictionary of BASIC commands together with sample programs. The second is a layman's overview of machine-language programming. The other two sections explain how to interface the computer to a number of devices and how to program for graphics and sound.

Another is the *Introduction to BASIC Programming*, a self-instruction course using both a manual and program cassettes. **R-E**

IBM

IBM's long-awaited small computer is here and it offers a surprisingly wide number of features to both personal and business users.

MARC STERN

ONE OF THE LONG-STANDING RUMORS IN THE COMPUTER INDUSTRY over the last few years had been that of IBM's entry into the personal computer field. That rumor began surfacing as early as 1976 and continued submerging and re-surfacing over the next few years.

The rumor became fact last summer when the computer giant jumped into the personal-computer fray with its IBM (IBM Personal Computer, P.O. Box 1328, Boca Raton, FL 33432) *Personal Computer*. This entry was made even more unusual by two departures from standard International Business Machine practices. First, the huge computer firm decided to use an outside software vendor for its personal computer, rather than using its own in-house resources. This was a radical departure in itself. The second departure was that the company actively encouraged software authors to write programs for the *Personal Computer* for its new Software Publishing Division.

In the past, it had been common practice for IBM to write its own computer software and set its own standards; then the rest of the industry had to follow its lead. However, this time, apparently acknowledging the long lead other software firms have had over the last few years, and recognizing the need for speed, the computer giant has changed its tack. But, despite this encouraging turn of events, there is a small fly in this ointment—the IBM disk operating system makes it mandatory that the user employ IBM's software.

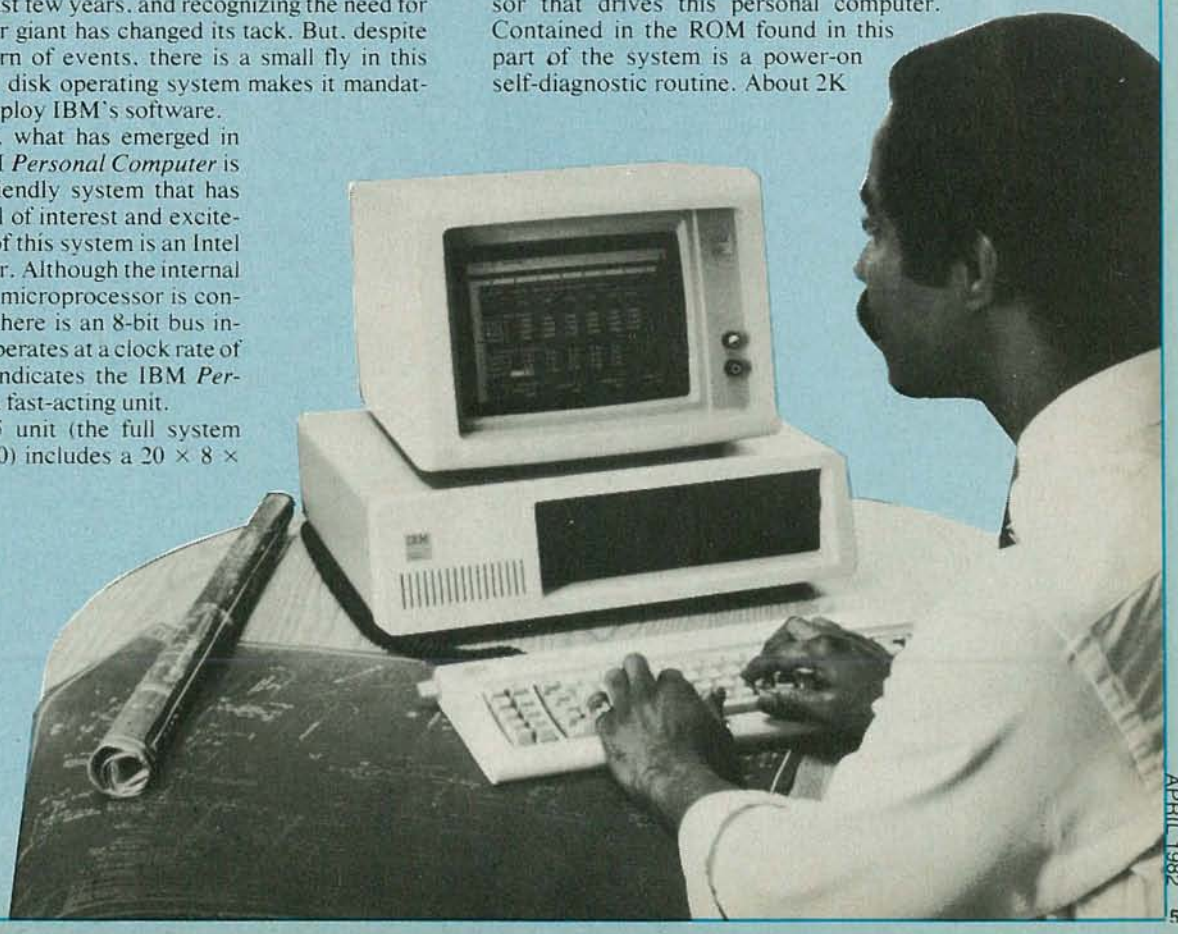
From all reports, what has emerged in the form of the IBM *Personal Computer* is a powerful, user-friendly system that has sparked a great deal of interest and excitement. At the heart of this system is an Intel 8088 microprocessor. Although the internal architecture of this microprocessor is configured as 16 bits, there is an 8-bit bus interface. The CPU operates at a clock rate of 4.77 MHz, which indicates the IBM *Personal Computer* is a fast-acting unit.

The basic $1,565 unit (the full system lists at about $4,500) includes a 20 × 8 × 2-inch sloping keyboard, that weighs 6 pounds. It has 83 full-function keys for text and data entry and includes 10 keys for numeric entry and cursor control. There are 10-special function keys for scrolling, editing and other purposes. All told, there is easy access to 256 ASCII and special characters. The keyboard is detachable from the System Unit. This latter part contains 40K of Read-Only Memory that holds the operating system and BASIC, as well as 16K of user-accessible Random Access Memory. The System Unit is the heart of the Personal Computer.

An interesting feature of the keyboard, is the 6-foot coiled cord that connects it to the System Unit. With this cord, it is possible to have a very wide work area. All the keys repeat automatically when held down.

Mass storage expansion is available by adding double-density, single-sided 5¼-inch disks. Up to 160 K-bytes can be stored on each disk, allowing for a total storage capacity of 320K-bytes. These disks are contained in the System Unit, that also houses a power supply, fan, the cassette input and output ports, and a speaker for musical programming.

The System Unit, as noted, contains the 8088 microprocessor that drives this personal computer. Contained in the ROM found in this part of the system is a power-on self-diagnostic routine. About 2K



IBM'S *PERSONAL COMPUTER* can be used in the office or schoolroom, or at home, for self-instruction or simply for pleasure.

of the ROM is used for this.It checks all parts of the unit, including the microprocessor itself. Any problems are reported to the user. The unit also contains the BASIC language interpreter and the 16K RAM. This is a fairly compact unit too. It measures 20 inches wide, 16 inches deep and 5½ inches high. By using memory add-ons (plug-in modules) the RAM can be expanded to 256K, which makes this quite a powerful system when fully configured.

Although the user has the option of adding his own printer, IBM offers its own dot-matrix unit. It is an 80-character-per-second unit that also runs its own self-diagnostic to assure proper operation. Twelve typefaces are available for various printing needs and features include page spacing and column skipping for word processing and column applications. It is a bi-directional unit for increased speed and can print 40, 66, 80 or 132 characters-per-line. It has a replaceable ribbon cartridge and print head.

The 11½-inch optional cathode-ray-tube display uses a green-phosphor screen. The 720-by-350-pixel resolution level and wide bandwidth produce a sharp, stable display. The display is 25 lines deep by 80 characters wide and includes capabilities for underlining, high-intensity blinking characters and a inverse video for highlighting information. There are also upper and lower case letters displayed for word processing and brightness and contrast controls for reading comfort. An interesting feature of the display is the potential for non-display of an area of the screen a user might consider sensitive. For users not wanting the IBM monitor, the System Unit outputs NTSC video, so you can connect the System Unit to television monitor or, using an RF modulator, to a standard TV receiver.

The IBM's *Personal Computer* also features extensive color and graphics capabilities. It can display alphanumerics using 16 foreground, and eight background, colors. In the graphics mode, four colors are available. Its medium-resolution graphics display allows an array of 320 by 200 pixels. In the high-resolution mode that increases to 640 by 200 pixels.

Communications Ability is available through the use of an asynchronous communication line. This makes it easy to interface the IBM *Personal Computer* with databases (*The Source*, *MicroNet*, *CompuServe*) other computers, laboratory instruments or any other devices with a standard RS-232C asynchronous adapter. It is reported to be programmable and compatible with different bit and parity rates.

Other optional features include the addition and use of joysticks and paddles.

There are three versions of BASIC available for the IBM *Personal Computer*. These are based on the popular Microsoft BASIC. The cassette level BASIC is included in the ROM of every system and provides all the input-output instructions needed to enter and retrieve data. It supports the use of the keyboard, display, light pen and printer and provides a full complement of editing and mathematical functions. It also allows the user to program the user-definable special function keys and will also display the function of each definable key, although this feature can be defeated.

The other two levels of BASIC—disk and advanced—are optional. The disk extension supports the use of disk, while adding date, time of day and communications capabilities to the system. The advanced extension enhances the display graphics to include features such as POINT, CIRCLE and GET/PUT display, while increasing light pen and joystick support for design work and home entertainment.

Disk BASIC is part of the IBM disk operating system and requires 32K of RAM, while the advanced level requires even more RAM—in the area of 48K. Interestingly, the disk BASIC also provides support for the system's musical functions when the PLAY command is used.

The disk operating system itself, which supports one or more disk drives, allows the user to write or read from the system's removable disks, display a directory and rename, erase, display or copy files. It is similar, but not exactly the same, as the popular operating system CP/M, which is found in many personal computers. This effectively restricts the user to IBM-supplied software—you can't use the large number of CP/M-based programs that are on the market. However, this situation should be rectified soon. IBM has been working with Digital Research, the creators of CP/M, to make the operating system available on the *Personal Computer*. It has also been working with SofTech Microsystems, Inc. to make the advanced UCSD p-System available. These two changes should provide the opportunity for current applications software to be moved over to the IBM *Personal Computer* with minimal changes.

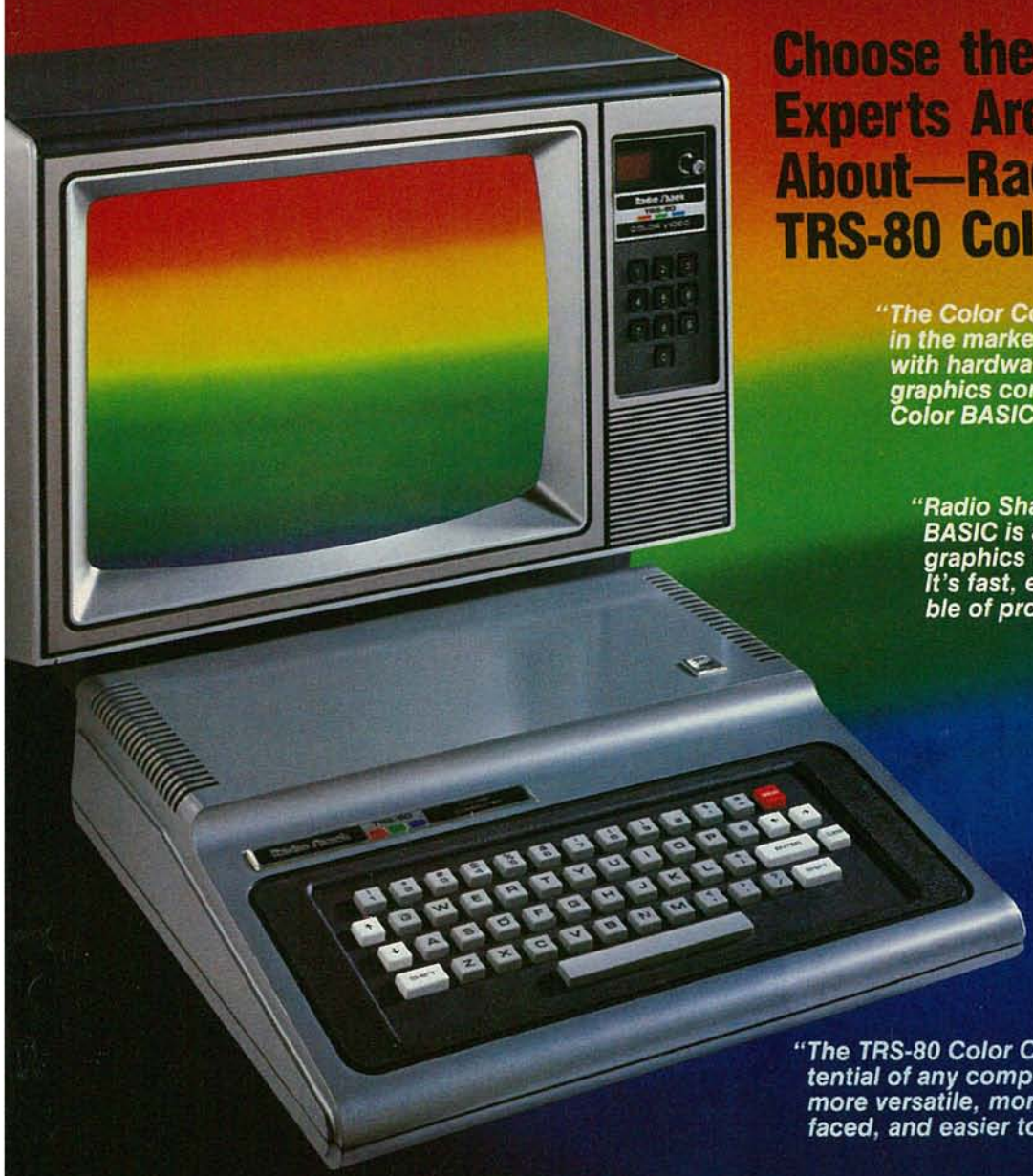
This personal computer also has another powerful language tool, a Pascal compiler. This language compiler allows separate compilations of program elements for maximum system performance. It also supports several programming features for advanced programming work.

A broad range of applications software is currently available from IBM for its new system. It includes the problem-solving program package for financial or mathematical forecasting and computations, *VisiCalc*; Peachtree Software's *General Ledger*; an accounts receivable and an accounts payable package, also from Peachtree Software; *EasyWriter*, a word processing package from Information Unlimited Software, Inc; Microsoft *Adventure*, and communications utilities.

The communications package is set up so the IBM *Personal Computer* will be able to communicate with larger systems. IBM intends to provide a full subset of 3270 emulation capabilities. As a result, this microcomputer is a good choice for both the hobbyist and the business system user and means that it can be interfaced with existing mainframes for data exchange.

And, speaking of data exchange, the Asynchronous Communications Adapter will support a baud rate of up to 9,600. This means you get a rapid and high-order data exchange.

Overall, the IBM *Personal Computer* seems to be a powerful tool for both the hobbyist and serious business user. Even though IBM entered the personal-computer fray late, it looks like its representative on the front lines of this battle is a potent contender. The competition will have its work cut out for it. **R-E**



Choose the System the Experts Are Raving About—Radio Shack's TRS-80 Color Computer!

"The Color Computer is THE best buy in the market today. It's jam-packed with hardware capabilities and the graphics commands of Extended Color BASIC are excellent. Look at it!"
—William Barden, Jr., Author

"Radio Shack's Extended Color BASIC is a breakthrough in color graphics for personal computers. It's fast, easy-to-use, and capable of producing striking graphics."
—Stan Miastkowski
BYTE Magazine

"Anyone seeking a machine that will serve well as a teacher and a base for a home communication and entertainment center will find the TRS-80 Color Computer a strong contender."
—Carl Warren
Popular Electronics

"The TRS-80 Color Computer has the most potential of any computer on the market today; more versatile, more expandable, easily interfaced, and easier to use—bar none!"
—H.C. Pennington, Author

The experts agree . . . the Radio Shack TRS-80® Color Computer lets you create spectacular high-resolution color graphics at an amazingly low price!

A Programmer's Dream at Only $599. Our 16K Color Computer with Extended Color BASIC uses one-line commands to make the creation of sophisticated graphics easy. Now you can produce color diagrams, business and engineering charts, even simple animation with a screen resolution of up to 49,152 programmable pixel points (256x192). You get eight vivid colors, sound, a 32x16 screen format, editing, user-definable keys, error messages, PEEK, POKE, a built-in RS-232 serial interface, and much more.

We Don't Skimp on Documentation. Our easy-to-understand tutorial manuals on our Standard and Extended Color BASIC will quickly have you using the full potential of your TRS-80 Color Computer.

Instant Fun and Games! Just plug-in a Program Pak™ for exciting game action! Our TRS-80 Color Computer connects to any standard TV set, including our $399 TRS-80 Color Video Receiver.

Expand At Any Time. Add a disk drive for only $599, and get 156K-bytes of on-line storage. Add more memory, a printer, and other accessories, too.

See It Today! Come in for a demonstration at a Radio Shack Computer Center, store or dealer near you.

Tell me more! Please send me your free TRS-80 Computer Catalog.

Mail To: Radio Shack, Dept. 82-A-438
1300 One Tandy Center, Fort Worth, Texas 76102

NAME _____

COMPANY _____

ADDRESS _____

CITY _____ STATE _____ ZIP _____

Radio Shack®

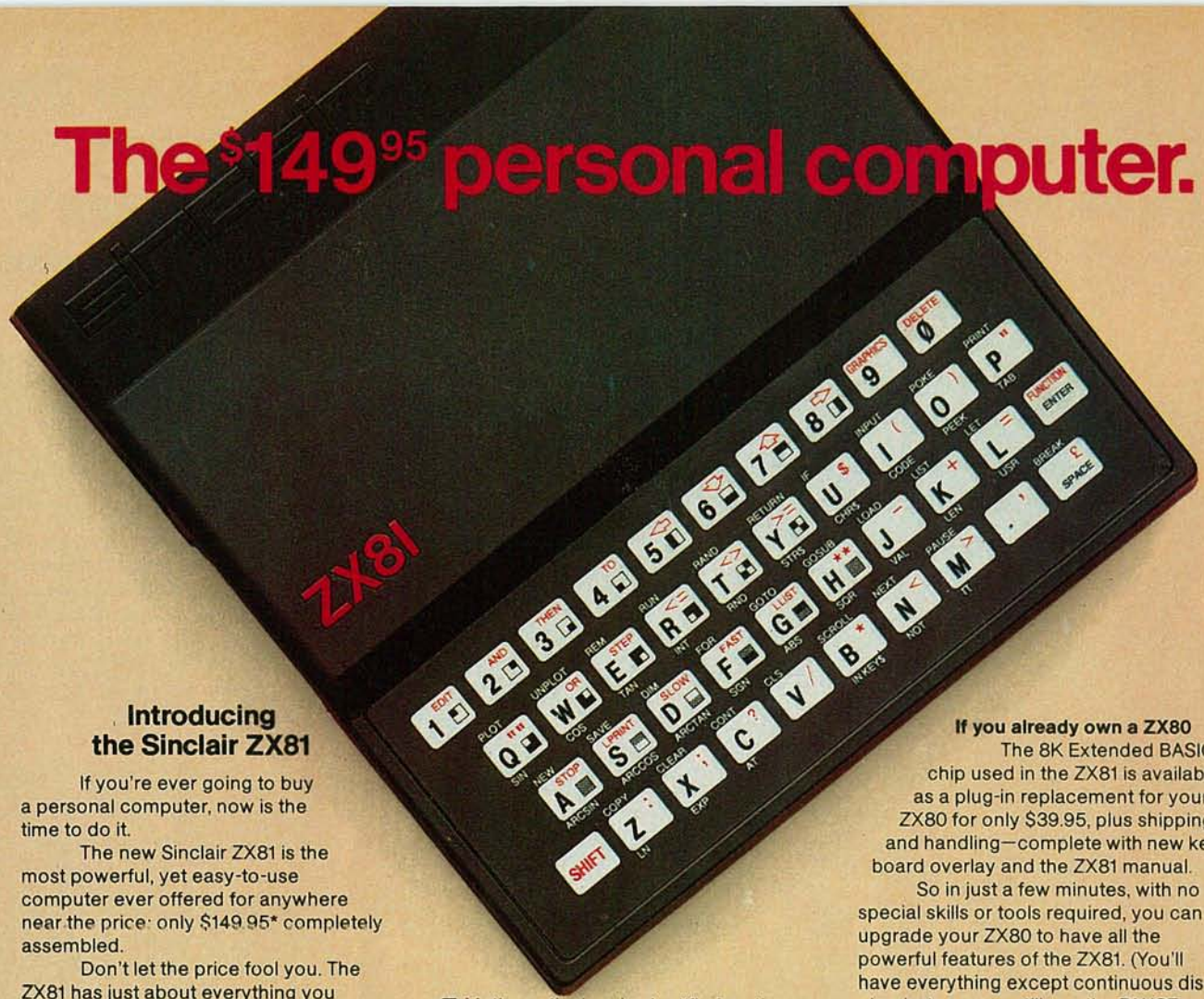
The biggest name in little computers™

A DIVISION OF TANDY CORPORATION

Retail prices may vary at individual stores and dealers. All quotes used with permission. BYTE, May, 1981, © 1981, BYTE Publications, Inc. Popular Electronics, July, 1981, © 1981, Ziff-Davis Publishing Co.

CIRCLE 5 ON FREE INFORMATION CARD

The $149.95 personal computer.



Introducing the Sinclair ZX81

If you're ever going to buy a personal computer, now is the time to do it.

The new Sinclair ZX81 is the most powerful, yet easy-to-use computer ever offered for anywhere near the price: only $149.95* completely assembled.

Don't let the price fool you. The ZX81 has just about everything you could ask for in a personal computer.

A breakthrough in personal computers

The ZX81 is a major advance over the original Sinclair ZX80—the world's largest selling personal computer and the first for under $200.

In fact, the ZX81's new 8K Extended BASIC offers features found only on computers costing two or three times as much.

Just look at what you get:

- Continuous display, including moving graphics
- Multi-dimensional string and numerical arrays
- Mathematical and scientific functions accurate to 8 decimal places
- Unique one-touch entry of key words like PRINT, RUN and LIST
- Automatic syntax error detection and easy editing
- Randomize function useful for both games and serious applications
- Built-in interface for ZX Printer
- 1K of memory expandable to 16K

The ZX81 is also very convenient to use. It hooks up to any television set to produce a clear 32-column by 24-line display. And you can use a regular cassette recorder to store and recall programs by name.

*Plus shipping and handling. Price includes connectors for TV and cassette, AC adaptor, and FREE manual.

If you already own a ZX80

The 8K Extended BASIC chip used in the ZX81 is available as a plug-in replacement for your ZX80 for only $39.95, plus shipping and handling—complete with new keyboard overlay and the ZX81 manual.

So in just a few minutes, with no special skills or tools required, you can upgrade your ZX80 to have all the powerful features of the ZX81. (You'll have everything except continuous display, but you can still use the PAUSE and SCROLL commands to get moving graphics.)

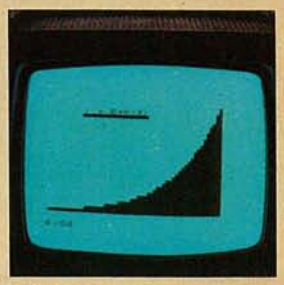
With the 8K BASIC chip, your ZX80 will also be equipped to use the ZX Printer and Sinclair software.

Order at no risk**

We'll give you 10 days to try out the ZX81. If you're not completely satisfied, just return it to Sinclair Research and we'll give you a full refund.

And if you have a problem with your ZX81, send it to Sinclair Research within 90 days and we'll repair or replace it at no charge.

**Does not apply to ZX81 kits.



NEW SOFTWARE: Sinclair has published pre-recorded programs on cassettes for your ZX81, or ZX80 with 8K BASIC. We're constantly coming out with new programs, so we'll send you our latest software catalog with your computer.

ZX PRINTER: The Sinclair ZX Printer will work with your ZX81, or ZX80 with 8K BASIC. It will be available in the near future and will cost less than $100.

16K MEMORY MODULE: Like any powerful, full fledged computer, the ZX81 is expandable. Sinclair's 16K memory module plugs right onto the back of your ZX81 (or ZX80, with or without 8K BASIC). Cost is $99.95, plus shipping and handling.

ZX81 MANUAL: The ZX81 comes with a comprehensive 164-page programming guide and operating manual designed for both beginners and experienced computer users. A $10.95 value, it's yours free with the ZX81.

The $99.95 personal computer.

Introducing the ZX81 kit

If you really want to save money, and you enjoy building electronic kits, you can order the ZX81 in kit form for the incredible price of just $99.95.* It's the same, full-featured computer, only you put it together yourself. We'll send complete, easy-to-follow instructions on how you can assemble your ZX81 in just a few hours. All you have to supply is the soldering iron.

How to order

Sinclair Research is the world's largest manufacturer of personal computers.

The ZX81 represents the latest technology in microelectronics, and it picks up right where the ZX80 left off. Thousands are selling every week.

We urge you to place your order for the new ZX81 today. The sooner you order, the sooner you can start enjoying your own computer.

To order, simply call our toll free number, and use your MasterCard or VISA.

To order by mail, please use the coupon. And send your check or money order. We regret that we cannot accept purchase orders or C.O.D.'s.

CALL 800-543-3000. Ask for operator #509. In Ohio call 800-582-1364. In Canada call 513-729-4300. Ask for operator #509. Phones open 24 hours a day, 7 days a week. Have your MasterCard or VISA ready.

These numbers are for orders only. For information, you must write to Sinclair Research Ltd., 2 Sinclair Plaza, Nashua, NH 03061.

sinclair

AD CODE 04RE

	PRICE†	QTY.	AMOUNT
ZX81	$149.95		
ZX81 Kit	99.95		
8K BASIC chip (for ZX80)	39.95		
16K Memory Module (for ZX81 or ZX80)	99.95		
Shipping and Handling	4.95		$4.95
		TOTAL	

MAIL TO: Sinclair Research Ltd., One Sinclair Plaza, Nashua, NH 03061.

NAME _____

ADDRESS _____

CITY/STATE/ZIP _____

† U.S. Dollars

Osborne

Considering the competition in the microcomputer field, this would not appear to be the best time for a young, small company to introduce a new computer. That is, of course, unless it has a product that is truly different.

MARC STERN

ENTERING THE MICROCOMPUTER FIELD IS A RISKY BUSINESS at any time. There are market forces to contend with, the challenge of bringing out the product on time, and the challenge of staying in business altogether. That has never been more true than today, now that two of the giants in the computer field—IBM and Xerox—have introduced their own personal computer. Add to that the strong competition from firms already established in the field, such as Radio Shack and Apple, and it's easy to see that it's a highly competitive market.

But market conditions did not deter Dr. Adam Osborne or his company, Osborne Computer Corporation (26500 Corporate Avenue, Hayward, CA 94545), from introducing their *Osborne I* computer. There are two factors that make that system unique. The first is that it is a complete turnkey system—from plug, to hardware, to software. The second factor is that the whole system is priced at less than $2,000, with the basic system carrying a suggested retail price of $1,795.

At the heart of the portable, full-featured unit is a powerful Z80 microprocessor. The system includes a full 64K of RAM. There is also 4K of bank-switch ROM containing the operating system. The access time for the programmable memory is 250 nanoseconds; it is 350 nanoseconds for the ROM.

The *Osborne I* comes with a detachable 70-key ASCII keyboard that connects with the monitor unit through a 10-inch ribbon cable. That makes the *Osborne I* convenient to use because the user can separate the keyboard from the monitor unit and move it nearer to where he is working.

The beauty of the machine, aside from its instant start up and run capability, is its size. It measures 20.5 × 13 × 9 inches and closes like a small suitcase. That means that the user can take it and use it anywhere (see (Fig. 1).

Another feature of the unit is its built-in video monitor. There is one serious problem with that monitor, however: It is extremely small. While the small size—3.55 × 2.63 inches—may be adequate for field work, in most cases it just won't do for other applications. However, there is an accessory 12-inch monitor available. The display system is memory mapped, and features full scrolling.

But it takes more than a monitor to make a full-featured, fully integrated system. For example: The *Osborne I* has two built-in 5¼-inch disk drives. That isn't an add-on, but is

part of the entire package. You can store up to 100,000 characters of data on each disk. The disks have 40 tracks each and 10 sectors per track. There are 256 bytes per sector. The maximum seek time is 12 milliseconds from track-to-track.

The *Osborne 1* also includes several built-in interfaces. One of them, a serial RS-232C interface, is used to connect the computer with serial printers, modems, or other devices. It is software-switchable, with a data rate of either 300 or 1,200 baud. It features handshaking to control the transmission rate. A nine-pin plug is included for use with an external modem. An adapter allows the connection of both a modem and printer simultaneously. Another interface is an IEEE-488 general purpose instrumentation bus for data communications with test instruments.

In an interesting departure from industry norm, Osborne is supplying EPROM's with its units in order to protect its software vendors. They are being used presently just for controlling various drive functions, but they could be encoded so that a machine's serial number would appear on the software if it were copied.

As a turnkey (plug-it-in-and-go) system, the *Osborne 1* is supplied with some sophisticated, powerful software. For starters, the disk-operating system is the industry-standard *CP/M*. Built into *CP/M* are such functions as a PIP (*P*eripheral *I*nterchange *P*rogram) that provides for file transfer between devices and disk files, and an ED text editor that allows the creation and modification of ASCII files. The ASM, fast 8080 assembler, also included in the *CP/M*, uses standard Intel mnemonics and pseudo operations with free-format input, conditional assembly, and assembly-line expressions. There is also a DDT (*D*ynamic *D*ebugging *T*ool) that contains an integral assembler/disassembler module that patches and displays memory in either assembler mnemonic or hexadecimal form, and traces program execution with full register and status display.

Some of the commands available include SUBMIT, which allows a group of *CP/M* commands to be batched together in a single SUBMIT command for submission to the operating system; a STAT command that displays and alters input-output device and file status, and a LOAD command that converts Intel hex format to absolute binary. In addition, the SYSGEN command will generate a new CP/M system diskette for backup purposes.

There are other powerful tools included in the software, including two forms of BASIC. Apparently not believing that one system of BASIC was enough, Osborne has included both the CBASIC (Digital Research) and MBASIC (Microsoft) languages.

CBASIC is a commercially oriented compiler/interpreter designed for use with *CP/M*. It consists of the compiler, the run-time monitor, and a cross-reference listing of all variables used in a CBASIC source program (XREF.COM).

MBASIC, on the other hand, is a BASIC interpreter for *CP/M* systems and is aimed more toward general computer applications. It supports many enhanced features and includes a line editor as part of the interpreter.

Because of the use of MBASIC, the *Osborne 1* has the potential to be a great personal computer. Through the PEEK and POKE commands, the user has direct access to memory. The EDIT command makes program editing easy, and provides an edit mode for subcommands.

Also included in the standard software is a *WordStar/MailMerge* package. That package, which is *CP/M*-compatible, provides an extensive text-processing and standard-form producing system. In addition to word processing and standardized forms, *MailMerge* lets you merge separate files and data into a single document. The standard package also includes *SuperCalc*, a management-oriented productivity tool. For a computer user who has had little or no programming expertise, that provides a way to manage and manipulate data interactively.

FIG. 1—A TRULY PORTABLE, all-in-one computer system the *Osborne 1* folds up like a small suitcase so that it can be easily transported.

There are few peripherals available since so much is included in this very-complete package. We've already mentioned the 12-inch monitor—which I think is almost necessary for serious computing. A modem cable is available for use with an acoustic coupler. There is also a battery pack available that is good for one or two hours of operation. (That power source, while useful for field work, also has a place in the office. It can provide system backup in case of power failure and could very well keep the system up and running at a crucial moment.) Mass storage capability can be increased markedly by using the accessory double-density disk drives. Those increase mass storage to 200K per drive. Further mass storage is available by using a Corvus hard-disk system, which allows the computer to be used for a broader range of industrial and commercial applications.

An optional *Micro-Link* program is available that lets microcomputers communicate over telephone lines with each other, large computer systems, and terminals. The program, which requires a minimum system memory of 16K, allows the *Osborne 1* user to contact data bases, bulletin boards, and time-sharing services.

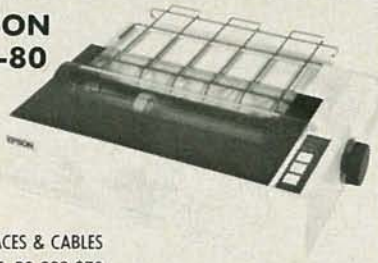
Osborne has some aggressive marketing plans. It plans to compete head-to-head with the computer giant, IBM, in terms of units in the field. IBM is said to be targeting to have five million of its personal computers in the field over the next five years, and Osborne is aiming at the same figure.

Osborne Computer Corp. has charted an ambitious course for itself. Only time will tell if this relatively new company will succeed in that aim, or whether the industry skeptics who have said it can't be done are right. **R-E**

APRIL SPECIALS

WE CARRY THE COMPLETE LINE OF ATARI SOFTWARE, PERIPHERALS AND ACCESSORIES.

EPSON MX-80



INTERFACES & CABLES
IEEE $55. RS-232 $70.
APPLE INTERFACE & CABLE $90.
TRS-80 CABLE $35.

$439.00

ATARI 400 16K

ATARI

$339.00

PERSONAL COMPUTERS

Televideo 925	849.00	NEC 3510 Spinwriter	$1750.00
Televideo 950	929.00	NEC 3515 Spinwriter	1795.00
CBM 8032 Computer	1149.00	NEC 3530 Spinwriter	1750.00
CBM 8050 Disk Drive	1349.00	NEC PC-8023 Printer	629.00
CBM 4032 Computer	975.00	NEC 7710 Spinwriter	2345.00
CBM 4040 Disk Drive	975.00	NEC 7720 Spinwriter	2695.00
CBM 4022 Printer	649.00	NEC 7730 Spinwriter	2345.00
CBM VIC-20	269.00	NEC JC 1201 M(A) — Color 12" monitor	359.00
Microtek 16K Ramboard for Atari 800	79.00	Okidata Microline-82A	499.00
Microtek 32K Ramboard for Atari 400 and 800	149.00	Okidata Microline-83A	729.00
Qume Sprint 9/45 (Full Panel)	2095.00	Diablo 630	1995.00
Atari 810 Disk Drive	449.00	M & R Sup-R-Terminal	279.00
Atari 850 Interface	169.00	Microsoft Soft Card (Z-80)	279.00
Atari 830 MODEM	159.00	Microsoft Ramcard 16K for Apple	149.00
Atari 800 16K	749.00	Hazeltine 1420	799.00
Epson MX-70	319.00	Amdek 100 G	169.00
Epson MX-80 FT	549.00	Anadex DP-9500/9501	1249.00
Amdek Video-300	199.00	Televideo 910	559.00
Hayes Micromodem II (Apple II)	299.00	Televideo 912C	669.00
Sanyo 9" B & W	189.00	Televideo 920C	689.00

NEC GREEN 12" MONITOR JB 1201M
$159.00

OKIDATA MICROLINE 80A MATRIX PRINTER
$329.00

AMDEK COLOR-1 MONITOR
$339.00

EPSON MX-100 FT PRINTER
$729.00

PRICES ARE SUBJECT TO CHANGE WITHOUT NOTICE

CALL TOLL FREE!

EAST COAST
1-800-556-7586
COMPUTER SHOPPING CENTER INC.
12 Meeting Street
Cumberland, RI 02864
1-401-722-1027
TELEX 952106

WEST COAST
1-800-235-3581
COMPUTER SHOPPING CENTER INC.
3533 Old Conejo Rd. #101
Newbury Park, CA 91320
1-805-499-3678
CA. TOLL FREE 1-800-322-1873
TELEX 182889

COMPUTER SHOPPING CENTER

We Accept C.O.D.'s • Stock Shipments Same Day or Next • No Surcharge for Credit Cards
• All Equipment Factory Fresh w/MFT Warranty • We Carry the Complete Line of Personal Software
• Prices do not Reflect Shipping Charges
Rhode Island and California Residents please add 6% Sales Tax

CIRCLE 12 ON FREE INFORMATION CARD

NEC

Nippon Electric Company's (NEC) PC-8000 computer is aimed at the business and high-end personal user.

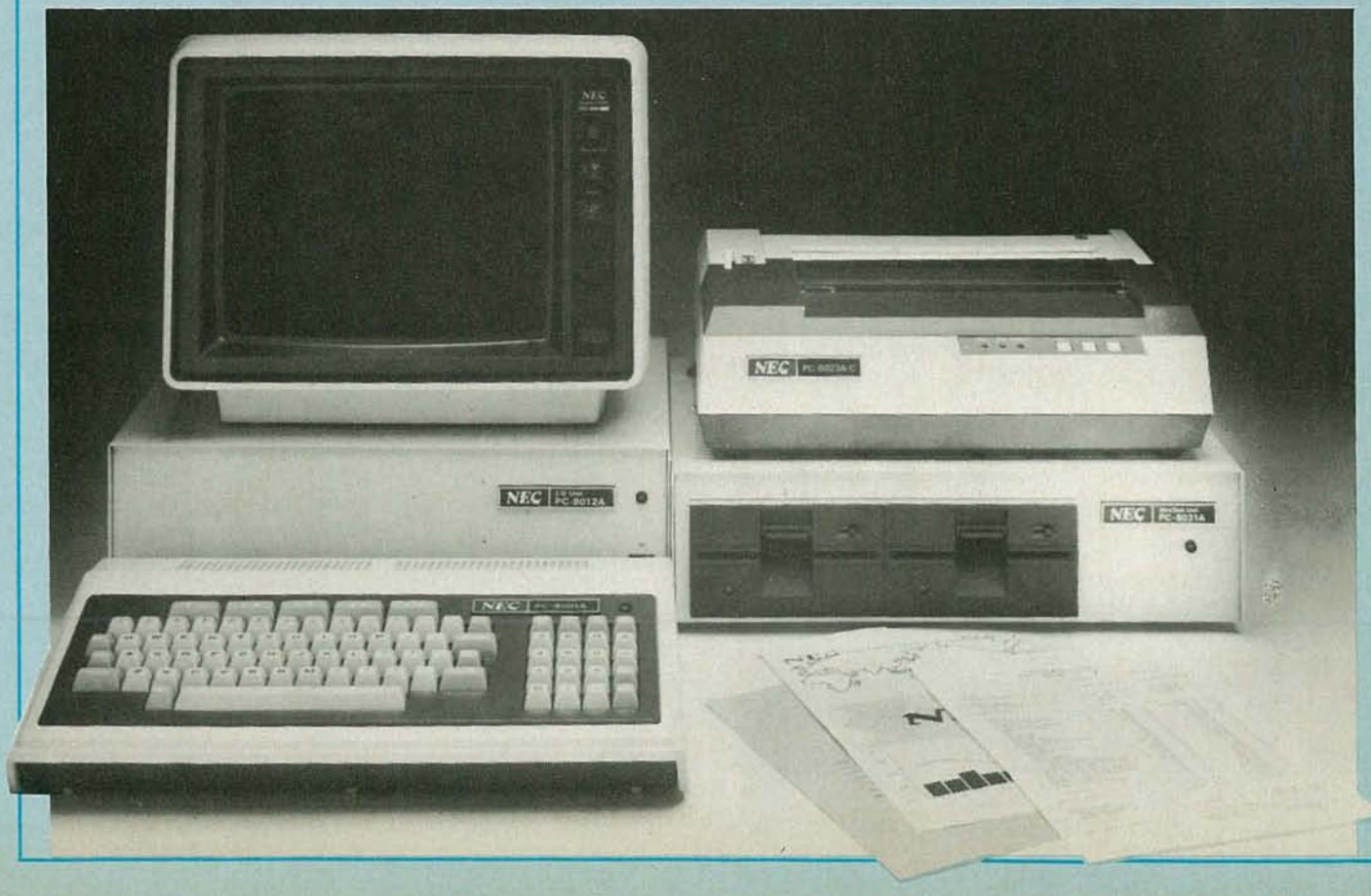
MARC STERN

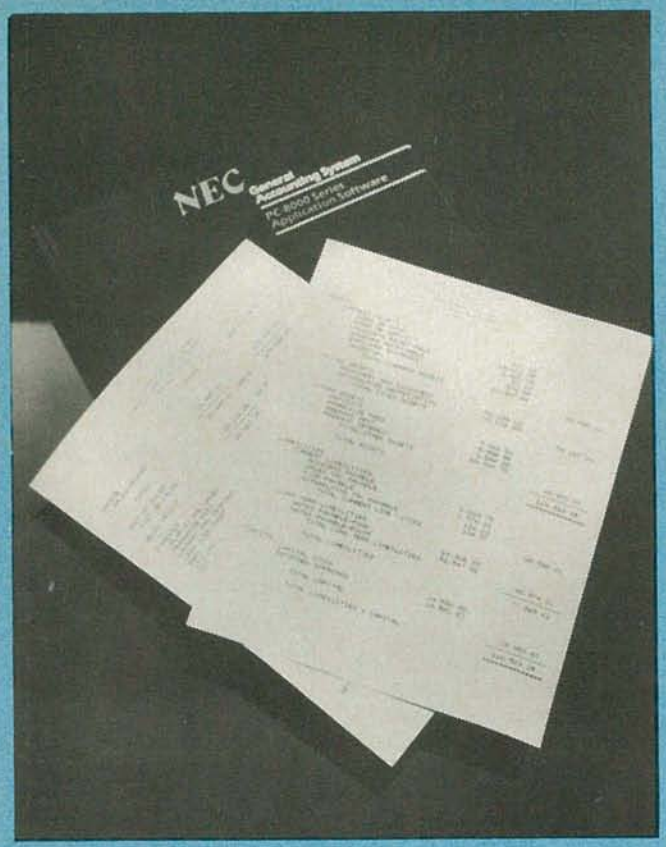
NEC HOME ELECTRONICS (USA) (1401 ESTES AVENUE, ELK GROVE Village, IL 60007), is another of the many companies competing in the hotly contested microprocessor market. In fact, this microcomputer company is trying a very aggressive marketing strategy, aiming at several markets at once: the home computer market, the corporate market, and the small business market.

With such an ambitious plan of attack, what features does its system offer that set it apart from the many other systems on the market? One thing the NEC system offers that no other system on the micro market seems to offer, is a 12,000-word dictionary built right into its word-processing software package. This feature aside, the other functions of the NEC *PC-8000*-series microcomputer system are comparable to the rest of the microcomputer market.

The heart of this microcomputer system is a Z80A-equivalent microprocessing unit (MPU). It is housed inside the *PC-8001A* keyboard unit. The MPU runs at a clock speed of 4 MHz. The keyboard, with full ASCII capability, has 82 keys and includes a numeric keypad for rapid data entry. Five of the keys are user-programmable and can define 10 functions. These keys are conveniently located at the top of the keyboard, rather than at the side as in other systems, which means that the operator doesn't have a long reach to access them. But the dual nature of the keys may present a problem. Since they are capable of supporting 10 functions, it could be hard to access the functions that need control key or shift access because that adds another step to an otherwise smooth process. Perhaps smaller single keys should have been considered rather than five large dual-function keys.

There is no built-in cathode-ray terminal, although two different 12-inch monitors are available: one color and the other black-and-white. The color monitor can display 20 or 25 lines. Line length, as in other microcomputers, is variable, allowing 36, 40, 72 or 80 characters per line. A powerful editor simplifies programming and includes four-way cursor control and character insertion and deletion. There are 248 characters available which can be presented in any one of eight colors. Resolution on the screen, while not as high as that of some others on the market, is 160 by 100, and more than adequate. In the word-processing mode, N-key rollover input allows high-speed typing. An 8-pin connector is used for the keyboard-monitor interface. The display also includes a built-in sound system that greatly expands the range of





A GENERAL ACCOUNTING SYSTEM package is just one available from NEC. Others include a report manager and word processor.

possible uses.

The basic *PC-8001* keyboard unit has 25K of read-only memory. Resident in this ROM is another of the many sons of Microsoft BASIC, N-BASIC. There is also 32K of random access memory available. This memory can be expanded to 64K when the *PC-8012* input/output unit is added.

Mass storage is available with dual floppy-disk drives. The drives are interfaced with the keyboard unit through a parallel port in the rear. This same port supports the expansion interface unit, the *PC-8012A*.

The expansion interface is the heart of the expanded system. It features 2K of read-only memory. Interestingly, this ROM is available as programmable read-only memory chips, which are optional. That could make it possible to tailor this unit more closely to a user's needs. The basic random access memory in the input/output unit is 32K. It is expandable to 128K with additional plug-in boards. There are seven slots available for expansion boards and eight priority levels can be implemented with up to 16 real-time interrupts possible. The I/O unit will also support the disk drive unit. There is also another parallel port available for further system expansion.

NEC's uses a parallel printer port. The printer is a dot matrix, bi-directional 100 character per second unit. The matrix is 7 by 9 for English characters, 8 by 8 for graphic characters and 8 by 8 for dot-graphic printing. It is a full ASCII printer with complete upper and lower case capability. It also includes graphics capability for Greek, mathematic and graphic characters.

As mentioned earlier, the *PC-8000* system can address up to 32K of memory in its standard form, and this is expandable to 128K by simply adding module boards. The resident N-BASIC is a version of the many Microsoft variants on the market and includes such commands as PRINT-USING and IF-THEN-ELSE. It also allows double-precision floating-point mathematics, flexible and powerful graphics commands, plus the ability to invoke TERMinal with a single command. Interestingly, the ability to program in a BASIC language that is compatible with the nearly universally accepted industry standard Microsoft BASIC should give the user access to many existing specialty programs. The user can also access various parts of the memory with the POKE command that will put data into specific memory locations. It is part of the resident monitor. The only departure from the normal list of commands is the lack of a PEEK command to allow the user access to what is already in various memory areas. This could be a drawback if the user already has crucial information in one area and then overwrites it with new information because he didn't know anything was already there.

For the beginning computer enthusiast, NEC offers a guide called *Creative Programming*. This book is designed to help early users become comfortable and confident with microcomputer operation in four lessons. It was developed in cooperation with Eastern Illinois University and designed by Creative Programming Inc. It covers such basics as an introduction to computer language instructions and the significance of the BASIC functions: how to write a program, and how to load a program. Each lesson is accompanied by a series of exercises that allow the user to experience first-hand the concept presented.

The *PC-8000* uses the CP/M operating system and, because it does, a wide range of software is available. This is especially important if the user would like to expand the capabilities of the system, although NEC does offer a wide range of software of its own. NEC software packages include a new microcomputer program with three active calculating dimensions; Report Manager. This program creates and instantly updates a complete variety of reports for financial, accounting, engineering and scientific applications. Unlike other two-dimensional reporting systems, this program generates business reports such as income statements, balance sheets and sales forecasts. It can also produce bar charts, a feature not included in other electronic spread sheet products. Also these reports can be created from any place in the X, Y and Z axis "data cube" generated by the program.

There are other software packages available that include a General Accounting Package and the Word Processing package with the unique built-in 12,000-word dictionary.

The basic *PC-8000* system lists for under $6,000 and includes the color monitor, input-output expansion interface unit, the keyboard/Z-80A unit and two 5¼-inch disk drives. The disks, probably single-density, provide mass storage capability of up to 100K.

It's interesting to note that nowhere in any of NEC's literature is there any mention of interfacing this unit with a phone modem for data network access, although this is probably possible by adding the expansion interface. Since the unit is set up for parallel I/O, this would require a modem capable of supporting parallel, rather than serial, I/O. This is an omission that should be clarified because the system is touted as being capable of working in a small business or corporate environment. If there is no phone or network interface then it's hard to see how this system might interface with a company's existing system or how it might transmit information from a satellite office to the mainframe in the home office.

The *PC-8000* is advertised as a word-processing color-computer system, and indeed it is. But, it is hard to see how its color capabilities can be fully realized in a work environment. In a home environment, where data isn't printed out in black-and-white report form, it is a nice feature—in a $6,000 package—but in an office environment it is hard to see how this feature will be used. In a scientific environment, though, it is a valuable aspect of the system.

NEC has entered the hotly contested microcomputer market with the first color word-processing system. It's a different approach from the manufacturer of the leading personal computer in Japan. **R-E**

Sinclair ZX81

Small size and a low price do not have to restrict a computer's capabilities.

MARC STERN

CLIVE SINCLAIR IS THE DEVELOPER OF THE FIRST UNDER-$200 eight-bit microcomputer programmable in BASIC. He is credited with developing the first pocket calculator 10 years ago and with the development of the first miniature television, Microvision, six years ago.

It has always seemed that when someone said that something couldn't be done, Sinclair has set out to do it. So it was only natural that he brought out a full-featured eight-bit microcomputer for under $200.

That was the Sinclair *ZX80*, introduced in 1980. However, as observers noted at that time, it had display problems and a very limited memory function. Those problems, though, have been corrected in the updated *ZX81*, which was released in Boston late last year. The unit is being marketed by Sinclair Research Ltd., 50 Staniford St., Boston, MA 02114.

The basic *ZX81* consists of a 40-key, pressure-sensitive keyboard; a built-in VHF RF modulator; 1K of static RAM memory, 8K of ROM, and generates a black-on-white display (when connected to the antenna terminals of a standard TV receiver) of 24 lines by 32 characters. The heart of the system is an eight-bit Z80A central-processor unit.

The power requirements of the computer are slight. It requires 9-volts DC at about 700 mA. Although a wall-plug type supply comes with the unit, you can also, if you choose, use your own. The supply need not be regulated. (While the computer-end plug of the supply provided by Sinclair will also fit into one of the device's tape jacks, no damage, we are assured, can take place if this is done.)

The updated *ZX81* is aimed at the beginner. The target is the person who wants to learn what computers are about and how they work. As such, it is an interesting building block for the novice computer user. It is a way to approach computers without being intimidated by them.

While its graphics capabilities have been improved, the *ZX81* is still limited to a rather large dot matrix configuration. The line length is short, as compared to other units. The *ZX81* is interfaced to a TV receiver by the RF modulator, which is one of the inhibiting factors in the display. To be fair, though, the graphics capability has been upgraded.

The computer has two keyboard-selectable operating speeds: COMPUTE AND DISPLAY and FAST. The first provides a continuous display, with the computer doing the actual computing during the TV's vertical-blanking interval. The second allows the computer to run its program most of the time, displaying information only at the end of the program, or at other specific intervals.

The result is that, while the computer runs about four times faster when in the FAST mode than when in the other, most of the time you are left staring at a blank display-screen.

Apparently answering criticisms raised when the unit was first introduced, Sinclair has opted for more power in the CPU. That includes Sinclair BASIC resident in the ROM that emulates the *Apple II* BASIC.

The BASIC interpreter allows multi-dimensional string and numerical arrays, while at the same time featuring floating-point arithmetic. That provides a range of 3×20^{-39} to $+7 \times 10^{+38}$, which is accurate to 9 digits.

That indicates that there is some scientific functionality built into the unit. The *ZX81* will also handle full log, trig, and their inverse functions.

It is a user-friendly machine that features automatic syntax-error detection and program editing. It also offers randomized functions that are useful for both game and serious applications.

There is no disk-operating-system available with the *ZX81*. Instead, it relies on program loading and saving through a serial cassette-recorder interface. However, it operates at a high enough baud rate to allow random file access under a master name. That puts the *ZX81*'s operating speed in the 1,000-baud area. No special recorder is required, the company says, so a beginning computer hobbyist can use his own.

In its basic configuration, the *ZX81* is equipped with only 1K of static RAM memory. However, that can be expanded by 16K via a module which plugs into the rear of the unit.

The keyboard is easy to use in that many of the keys serve

multiple functions and have the BASIC statements assigned to them. In other words, there are various single-stroke capabilities built in. That feature makes the machine a good choice for the beginner. For instance, when using the monitor program, a user can PEEK or POKE information into specific memory locations at the touch of one key. The same is also true of such statements as THEN, TO, GOTO, GOSUB and AND. As noted, it is also possible to edit with a single keystroke, and it is possible to use one keystroke for PRINT, LIST and RUN.

Those easy functions should prove a boon to the beginning computerist, but they indicate that the system could have been more powerful. It seems there is more capability hidden in the microcomputer than this particular system allows.

One drawback that many people who are comfortable with standard keyboards may find is the use of a plastic-membrane type of keyboard. Granted, the 43-character keyboard does offer a total of 91 built-in functions (when the inverse key is hit) but it is still a membrane unit. When other manufacturers have tried that approach, they have met with little success and have had to return to typewriter-type keyboards.

When you first look at the *ZX81* it looks like a very modest microcomputer. Weighing in at only 12 ounces and with dimensions of 6 × 6.5 × 1.5 inches, it is very little larger than a coffee-table book. Yet its developer is quick to defend its capabilities. When the new version of the *ZX81* was introduced at a Boston press conference late last year, Sinclair said that it was "...not a reduced-support machine. The language it uses is complete."

One of the reasons given for the ability to keep the cost down was Sinclair's policy of doing only the tooling, development technology, and design work in house. The manufacturing is done for Sinclair by Timex in Dundee, Scotland.

Not only does Sinclair offer the plug-in memory-expansion module; it also offers its own cassette software. That includes business and household programs, as well as educational and game programming. However, outside software houses are also being encouraged to write programs for the *ZX81*.

Aside from the plug-in module, there is only one other peripheral offered: the *ZX* Printer. That is a dot-matrix unit which is rather noisy. The *ZX81* comes with a built-in printer interface.

Since this microcomputer is a learning tool, one would expect it to come with a programming course—and it does. Sinclair provides a 164-page elementary programming course.

Not only does the manual provide a rather thorough explanation of how to program the *ZX81* in BASIC, but its later chapters and appendices supply other useful information as well.

There is information on machine-language programming, which can make programs run much more quickly than in BASIC. A listing of system variables along with their memory locations is provided—knowing what information is stored where can make program debugging easier, and can also allow you to make your programs run with more style and efficiency.

There is even a section for people who already are familiar with BASIC explaining the differences between Sinclair's BASIC and others.

The Sinclair *ZX81* is priced at $149.95 in wired form. For the adventurous person, who wants to learn computers from the ground up, there is a $99.95 kit available. The memory-expansion module is priced at $99.95 and the serial dot-matrix printer is available for less than $100.

In the final analysis, the *ZX81* is a building-block unit. It is intended to train people who know little or nothing about computers. That is all it is intended to be.

If you already own a *ZX80* and want to upgrade it, a plug-in replacement ROM that contains the new 8K extended BASIC used in the *ZX81* is available. That is a user-installable IC that can be installed in the *ZX80* in a few minutes by using a screwdriver. That will allow the *ZX80* to use the new *ZX* Printer and it will also be software-compatible with programs intended for the *ZX81*. The replacement ROM and a new overlay for the *ZX80* keyboard sell for $39.95. **R-E**

Pocket Computers

Once thought of as an impossible fantasy, a true computer that you can slip into your pocket is now a reality. Four such units are now available, with two more on the way.

MARC STERN

THE CONCEPT OF A POCKET COMPUTER BEGAN TO APPEAR IN science fiction stories, such as those by Isaac Asimov, over 30 years ago. Like many of the other ideas proposed in those stories, it was dismissed as pure speculation by most readers. But, again like many of the devices described in those stories, the pocket computer has become a reality.

The first hint that the pocket computer was indeed on its way came in the early 1970's with the introduction of hand-held calculators. While those first units were relatively simple, and rather expensive, that soon began to change.

Calculators acquired features such as memory, which allowed the user to store a single value that could be recalled or modified at any time, and functions such as roots, logs, powers, and trigonometic functions; that made the calculators much more useful in engineering, mathematical, and scientific applications. Soon after, calculators with individually addressable memories became common.

In the late 1970's, programmable calculators were introduced. The term "programmable" referred to the calculator's ability to "remember" a sequence of key strokes, and reproduce them on command. Some even had the ability to store the "program" on a small piece of magnetic material, so that it would not be lost when the unit was shut off. It was even possible to purchase plug-in ROM's for those units. Some of those ROM's were programmed to perform complicated calculations; others were programmed to—what else?—play games. But even a programmable calculator with all of the "bells and whistles" is not a computer.

That brings us to the subject of this article: true "pocket" computers. Pocket computers are sold in this country by four companies—Radio Shack (One Tandy Center, Ft. Worth, TX 76102); Sharp (10 Sharp Plaza, Paramus, NJ 07652); Panasonic (One Panasonic Way, Secaucus, NJ 07094), and Quasar (9401 W. Grand Ave., Franklin Park, IL 60131). Those units are not much larger than some of the early calculators that we talked about, but they are far more complex and powerful.

The Radio Shack/Sharp pocket computers

If the Radio Shack *TRS-80* (see Fig. 1) and Sharp *PC-1211* pocket computers seem similar, there is a good reason—both, although sold and serviced by different companies, are functionally identical. Both measure $^{11}/_{16} \times 6\frac{7}{8} \times 2\frac{3}{4}$ inches and feature a 57-key mini-keyboard, 1.9K of RAM, and 11K of internal ROM. Two proprietary four-bit CMOS microprocessors form the heart of the units. One of the microprocessors handles the arithmetic operations and the display routines; the other handles the BASIC interpreter and the input from the tiny keyboard. The units use a 24-character LCD display; each character is formed by a 7-by-5 dot matrix.

The pocket computers let you store several different programs in memory. Those programs are individually identified, and any one can be run by simply pressing a specified key. A 1,424-step memory is automatically partitioned for program and data storage, and there is a 25 data-element memory and a 48-step reservable memory for storing functions.

Although those are relatively tiny devices, they have many

FIG. 1—THE *TRS-80* POCKET COMPUTER is shown here with the slide-in printer and cassette interface. That peripheral turns the pocket computer into a complete "mini-system."

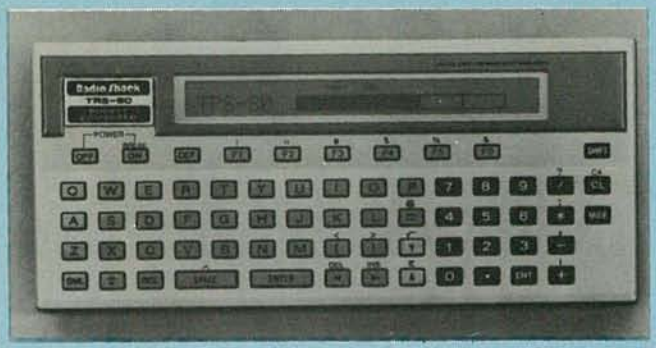


FIG. 2—AMONG THE FEATURES of the new *TRS-80 model PC-2* is a new, more powerful version of pocket BASIC.

of the features of larger units: for instance, an edit and debug mode that should make finding programming errors and correcting them easy. That function is handled by the monitor program; the monitor is contained in 4K of the ROM.

What sets the pocket computer apart from programmable calculators is that you can write and run your own programs in BASIC. The BASIC used by those devices is called pocket BASIC and appears to be related to Radio Shack's *TRS-80 Level I* BASIC, although there are considerable differences. The language is capable of supporting up to four nested subroutines or FOR/NEXT loops, and up to 15 levels of parenthesis. It can handle strings of up to seven characters.

Programs and data are entered using the 57-key keyboard. Although that keyboard is laid out much like a typewriter keyboard, don't expect to be able to touch-type; the unit's small size makes that all but impossible. If you prefer, Radio Shack offers a library of software, available on cassette. Among the subjects covered are real estate, personal finance, aviation, and games. To load those programs, or to save your own on cassette, you'll need one of the two peripherals available for the units—a slide-in cassette-tape interface.

The second peripheral is a dot-matrix printer that also includes the cassette interface. That peripheral turns a pocket computer into a complete mini-system. With that device, it is possible to get hard copies of programs, data, or results. It uses the pocket computer's PRINT and LIST commands to provide a 16-column alphanumeric printout on ordinary electronic-cash-register-type paper. The printer writes at approximately one line per second, and is powered by rechargeable nickle-cadmium batteries.

The *TRS-80 Pocket Computer* sells for $169.95. The suggested retail price for the Sharp *PC-1211* is $179.00.

As we went to press, both Sharp and Radio Shack announced the introduction of a new, more powerful pocket computer. The new units, the Radio Shack *TRS-80 model PC-2* (Fig. 2) and the Sharp PC-1500 (which, again, are functionally identical), feature a single 8-bit microprocessor and 2.6K of user-available RAM, expandable to 6.6K. The computer uses an advanced version of the pocket BASIC described earlier, and can handle two-dimensional arrays and strings of as long as 80 characters. For display, a fully addressable 7 × 156 dot-matrix LCD is used; both upper and lower case characters can be generated by the unit. Among the peripherals planned is an RS-232 interface. Currently available is a combination four-color printer/plotter and cassette interface.

The Panasonic/Quasar *HHC* pocket computers

The *HHC* pocket computer, developed jointly by Matsushita and Friends Amis of San Francisco, is sold in this country by both Panasonic (see Fig. 3) and Quasar; although the packaging is somewhat different, here is another case of two functionally identical computers.

The basic machine is available with either 2K or 4K of RAM. That user-available RAM is expandable, using external modules; up to six 8K modules can be connected to the computer at one time. The unit comes with 16K of internal ROM, which can be expanded to 48K.

The microprocessor used by the computers is a 6502. Special circuitry is used to keep the microprocessor "dormant" until it is called on to perform some task. That feature prolongs the time between battery charges (rechargeable nickle-cadmium batteries are used to power the unit and most peripherals), making the use of the 6502 possible. Without the circuitry, the microprocessor would discharge the batteries in about two hours. Other special circuitry is used to let the *HHC* retain programs and data after the unit is switched off.

The operating-system language used in those computers is SNAP, which is derived from FORTH. SNAP is used for maximum efficiency, as it is a fast-running, compact language. Programming can be done in either SNAP or BASIC, if the appropriate internal ROM is installed. The internal set of applications programs include a four-function calculator, a free-form file system, and an editor. Programming is done using the 65-key, typewriter-like keyboard. Any of the keys

FIG. 3—SOME OF THE PERIPHERALS available for the Panasonic *HHC* pocket computer, as well as the device itself, are shown here. Among them are an RS-232 interface, a modem, a color-TV interface, and a small printer.

can be redefined by the user when needed.

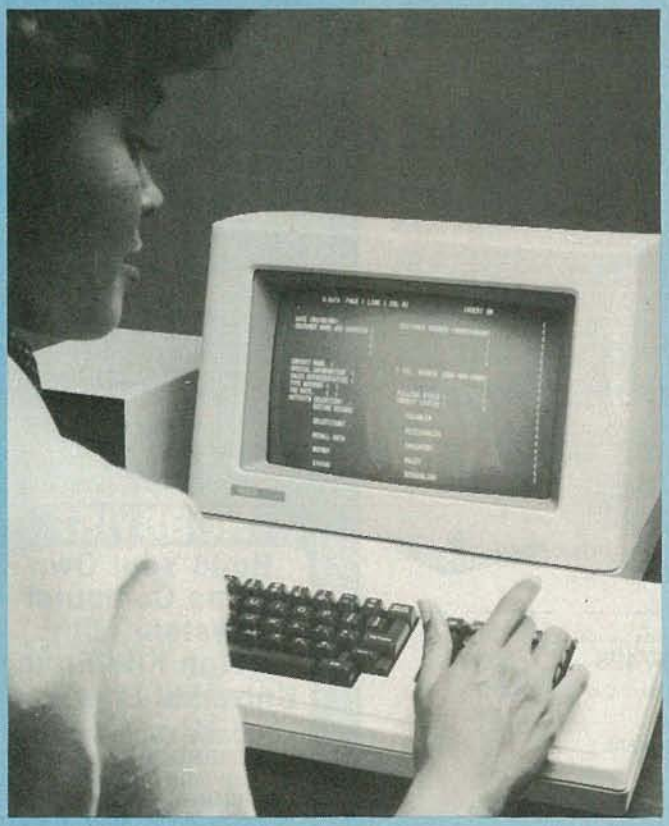
A 44-pin connector is used to connect peripherals to the *HHC*. Among the available peripherals are a bus expander, which lets you connect up to six different peripherals to the computer; a 15-column thermal printer; a modem; RAM, with battery back-up so memory is retained even when disconnected from the computer, and a color-TV interface. The interface will let you use a TV to display 16 lines of up to 32 characters each, or up to 48 × 64 graphic elements in eight colors and black. A monitor output is also provided.

The suggested retail price for the Panasonic *HHC* 4K pocket computer is $500.00; the suggested retail price for the 4K version is $600.00. The suggested retail price for the Quasar 4K computer is $525.00; the suggested retail for the 4K version is $595.00. **R-E**

Xerox

The king of office copiers has entered the personal computer market. Here's a close up look at the Xerox 820.

MARC STERN



THE BATTLE IN THE HOTLY contested personal computer market stepped up its pace early last summer when Xerox introduced its long-awaited microcomputer. Rumors and speculation abounded for some time about the impending entry of the copy king into the fray. It seemed like a natural extension of this company's work in the field of office automation. After all, it wasn't very long before the introduction of the *820* computer that the company introduced the *860* information processing system.

This entry into the personal computer market added another potent force in this volatile marketplace. However, Xerox's entry wasn't aimed at the personal computer hobbyist, it was targeted primarily at the business user. Although the potential for BASIC programming is more than evident with the availability of the Microsoft BASIC included in the software, Xerox downplays this aspect of the interactive nature of this machine. Instead, it emphasizes the user-friendliness and ease of use of the *820 Information Processor.*

The heart of this menu-driven system is a Z80 microprocessor. The MPU is contained in the display cabinet. It is softloaded with 64K of Random Access Memory and 4K of Read-Only Memory. The MPU operates at a clock speed of 2.5 MHz. This is slow when compared with other units on the market.

The basic system includes a 12-inch screen that displays white characters on a dark background and has a capacity of 24, 80-character lines. It includes dual RS232 serial ports, one for the printer and one for communications. Dual parallel ports are also standard.

The 96-character ASCII keyboard can be detached from the CRT unit to allow more convenient system use. A user can move the keyboard around (within the limits of the interface cable) so that the keyboard unit is nearer the work being done. This is convenience, because the user sometimes needs that kind of portability in a work area. It is also possible to position the keyboard on a user's lap for ultimate convenience. The keyboard is laid out as a conventional typewriter keyboard, but it also includes some extra function keys, such as HELP (more about this one later). To the right of the conventional keyboard are the rest of the special function keys, plus a numeric keypaid that facilitates entering statistical and numeric information.

Dual 5¼-inch floppy disk drives are included in the basic unit. With this form of mass storage, a user can store up to 92,000 characters of data on the single-sided, single-density disks. Optional 8-inch, single-sided, single-density, dual drives, that can store 250,000 characters, are also available.

A Daisywheel printer—the Xerox *630*—is available as an option. A 40-character-per-second device, it is a bidirectional unit.

Since the operating system of the *820* is CP/M and since the use of CP/M is so widespread, there are many applications packages written for it and the user can purchase a wide variety of standard business applications programs from Xerox or other vendors.

Let's face it, the Xerox *820* is not aimed at the computer hobbyist, but at the business user who may know little or nothing about computers. Since it is aimed at this user, it is menu-driven and, to be fair, those menus lead the non-knowledgeable user through the paces quite well.

For example, suppose a user presses the C (CLEAR) command to delete a file. The screen then prompts for the name of the file to be deleted. Next it returns to the Directory Menu after the deletion. To work on an existing document or to create a new one, a user presses command "A" from the Directory Menu and the screen prompts him for the name of the file. When the file is recalled to the screen, the Directory Menu is replaced by the Main Menu. The user doesn't even have to memorize keystrokes because the menus tell him what to do.

The HELP key is a special aid to the non-computer-oriented user. With it, if the user runs into trouble, simply pressing the button will bring up more information than is normally listed on the menu. This information then leads the user through the task.

The software available for the Xerox *820* includes word processing, CP/M, *Teletype* communications, MBASIC, CBASIC-2, COBOL 80, MSort and an electronic worksheet package. This is a powerful, but **very** expensive, selection of

software. Since there is very little ROM resident in the *820*, you can see that the software must be disk-loaded.

The software expense looks something like this. The 5¼-inch word processing package costs $500, while the 5¼-inch CP/M operating system costs another $200. The 8-inch word processing package costs $500, while the 8-inch disk CP/M operating system costs $200. And, anyone who chooses to buy the Xerox software must buy at least one software package for each system that is ordered.

The system itself is by no means inexpensive either. The minimum system costs $2,995 (display/processor, keyboard and dual 5¼-inch disk drives). The 8-inch disk-based system costs $3,795. These don't apear to be the kind of figures an average computer hobbyist would spend. The user does have the option of supplying a separate printer, although Xerox offers one for $2,900. It is interesting that the higher-level information system—the *860*—offers more and easier functionality, to boot.

One of the most interesting capabilities of this system is the potential of interfacing with the Xerox *Ethernet* network. This is, essentially, a high-speed data communications network. This is done through the *872/873* Communications Servers. Through the *871* Interactive Communications Emulator, the *820* can access a host computer at another location too.

As is obvious, though, all of these potentialities are aimed at the business, rather than the hobby, user. It is clear from the type of menu-driven system that is offered and from the lack of encouragement in programming. Of course, that potential is there should the user opt for the MBASIC package. But, it is an expensive option, much more in line with a business type of investment than with a consumer purchase. **R-E**

Disk Operating Systems

If you're not in any great hurry, a cassette-tape operating system may be adequate; but for fast and efficient loading and saving of your programs, you'll want a disk-operating system.

HERB FRIEDMAN

FOR PERSONAL COMPUTERS, FLOPPY DISKS (AND, MORE RECENTLY, hard Winchester-disks) are the most rapid and convenient way to store (save) or retrieve (load) data and programs.

The basic personal-computer package almost always includes some form of cassette data-storage system. Such a system converts the computer's digital signals to audio tones that can be recorded by an ordinary, low-cost cassette-tape recorder. However, a cassette-based system is much slower than a disk-based system. Of course, "slow" is a relative term, but to give you an approximate idea of what we mean, an income tax program that seems to take forever to load from tape—actually 3½ minutes—loads from a standard disk system in less than three seconds; a BASIC interpreter that takes over seven mintues to load from cassette tape takes about five seconds to load from disk.

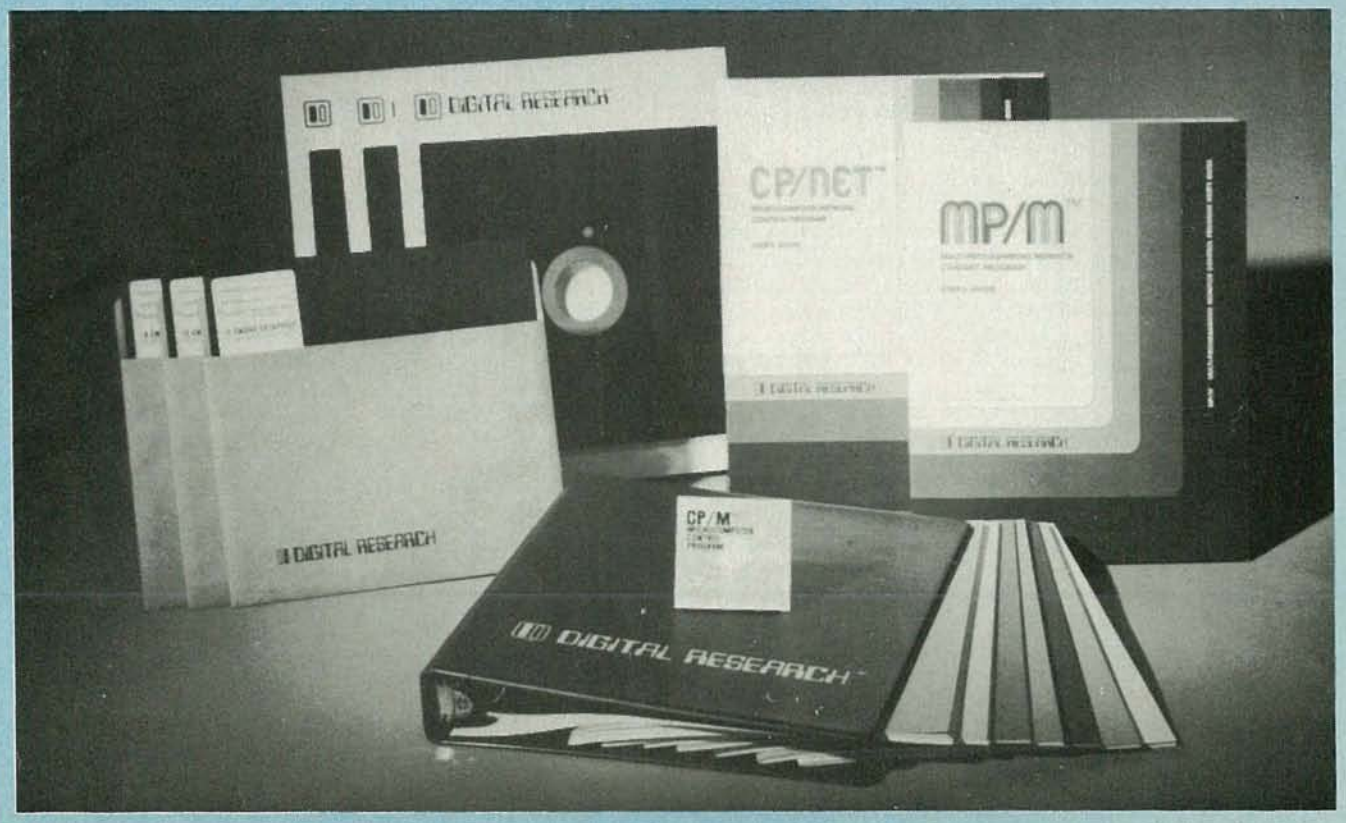
For those not in a particular hurry, a cassette-tape system often proves more than adequate, expecially when you consider that the tape system is supplied with the computer (or costs an additional $50 or so for the recorder), while a basic disk system will usually cost upwards of $500.

But even when the time required to save and load isn't critical, there is still one major drawback to tape-based systems—file handling is rather difficult. If you're running software that writes data to and reads data from the tape, you must constantly jockey the tape to find "clean" (unrecorded) tape for saves, or the location of specific files to be loaded. It is true that some tape systems will search for particular files by reading the file names; but if the file is the third, fourth, or fifth recording on the tape, it can be frustrating to wait for the computer to locate and load the file, particularly when you're attempting to update the file(s) with new data.

A disk-based mass-storage system, on the other hand, handles files much more efficiently and conveniently. Files can be saved to or loaded from disks very quickly. It is possible to acess, use, or update/modify (read and write) complete files or parts of files. You can also chain files, append them, and intersperse them.

The precise method of handling disk files is determined by what's known as a DOS (*D*isk *O*perating *S*ystem), a set of control and utility programs that may also control the computer's peripherals. Primarily, the DOS performs housekeeping chores on the disk itself, reserving and managing the space on the disk and determining how the files will be accessed by the computer.

The methods used to access files are either *sequential-*



access, whereby a unit of stored data can be accessed only by reading every unit of storage that precedes it in a file (similar to a cassette-tape file), or *random access* (also known as *direct access)* whereby a block of data is accessed through a directory located on the disk that keeps track of physical location of every file on the disk.

The DOS also determines the memory requirements for files, controls the distribution of memory for files, regulates the execution of programs and utilities, controls the interchange of files between peripherals, and even gets the computer up and running.

The exact manner in which files are handled, and the types of utilities provided, are determined by the particular DOS and the computer for which it is intended. If the DOS is a "universal" one—like *CP/M*, which we'll discuss later—it can run on any computer using a microprocessor from the 8080/Z80 family. If the DOS is written specifically for one type of computer, such as *TRSDOS* for Radio Shack computers, or *HDOS* for Heath/Zenith computers, it can be used only with those types of computers. There are many reasons for that, among them are the different ways the tracks on the disk are used, different computer memory-configurations and device drivers, etc.

WHEN A DOS IS BOOTED it generally presents a display giving a description of itself and a "prompt" for the user to follow.

Another factor that can lock a computer into a particular DOS is the way it "boots" the computer. In order to run, a computer must contain a program, but in order to load a program a computer must be running. It's a sort of "Catch 22" situation: Essentially, the computer must lift itself up by its own bootstraps. In many modern computers the boot program is in a ROM which, in turn, causes the DOS to be loaded. In others, the ROM has just enough "intelligence" to read a boot program from a disk. Then *that* program loads the DOS into the computer.

There are major differences in the way various disk operating-systems handle the transfer of information between peripherals. For example, in some computers the output to a printer, modem, or other devices (such as another terminal) is memory-mapped within the computer—the device is accessed by routing the data to be output to a certain memory-address—or is available at an I/O port. Those functions are controlled by the computer's *own* internal operating system. (In an *Apple II* computer, for example, the output to the printer is usually through I/O port #1.)

In other systems, particularly *CP/M* and *HDOS*, information interchange is through device drivers or peripheral interpreters contained in the disk operating-system. Without the DOS there would be no communications with or between peripherals, because the driver routines in those cases are not contained in ROM.

Utilities are important

In addition to taking care of file management and peripheral interchange, a personal-computer DOS will include several other utility programs. Among the most popular utilities for personal-computer systems are those that permit files to be copied from one disk to another. And, to copy files—or even just to *use* a disk—it must be formatted or initialized. Thus a DOS must also include a utility for initializing and formatting, one that will make a backup copy of a disk, one that will verify that the data written to the disk has been recorded correctly, and one that will check for system errors and inform you when they occur.

Because disks are extremely delicate—far more so than tapes—there is generally a "media test" utility to check the condition of the disk. That utility will either "lock-out" any defective tracks (by noting in the disk directory that those tracks should not be used by the computer), or will let the user know which tracks are defective so that he or she will not use them.

Once a disk is in use, the DOS provides complete file-management, providing, among other things, a directory of all the files on the disk that can include a description of the file-type and its size and attributes, and, in some cases, password protection, "kill" protection, and other useful features.

Aftermarket DOS's

Sometimes a DOS will not have all the features required by a user: the solution to that problem is to use what's known as an *aftermarket* DOS. For example, consider file copying. Although, generally, disks can be copied using only one drive, some DOS's, such as *TRSDOS*, require the use of two drives. That is one of the reasons for the popularity of *NEWDOS*, an aftermarket DOS for the Radio Shack *TRS-80;* it permits file copying with a single drive. *NEWDOS* also has a "screen print" function that lets you get a printout of whatever is displayed on the screen. There are other aftermarket DOS's offering similar, and additional, features as well as utility programs that can enhance existing DOS's.

CP/M—a "universal" DOS

Digital Research's *CP/M*, which is commonly taken to stand for *C*ontrol *P*rogram for *M*icrocomputers, is intended for computers using an 8080, 8085, or Z80 CPU, or another microprocessor capable of handling the 8080 instruction set. Its configuration in terms of the number of disk drives and number and type of peripherals is determined by the computer that it is used with. *CP/M* is used by more different personal computers than any other DOS. It is often implied that, because it is the most commonly used DOS, it is also "universal" in the sense that *any* computer running *CP/M* can use *any* *CP/M* software. That isn't exactly the case. Each computer requires a *CP/M* specifically prepared for it, or for a similar "family" of computers. Once *CP/M* is available for a particular computer, the machine can generally use software written to run under that operating system. If a particular piece of software cannot be used, the problem is most likely one of media incompatibility.

Media limits

CP/M, and software using it, is normally supplied on 8-inch, single-sided, soft-sectored, single-density disks. A number of software vendors, however, can provide the operating system and programs in a format to fit your particular system (5¼-inch hard-sectored, for example). There are versions for use on almost every computer that can follow the 8080 instruction set—and even for a few that don't.

The other major hindrance to "universality" can be the memory configuration of your computer. *CP/M* normally starts at memory location 0000H (hex). Some computers, though, already use that location (and the ones following it)

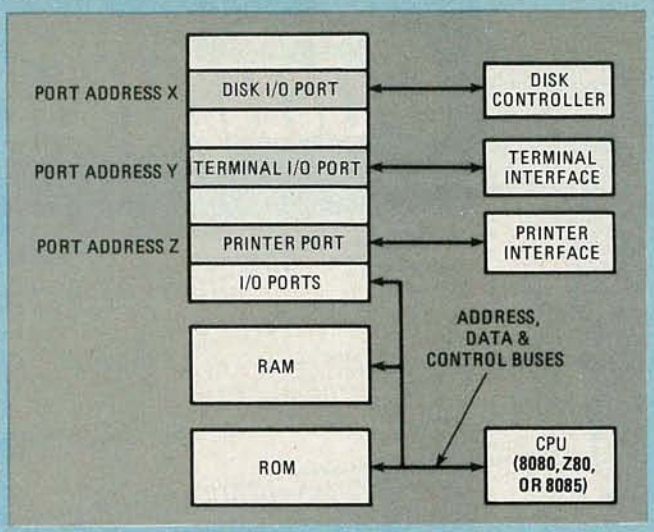


FIG. 1—HOW A TYPICAL COMPUTER SYSTEM is configured. *CP/M* eliminates the need for the programmer to worry about the exact I/O configuration of the machine by replacing the physical I/O devices with logical representations.

for their own purposes. Fortunately, customized versions of *CP/M* are available for most computers to overcome that problem.

"Logical" design

One of the problems in adapting a DOS to operate on a number of different machines is that the addresses of the I/O ports vary from machine to machine (see Fig. 1). *CP/M* solves that problem by not referring to specific addresses (in its unconfigured form). It replaces physical I/O devices with "logical" devices—programming representations of the physical devices. In effect, the DOS sees only command words and peripheral descriptions such as LPT (line printer), CON (console, or control terminal), TTY (teletypewriter), PTR (paper-tape reader), and so on.

CP/M has no idea where CON or LPT are located. Each manufacturer who takes out a license to use *CP/M* provides a BIOS (*B*asic *I*nput/*O*utput *S*ystem) that is tailored to the configuration of his computer. It is the BIOS that knows the hardware connections for the "logical" terms and recognizes, say, the command "PRINT" and directs the data that is to be printed to the proper port or memory location. If the user commands the DOS to copy a file from disk drive A to drive B, it is the BIOS that knows the I/O for the disk drives. Essentially, the BIOS serves as an interpreter between the "logical commands" and the hardware.

It is that logical structure that is primarily responsible for *CP/M's* universality, and the most important reason for its popularity. Here's an example that illustrates how it works: Assume that you are running *CP/M* on your system. When your version of *CP/M* runs a piece of software your BIOS will recognize the logical commands it contains and route data to the proper ports or memory addresses. If the software calls for an output to a printer, your own BIOS will recognize the logical instruction and see to it that the data goes to the appropriate port or address for the printer. The very same software can be used on another computer because *its* BIOS would provide the necessary hardware interpretations for the logical commands.

Should you create your own software, the program will contain only the logical commands. Assuming media compatability, you could give the disk to someone with a completely different *CP/M*-based computer and he would almost certainly be able to run the program without problems becauses his BIOS would interpret them in terms of his own computer's operating system.

Configuration

Another feature that makes *CP/M* "universal" is its flexibility regarding the use of peripherals. Each manufacturer decides specifically what peripherals and interconnects will be available through his BIOS; those can include many that were not originally intended for use with *CP/M*. A typical BIOS might include drivers for a CRT terminal, teletype, line printer, batch processor (card reader), paper-tape punch, etc. Considerable leeway is allowed as to what peripheral is used for a specific application, and the user makes the selection using a "CONFIGUR" utility. Incidentally, as is typical with many disk operating-systems for personal computers, *CP/M* comes with many commonly used utilities, including an editor and assembler.

More, more, more

As the software for personal computers becomes more sophisticated—in many instances equalling or surpassing the quality of that for the large mainframes of just a few years back—DOS's will also change and become more sophisticated to make best use of that new software. Disk operating-systems are constantly being improved, and steadily growing more complex. Even the "universal" *CP/M* may be replaced by a more versatile DOS eventually, and each innovation will generate further innovations. Every change or addition will contribute to making personal computers more powerful and easier to use. **R-E**

DOS SUPPLIERS

While nearly every manufacturer supplies a disk-operating system for his machine, operating systems are also available from many independent suppliers. The following is a partial listing of those suppliers.

ALTERNATE SOURCE
1806 Ada Street
Lansing, MI 48910

APPARAT, INC
4401 S. Tamarac Parkway
Denver, CO 80237

DIGITAL RESEARCH
PO Box 579
Pacific Grove, CA 93950

DYNAMIC MICROPROCESSOR ASSOCIATES
545 Fifth Avenue, Suite 1400
New York, NY 10017

LOGICAL SYSTEMS INC.
Mequon, WI 53092

MICRO SYSTEMS SOFTWARE, INC.
5846 Funston Street
Hollywood, FL 33023

MICRO MIKE'S INC.
905 S. Buchanan
Amarillo, TX 79101

MIDWEST SCIENTIFIC INSTRUMENTS, INC.
220 W. Cedar
Olathe, KS 66061

PHASE ONE SYSTEMS
770 Edgewater Drive, Suite 830
Oakland, CA 94621

SOFTWARE DYNAMICS
2111 W. Crescent
Anaheim, CA 92801

TECHNICAL SYSTEMS CONSULTANTS, INC.
Box 2570
1208 Kent Avenue
West Lafayette, IN 47906

Radio Electronics YOUR OWN Computer Software APRIL 1982

High-Level Basic

What started out as a "primitive" teaching language has evolved into one sophisticated enough for creating complex programs.

JOSEPH BERNARD
TECHNICAL EDITOR

THERE'S A GOOD CHANCE THAT YOU'LL HAVE PURCHASED your computer with an eye toward using it eventually to help you in your business, or in your personal affairs.

In business, your computer can perform many of the functions that you may currently be doing by hand or "farming out" to a service that uses its own computers to process the data you supply.

At home you may want to use your computer to assist you with "homework" from your business life or just to simplify your personal record-keeping. Like many other people, these days, you may supplement your regular income with work performed at home; your computer can also prove useful for that purpose.

Whatever your purpose for using a small computer for serious applications, you should be familiar with what's involved. Other parts of this special section deal with operating systems and some of the application programs that are available. Here, we'll discuss what makes those programs run.

Operating systems

The monitor or operating system contained in your computer's ROM's is essential to bringing the machine to life. Without instructions, computers can do *nothing!* The first microcomputers (and the early big computers, too) didn't have resident operating systems. When the machine was turned on, a "bootstrap" program—one that gave the computer a rudimentary sense of "intelligence"—had to be entered manually through a series of toggle switches or, in the case of big machines, through punched cards (the kind you are not supposed to "bend, fold, mutilate or spindle").

The development of permanent computer memories, particulary ROM, allowed the machine to "come up" as soon as power was applied. From that point, the user could communicate with the computer and proceed to more complex tasks.

Your computer's operating system allows you to instruct the machine to perform a number of functions. Some of the ones that you probably don't think about are accepting input from the keyboard and generating the video display.

Others tell the computer how to output to a printer and how to communicate with a cassette recorder. CP/M, discussed elsewhere in this section, is a complete disk operating-system that makes use of your computer's operating system, and, in addition, adds features of its own.

```
791:
792: REM*************************************************************
793: REM*************************************************************
794: REM -- SUBROUTINE TO PRINT SECTION HEADING:
795:
796:          REM -- CLEAR SCREEN:
797: 9500     PRINT ERASE$ : CURSOR%=FN.CURSOR2%(1,1)
798:
799:          REM -- PRINT CUSTOMER INFO & SECTION TITLE:
800:          PRINT TODAYS.DATE$;TAB(LINE.LEN%- \
801:             LEN(LEFT$(CUSTOMER$,15)));LEFT$(CUSTOMER$,15)
802:          DUMMY%=FN.MESSAGE%(TITLE$,-1,0,128,TRUE%,0,1)
803:
804:          RETURN
805:
806: REM*************************************************************
807: REM*************************************************************
808: REM -- ADD RECORD TO HISTORY FILE:
809:
810:          REM -- OPEN HISTORY BATCH FILE IF NECESSARY:
811: 9600     IF END #11 THEN 9605
812:          IF NOT HISTORY.BATCH.OPENED% THEN \
813:                 HISTORY.BATCH.OPENED%=TRUE% :\
814:                 TEMP$="A:"+LAST.CUSTOMER$+"HIST.BAT" :\
815:                 OPEN TEMP$ AS 11 :\
816:                 DUMMY%=FN.EOF%(11,0,TEMP$) :\
817:                 GOTO 9610 \
818:             ELSE \
819:                 GOTO 9610
820:
821:          REM -- HERE TO CREATE BATCH FILE:
822: 9605     CREATE "A:"+LAST.CUSTOMER$+"HIST.BAT" AS 11
823:
824:          REM -- HERE TO CONTINUE PROCESSING:
825: 9610     PRINT #11; \
826:                 TRANS.DATE$, \
827:                 FIELD.INFO$(1), \
828:                 TRANSACTION%, \
829:                 TRAIL$
830:
831:          RETURN
832:
833: REM*************************************************************
834: REM*************************************************************
835: REM -- ADD RECORD TO INVOICE FILE:
836:
837:          REM -- OPEN INVOICE BATCH FILE IF NECESSARY:
838: 9700     IF END #10 THEN 9705
839:          IF NOT INVOICE.BATCH.OPENED% THEN \
840:                 INVOICE.BATCH.OPENED%=TRUE% :\
841:                 LINE.ITEMS%=0 :\
842:                 OPEN "A:INVOICE.BAT" AS 10 :\
843:                 DUMMY%=FN.EOF%(10,0,"A:INVOICE.BAT") :\
844:                 GOTO 9710 \
845:             ELSE GOTO 9710
846:
847:          REM -- HERE TO CREATE INVOICE BATCH FILE:
848: 9705     CREATE "A:INVOICE.BAT" AS 10
849:
850:          REM -- HERE TO CONTINUE PROCESSING:
851: 9710     LINE.ITEMS%=LINE.ITEMS%+1
852:
853:          REM -- IF 1ST LINE ITEM ON THIS INVOICE THEN PRINT DATE
854:          REM -- & CUSTOMER NAME:
855:          IF LINE.ITEMS%=1 THEN \
856:             PRINT #10;TODAYS.DATE$,LAST.CUSTOMER$
857:
858:          REM -- NEXT ADD LINE ITEM DATA:
859:          TEMP=VAL(FIELD.INFO$(4))*VAL(FIELD.INFO$(6))
860:          IF VAL(RIGHT$(FIELD.INFO$(2),2)) <= FLOAT(CUT.OFF%) THEN \
861:                 TEMP=FN.ROUND(TEMP,.01) \
862:             ELSE \
863:                 TEMP=FN.ROUND(TEMP*FACTOR,.01)
864:          PRINT #10; \
865:                 FIELD.INFO$(1), \        REM -- LOT #
866:                 INVOICE.DESCRIPTION$, \
867:                 FN.ROUND(VAL(FIELD.INFO$(5))*VAL(FIELD.INFO$(6)),.01), \
868:                 TEMP, \ REM -- STORAGE PRICE
869:                 VAL(FIELD.INFO$(6)), \  REM -- QUANTITY
870:                 FIELD.INFO$(3), \       REM -- ITEM TYPE
871:                 FN.ROUND(EXTRA.LABOR,.01), \    REM -- EXTRA LABOR
872:                 EXTRA.LABOR$, \
873:                 RECEIPT$
874:
875:          REM -- IF FULL INVOICE THEN PREPARE FOR NEW ONE:
876:          IF LINE.ITEMS%=7 THEN \
877:                 LINE.ITEMS%=0 :\
878:                 PRINT #10;"$$"
879:
880:          RETURN
881:
882: REM*************************************************************
883: REM*************************************************************
884: REM -- ENTER THIS ROUTINE TO DO SPECIAL PROCESSING FOR
885: REM -- LOT'S DATA:
886:
887:          REM -- DISPATCH TO THE SPECIAL ROUTINE:
888: %INCLUDE B:PROCESS
889:
890: REM*************************************************************
891: REM*************************************************************
892: REM -- OPEN FILE ROUTINE:
893:
894:          REM -- FIRST SEE IF INITIALIZATION FILE EXISTS:
895: 9900     IF NOT FN.OPEN.FILE%(1,"B:INITIAL.FIL",0) THEN 9910
896:
897:          REM -- OK, FILE EXISTS SO LETS READ
898:
899:                 REM
900:
901:
```

A short history of BASIC

Besides coming with an operating system, your computer also contains a programming language called BASIC (*B*eginners' *A*ll-purpose *S*ymbolic *I*nstruction *C*ode). The original version of BASIC was developed at Dartmouth University in 1963 by Professors J.G. Kemeny and T.E. Kurtz to allow students to program computers easily. It was what's known as an *interpreted* language, as opposed to most computer languages of the day, which were *compiled* (more about that later).

Serious programmers tended to look down on BASIC—while it was easy to use, it was quite inefficient for their purposes. The arrival of small computers change that.

The first personal computers required all data to be entered in binary form through toggle or paddle switches. The *IMSAI* computer, shown in Fig. 1, was such a machine. As you can imagine, it was extremely difficult to accomplish anything of practical value using the switches to enter data and the LED's to read it in binary form. A practical programming language was needed.

The first BASIC for personal computers was Tiny BASIC, which occupied only a couple of thousand bytes of memory (memory was very expensive in those days—4K cost about $250.00). That Tiny BASIC, a subset of the original Dartmouth BASIC, was refined and expanded in a series of articles and forums in *Dr. Dobb's Journal of Computer Calistheics & Orthodontia*.

The first "full-blown" BASIC was known as Altair BASIC, because it was written for use on a computer known as the *Altair*. It occupied 8K of memory, and was loaded into the computer from a cassette, taking several minutes to load. It was created by a company called Microsoft.

In those early days, each computer manufacturer supplied its own BASIC. Among the early BASIC's for microcomputers were a 4K (and later 8K) BASIC for the SWTP (*S*outhwest *T*echnical *P*roducts) *6800*, a BASIC for Processor Technology's *SOL*, known as "BASC5" and occupying a little under 6K (later—much later—followed by an "Extended BASIC" that required about 16K) and *Northstar* BASIC, on disk, for Northstar's *Horizon* computer. Finally there was a language called MBASIC for Microsoft. It was really the Altair BASIC, but was usable on other computers with the appropriate operating systems.

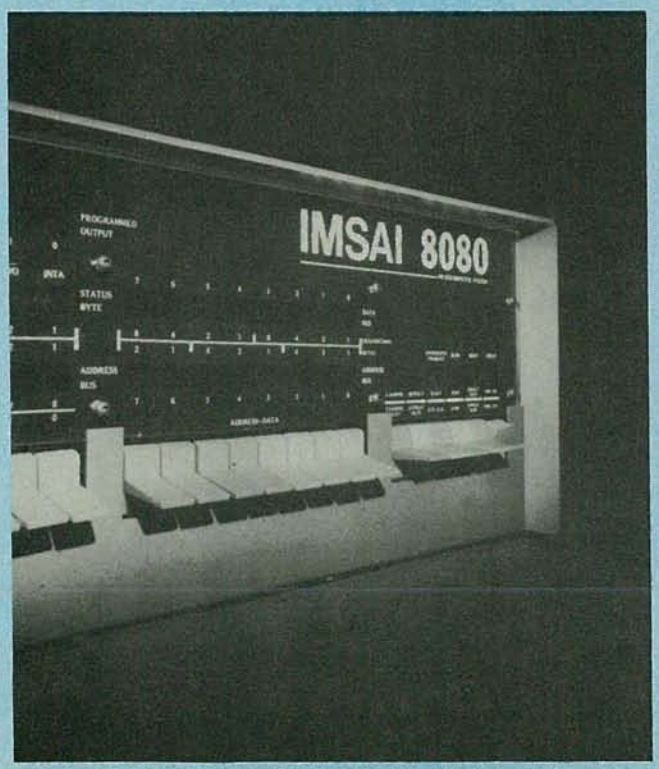


FIG. 1—AN EARLY PERSONAL COMPUTER, all data had to be entered manually, and in binary form, on this *IMSAI* computer.

BASIC's got larger and more sophisticated, but there were still problems. Each version of the language was tailored either to a particular computer and operating system or to a particular microprocessor—BASIC's were written in *machine code* (binary code using the *instruction set* specific to each microprocessor—the 8080, 6800, or 6502). There was no compatibility; BASIC for one computer would run only on that computer and, even then, all of the elements of the rest of the system had to be just right; a BASIC recorded on cassette using the Kansas City Standard could not be loaded into a system that used the Tarbell Standard. BASIC was evolving, but things were still a mess.

Even worse was the fact that what software (programs) existed had been created using a specific version of BASIC. A program, good or bad, could almost always be run on only one type of computer.

Getting it together

Finally the day arrived when the all-in-one computer made its debut. Among the first of those computers were the *PET*, the early OSI computers, the *Apple II*, and the original Radio Shack *TRS-80*.

Typically, those computers included a keyboard, cassette interface, a built-in video display (in the case of the *PET*) or output for a video monitor (which could also be used to modulate an RF carrier and be fed to an ordinary TV receiver) and—most significantly—BASIC in ROM.

That meant that you could go out and buy a computer, connect it to a monitor or TV set, turn it on, and be ready to program in BASIC right away. What a revolution!

Interestingly enough, many of the BASIC's in ROM came from the same source—Microsoft, the company that had created the original *Altair* BASIC. Sometimes the BASIC was clearly identified as coming from Microsoft and sometimes, for one reason or another, its origin was obscured.

Of course, each version of the language would run only on the computer it was designed for, due to specifications set forth by the computer manufacturer, the microprocessor (CPU) used, and the graphic and color capabilities of the computer, if any. A program written in BASIC for one computer would not run on another, even though the language in which it was written had originated from the same company.

To help cope with the proliferation of BASIC's, a book called *The BASIC Handbook* was put together by David A. Lien. It was (and still is) a cross-reference of BASIC statements and commands. The book allows you to "translate" a program written in one version of BASIC into a different version and proved especially handy when a program listing was published that you wanted to use but could not, directly, because it was written for a computer other than your own.

Despite the many different variations of BASIC abounding, the fact that Microsoft was responsible for many of them was an important step toward standardization.

Floppy disks

What probably made "universal" software possible was the floppy-disk drive. Originally developed by IBM in the 1960's, floppy-disk drives, while considerably more expensive than cassette recorders, quickly found favor among serious small-computer users.

They offered speed (a program that might take several minutes to load from cassette could be loaded in a couple of seconds from disk) and convenience. Instead of trying to locate a program manually on a cassette, or having to maintain a vast library of cassettes, each holding one program, the floppy disk, together with its DOS (*D*isk *O*perating *S*ystem) made it possible for the user to simply request a particular program and, within seconds, the DOS would locate it on the disk—wherever it might be stored—and load it into the computer. Programming languages, too, could be stored on

disks.

Disks have other advantages, too. Almost any business program requires that files consisting of names, addresses, inventory items, accounting figures, etc., be maintained and updated. Trying to perform file maintenance using cassettes was extremely time-consuming—and risky. Each time a file was changed—even by only one item—it had to be completely rewritten on cassette. That was not only time-consuming, but also potentially dangerous, since if procedures were not followed carefully, the old file—and sometimes the new one—could be lost forever. File handling on disk was effortless; simply load in the old file, modify it, and save it on the disk under a slightly different name—perhaps keyed to the date. Both files would be readily available should either of them be needed.

Disks could hold a large amount of information—be it program, file, or language—that could be called up nearly instantaneously.

Enter CP/M

As with everything else in the early small-computer world, each disk operating-system was incompatible with any of the others. Then, in 1973, came a DOS that changed things completely.

That operating system was CP/M (usually taken to stand for "*C*ontrol *P*rogram for *M*icrocomputers"), developed by Gary Kildall, who formed a company called Digital Research. CP/M was a disk-operating system that could be used on *any* 8080 or Z80-based computer. It was designed so that it could be modified (by anyone who could interpret the six manuals that came with it) to use any computer's I/O (*I*nput/*O*utput) scheme. It was the *universal* DOS! (It has proven so popular that even computers that do not use CPU's belonging to the 8080/Z80 family—such as the *Apple II*, which uses a 6502—can now use it after certain hardware modifications have been made.)

With CP/M, and a "standard" BASIC, a program could be written on one type of computer, and run with little or no modification on any number of others. That was a giant step toward making good business software available to a large number of users at a reasonable price.

A separate article in this section discusses CP/M, and a description of its characteristics will be found there.

A standard BASIC

While Microsoft was developing BASIC's to be implanted in ROM for use in "plug-in-and-run" personal computers, it, and other companies, were developing versions of BASIC that were specifically designed to operate with CP/M. One of those companies was Software Systems, which created CBASIC (later revised to CBASIC 2, now itself known as "CBASIC").

CBASIC, which can be run on any CP/M system, brought to the world of small computers many of the features of languages that ran on the larger computers many businesses wanted to get away from. In fact, it was designed specifically with business applications in mind.

Programmers who used to scream for microcomputer version of FORTRAN and COBOL—high-level languages with which they were familiar—soon discovered that CBASIC and its relatives could do the job for them. While microcomputer versions of FORTRAN and COBOL are available, they never attained the popularity that was expected of them.

One major advantage tht CBASIC has over the BASIC's that come with most personal computers is the fact that it is *compiled* rather than *interpreted*, and consequently runs much more quickly.

By analogy, running an interpreted program is similar to your having to relearn the rules of multiplication from scratch *each* time you have to multiply two numbers. A compiled program, on the other hand, is like doing multiplication the way you normally do after having learned the rules—you just do it without thinking about it. As you can imagine, that speeds things up considerably.

A CBASIC program is written in two steps. The first is the creation of the *source* program, which looks more or less like any other BASIC program—with the exception that line numbers do not have to be used unless they are necessary to identify certain sections of the program.

The source program is then compiled—it's "boiled down" to a series of machine-language instructions which become known as the *intermediate* file. To protect their investment of time and knowledge, programmers usually sell their programs in intermediate form; while it's easy to understand (and modify) a source program, an intermediate file can only be run...the computer can understand it, but the curious human cannot.

The intermediate file is run with the aid of what's known as a *run-time program*. Without that program it's useless.

Programming in a compiled language is more difficult than working with an interpreted one. If the program is compiled and there is an error, correction is not simply a matter of editing or rewriting the incorrect line(s). Instead, the original source program must be edited, and then recompiled. Compiled-language programming and debugging can be more time-consuming than writing an interpreted-BASIC program; but the speed and efficiency resulting from a compiled language, such as CBASIC, more than makes up for that. And, you, as the end user of a software package, benefit without having to program at all.

Recently, a language called CB-80 was released. It is essentially compatible with CBASIC, but offers certain advantages to both the programmer and the end-user. (Note: Both CBASIC and CB-80 are now supplied by Digital Research.)

Among its features is a *linking* capability that allows programs to be written as a series of modules. Those modules can be easily combined to form complex programs and, in addition, allow infrequently used modules to remain on disk until called for, which means that more computer memory-space is available for data.

It also can handle (like CBASIC) extremely large numbers—up to 14 significant digits—which makes it valuable for accounting purposes.

A final look

A final (as of the time of writing) development is the arrival of *program generators*. They can generate BASIC programs without your having to know anything about formal programming. One such program is "The Last One," from D.J. 'AI' Systems, Ltd.

The program takes the shape of a flowchart: "First do this, then do that, but if *this* results in something special, then *don't* do that, but do something else."

It allows you to create your own programs in the BASIC you normally use on your system simply by responding to a series of questions presented on your video terminal. A "menu" of choices; or opportunities for you to indicate your own categories, title headings, and other items or operations to be performed, lets you create your own program in BASIC without even knowing how to program.

How that affects you

After you bought your computer, and decided to use it for business or serious at-home use, you almost certainly ran into the problem of finding software that would do what you wanted. In a few cases, if you were lucky, you found a program written specifically for your machine that served your purposes; in most cases you probably did not.

However, with CP/M, and with BASIC's far more highly developed than the original from Dartmouth (and the truncated versions of that one for the first personal computers), you now can choose from a wide variety of programs that will run under a language you can load into your computer and do precisely the job you want. **R-E**

Word Processing

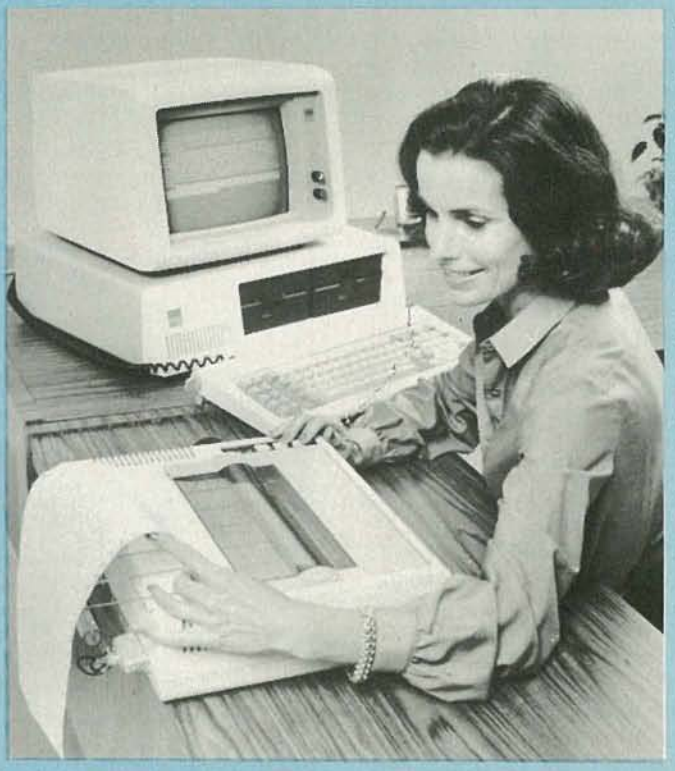
Writing, and editing, can be a lot easier with one of these word-processing programs. Some will even check your spelling!

HERB FRIEDMAN

A COMPUTER DOES NOT REALLY know, nor does it care, whether the signals racing through its circuits represent calculations, data, graphics, or words. A computer will do what it is instructed to do by its programming. If the software tells it to allow the creation of character strings—that is, words—and how to manipulate them for editing, revision, organization, and, finally, printing, it will do just that.

As a general rule of thumb, if the computer is programmed specifically to manipulate words in any manner we say the computer is functioning as a "word processor."

In actual fact, while the term "word processing" was originally understood to mean the preparation of documents, for the modern personal computer it has come to hve a more flexible meaning, and the word-processing software available for personal computers encompasses a wide range of applications. About the only limitation to the features available in word-processing software is price—the greater the number of features, the more you will have to pay for your software.



Line-oriented editors

The simplest—and least expensive—form of word processor is the so-called "line-oriented editor," some version of which is often supplied as part of the computer's operating software, or in an editor/assembler package. In a line-oriented editor, each line of text, data, etc. in the file is displayed along with a line number. Modification of the "test" is done by referring to the line number. To rearrange the text line-by-line the user actually operates on the line numbers. To add or delete characters, strings, etc., instructions are entered referenced to specific line numbers.

It's a cumbersome method that works with modest efficiency when editing assembly-language programs, and can be used—when nothing else is available—to edit text. Given enough time, one can create, edit, and revise text with a line editor. If the final, edited, version displayed on a CRT terminal is fed to a printer, the output resembles the CRT display—with or without the line reference-numbers, as determined by the user. If a printing terminal is used for editing, the printout will resemble the original input, less the line numbers.

Full-screen editors

Editing files using a CRT terminal is handled more conveniently with something called a "full-screen editor"—software that allows the user to work directly on any portion of the screen display. With a screen editor it's no longer necessary to refer to a specific line number. The full-screen editor permits the user to make modifications to the text anywhere on the screen, starting at the position of the cursor (the square or underline on the display that indicates that "you are here"). The cursor can be positioned anywhere on the screen and the user can insert single characters or strings—words, sentences, phrases, and paragraphs—at that point. He can change data at the cursor position or delete forward or backward from the cursor-position by word, by line, to the end of the text, and so on.

In the more advanced full-screen editors, the user has total control over line and page movements, and in the "full-blown" screen editors such as *VEDIT*, the user can customize the keyboard for specific single-key commands. For example, a Heathkit/Zenith *H-19* terminal can be customized so that the blue key is BACK TAB, while the "F_1" and "F_2" keys can provide different indentations. Advanced screen-editors are almost a necessity when editing large data-files or formatted data that extends for hundreds of lines, or for the structured programs of Pascal and FORTRAN. They can be edited without a full-screen editor, but the process can seem to take forever.

Both line and full-screen editors can be memory- or disk-oriented. A memory-oriented editor retains everything in memory until the data is dumped deliberately to the storage medium. When the memory is full the user *must* do that, or lose whatever is input after the memory is full. (The editor will generally warn him when that situation is about to arise.)

Disk-oriented editors read and write a specific number of lines of screen display, or the contents of a specific amount of memory, to disk at a time. A disk-oriented system works a little slower when *macros* (large quantities of data considered as a whole, but made up of separate entities such as individual lines) are being written continuously, and it can be slow when you want to scroll back through previously written macros, but it does permit continuous creation or editing of

documents larger than the computer's memory can hold. The producer of an editor generally specifies the minimum amount of RAM required to use his software and approximately how much data or text can be stored. He may also indicate how much memory will be used before a macro is written to the disk.

Output formatting

A significant difference between software described as "editors" and "word processors" is *output formatting*: how the text, data, etc., appears in its final, printed, form. For basic line-oriented editors, the printout is exactly the same as for an ordinary file. If you want a space between lines you must imbed a carriage return (CR) or linefeed (LF), or both, within the file. If you want indentation you must imbed indents within the lines. (But as stated earlier, that's slow.)

Full-screen editors often permit formatting of the screen display, thereby allowing the software to step the cursor to the proper position for data entry. An ordinary screen-editor, however, has no provision for formatting the output to the printer. Depending on the editor, the output of the printer might be a straight line-by-line printout, or, possibly, continuous copy with the end of each line wrapping around to the next line. The printed output of a screen editor usually bears no resemblance to what was created on the screen of the CRT.

To permit formatting—that is, to specify the line and paragraph spacing, paragraph indenting, margin size, etc.—of the printout of some of the less-expensive text and screen editors, a special utility program called a "print formatter," or simply a "formatter," is usually available.

If you are budget-minded, you need not spend funds for printer software if it isn't required. The lack of print formatting should never be assumed to imply inferior software quality. Print formatting, which often includes a selection of drivers for a variety of printers, is expensive—often representing several hundred dollars of the price of good word-processing software—and if you don't need formatting there's no reason why you should spend money for it. For example, *Wordmaster*, a video text-editor, sells for about $99.00. It has many of the capabilities of *WordStar*, which many consider to be the most powerful of the word processors; but that program carries a price tag of $395.00. If you don't require hard copy (a printout), but need screen formatting and other word-processing features, *Wordmaster* may prove more than adequate, and save you a considerable amount of money.

Print formatting

The appearance of the final product is what word processing is really all about. Precisely *what* a word processor is supposed to do depends on who is describing it. The commercial, and generally expensive, word processors of the 1970's from companies such as IBM and Xerox were basically intended for the preparation of letters and similar documents by skilled typists. Many "processors" in the $8000 to $13,000 price range lacked a CRT; they weren't much more than electronic typewriters with memory, and the typist composed directly on a sheet of paper. (One could say he or she used a "printing terminal.") Now, just a couple of years into the 1980's, we find that the modern personal-computer can far outperform the commercial word-processing machines of the mid 1970's.

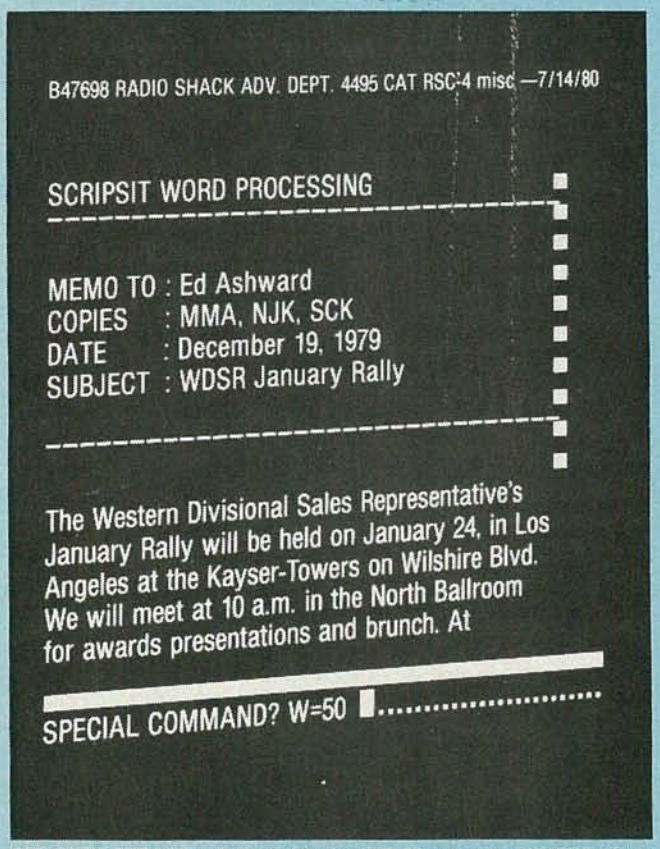
Word processors for personal computers

Personal-computer word processing got its start in 1977, which Michael Shrayer introduced a program called *The Electric Pencil*. Eventually there were over 40 different versions of the program, configured for different video-display boards, cassette interfaces, disk operating-systems, printers, etc.

Several versions were produced for Radio-Shack's *TRS-80*, and proved immensely popular. The original *TRS-80* could output both upper- and lower-case characters to a printer, but its video display had only upper-case capability. However, a simple modification that cost less than $10, and which was described in *The Electric Pencil's* documentation, could force the CRT to display both upper and lower case, and provide a keyboard CONTROL key (required by the program). Shrayer's program provided the software to display upper and lower case with the proper shift sequence, and drivers for both parallel and serial printers. The program even provided for underscore (on printers that could backspace, such as some *Selectrics*).

The cassette-based software provided just about all the editing features common to commercial word-processors of its day: insertion and deletion of characters, words or lines, block-move, string search, search-and-replace, extensive printer formatting—including right-hand justification—and tape storage. A disk version provided disk files and a disk directory. *The Electric Pencil* was a software "heavyweight" by the word-processing standards of its time. A program similar to *The Electric Pencil, EasyWriter*, was created for the *Apple II* computer, and has also been adapted to run on IBM's *Personal Computer*.

Though the *TRS-80* wasn't really intended for word processing—among other things it lacks an underscore—Radio Shack wasn't going to let almost a quarter-million *TRS-80* owners go somewhere else for software, so it introduced *SCRIPSIT*, a $29.95 cassette-based word processor that many consider one of the easiest, most convenient systems to use.



DESIGNED FOR USE with Radio Shack's *TRS-80* computer, *SCRIPSIT* offers the features of a word processor at a budget price.

SCRIPSIT differs from most other word-processing software in that it is organized for direct composition by an author rather than for the preparation of documents. All functions—create, edit, revise, insert, paragraph/phrase move, word and block interchange, and others—are performed in one operating mode. When a document is saved on disk or cassette the new or revised work—the last effort—is

WORD-PROCESSING PROGRAM SUPPLIERS

Word-processing programs are available from both equipment manufacturers and independent software suppliers. To help you find the proper program for you, a few of those independent suppliers are listed below.

ASTRO-STAR ENTERPRISES
5905 Stone Hill Drive
Rocklin, CA 95677

BILL'S MICRO SERVICES
210 S. Kenilworth
Oak Park, IL 60302

CHARLES MANN AND ASSOCIATES
7594 San Remo Trail
Yucca Valley, CA 92284

COGNITIVE PRODUCTS
PO Box 2592
Chapel Hill, NC 27514

COMPCO
8705 North Port Washington Road
Milwaukee, WI 53217

COMPLETE BUSINESS SERVICES
90 West Center Street
Logan, UT 84321

COMPUSOCO
26251 Via Roble
PO Box 2325
Mission Viejo, CA 92690

COMPUTER BUGS
PO Box 789
Boynton Beach, FL 33435

COMPUTERWARE
1512 Encinatas Blvd.
Box 668
Encinatas, CA 92024

COMPUVIEW PRODUCTS, INC.
618 Louise
Ann Arbor, MI 48103

CORNUCOPIA SOFTWARE
PO Box 5028
Walnut Creek, CA 94596

CREATIVE SOFTWARE
PO Box 4030
Mountain View, CA 94040

DATA STRATEGIES, INC.
PO Box 178446
San Diego, CA 92117

DUANE BRISTOW
Route 3
Albany, KY 42602

FMG CORPORATION
5280 Trail Lake Drive
Ft. Worth, TX 16020

HAYDEN BOOK COMPANY
50 Essex Street
Rochelle Park, NJ 07622

HEXAGON SYSTEMS
PO Box 397, Station A
Vancouver, B.C., Canada V6C 2N2

INNOVATIVE SOFTWARE APPLICATIONS
260 Sheridan Avenue, Suite 300
Palo Alto, CA 94306

INTERACTIVE MICROWARE, INC.
PO Box 771
State College, PA 16801

MARK OF THE UNICORN
PO Box 423
Arlington, MA 02174

MICRO PRO
1299 4th Street, Suite 400
San Rafael, CA 94901

MIDWEST SCIENTIFIC INSTRUMENTS, INC.
220 W. Cedar
Olathe, KS 66061

MONUMENT COMPUTER SERVICE
PO Box 603
Village Data Center
Joshua Tree, CA 92252

THE MUSE COMPANY
330 North Charles Street
Baltimore, MD 21201

OMIKRON
1127 Hearst Street
Berkley, CA 94702

PROFESSIONAL SOFTWARE, INC.
1666 Crescent Road
Needham, MA 0194

SMALL BUSINESS APPLICATIONS
3220 Louisiana Street
Houston, TX 77006

SMALL SYSTEMS SOFTWARE
PO Box 366
Newbury Park, CA 91320

SOFTWARE WORKS
8369 Vickers
San Diego, CA 92111

THORPE DATA SYSTEMS INC.
22968 Victory Blvd.
Woodland Hills, CA 91367

automatically saved under its original title. If, using the disk version, the revised work is saved in place of the original, a series of files representing earlier versions can be avoided, and disk storage-space conserved.

Commands for virtually any type of print formatting—on a page basis—can be imbedded in the text. (It is difficult to format on a line-by-line basis, as might be required for business applications.) Also, tabulation, which might be required for business documents is somewhat difficult, as there are few "business document' format commands. (Note: The version of *SCRIPSIT* for the *TRS-80 Model II* is a "standard" document-type system suitable for business use.)

A word-processor modifier, Acorn Software Products' *SUPERCRIPT* restructures *SCRIPSIT* to add many features that were originally left out. Those features included underscoring, kerning (i.e., slashed zeroes), subscripts and superscripts on printers capable of printing them, and, perhaps most important, drivers for virtually every possible printer, including both types of *Selectric* mechanisms available for personal computers.

A few other early attempts at word processing for personal computers were not as successful, or as easy to use. For example, one word-processing system could feed upper- and lower-case characters to the printer, but the CRT display was upper-case only; a "special mark" was used to indicate which character displayed on the CRT screen in upper case would actually print as upper case.

Then again, there was, and still is, the problem of screen size. Some CRT displays are only 40 or so characters wide. Few document lines are that short, so some form of "window" device has to be provided so the user can move off-screen to the right. It's a sort of a horizontal version of the vertical scroll.

The popularity of *The Electric Pencil* and *SCRIPSIT* was just the starting point for small-computer word processors, and over the next few years, word-processing software was introduced that rivals or excels what's available for mini and large computers. A number of computer manufacturers introduced their own word processors, such as *WordPro* from Commodore International, for use on their *PET* and *CBM* machines, and *AutoScribe II* for the Heath/Zenith *H89*.

Many "full-blown," multi-feature, word processors are available for computers that use the CP/M disk operating-system, and are also specifically structured for other computers. By "full-blown" we mean word-processing software that goes beyond the mere preparation of letter-type documents and provides a means to do more than revise thoughts, correct errors, or format for justified printing with its flush left and right margins, or centered headings.

For example, *Magic Wand,* which sells for a bit under $300, provides "spooling," which makes it possible to print one document while working on another. It also features

EQUIPMENT AND TRAINING NO OTHER SCHOOL CAN MATCH.

NTS HOME TRAINING INVITES YOU TO EXPLORE MICROCOMPUTERS, DIGITAL SYSTEMS AND MORE, WITH STATE-OF-THE-ART EQUIPMENT YOU ASSEMBLE AND KEEP.

Without question, microcomputers are the state of the art in electronics. And NTS is the only home study school that enables you to train for this booming field by working with your own production-model microcomputer.

We'll explain the principles of trouble-shooting and testing your microcomputer and, best of all, we'll show you how to program it to do what you want.

You'll use a digital multimeter, a digital logic probe and other sophisticated testing gear to learn how to localize problems and solve them.

Send for the full color catalog in the electronics area of your choice – discover *all* the advantages of home study with NTS!

NTS also offers courses in Auto Mechanics, Air Conditioning and Home Appliances. Check card for more information.

1.

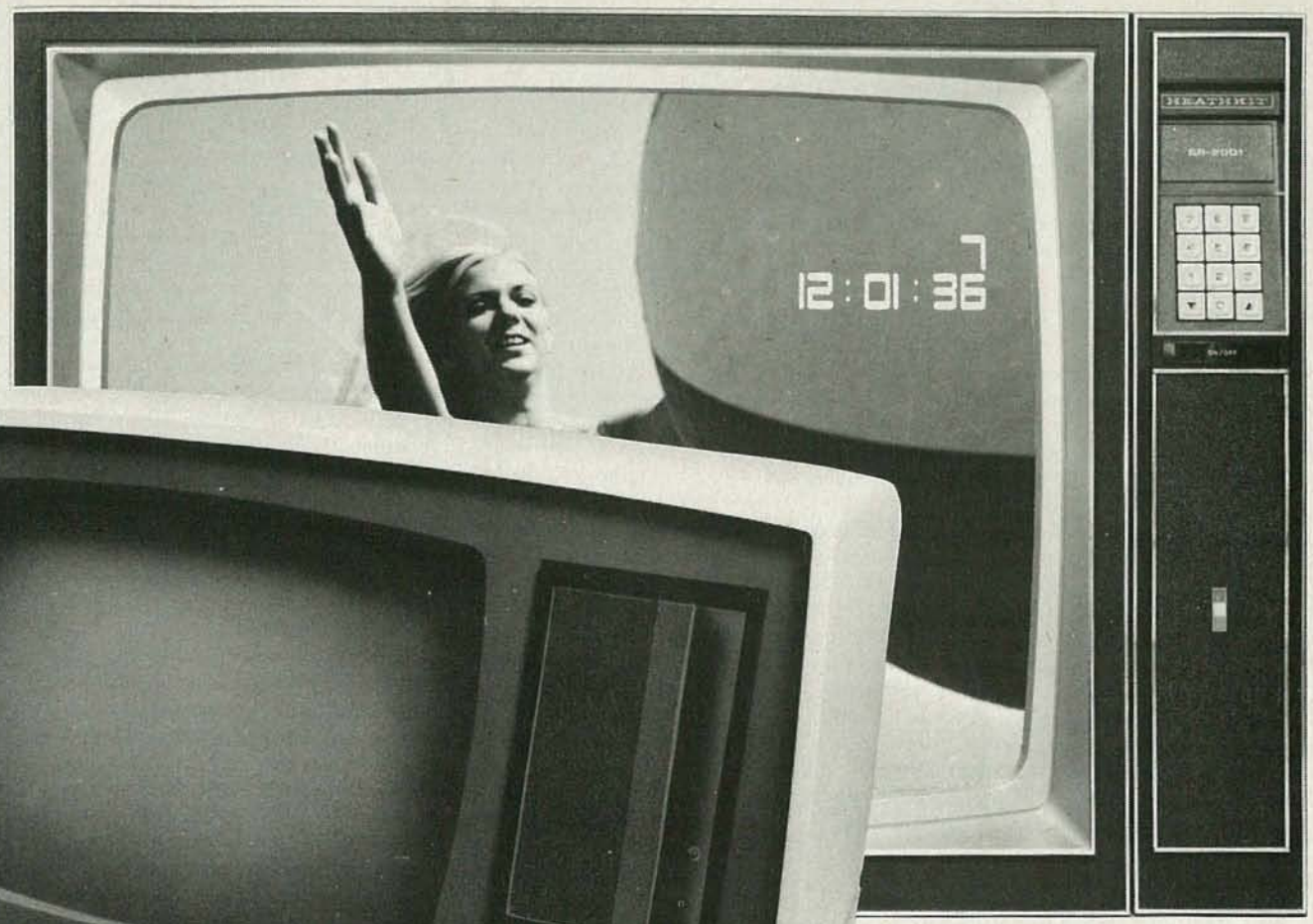
2.

We believe that training on production-model equipment, rather than home-made learning devices, makes home study more exciting and relevant. That's why you'll find such gear in most of NTS's electronics programs.

For instance, to learn Color TV Servicing you'll build and keep the 25-inch (diagonal) NTS/HEATH digital color TV.

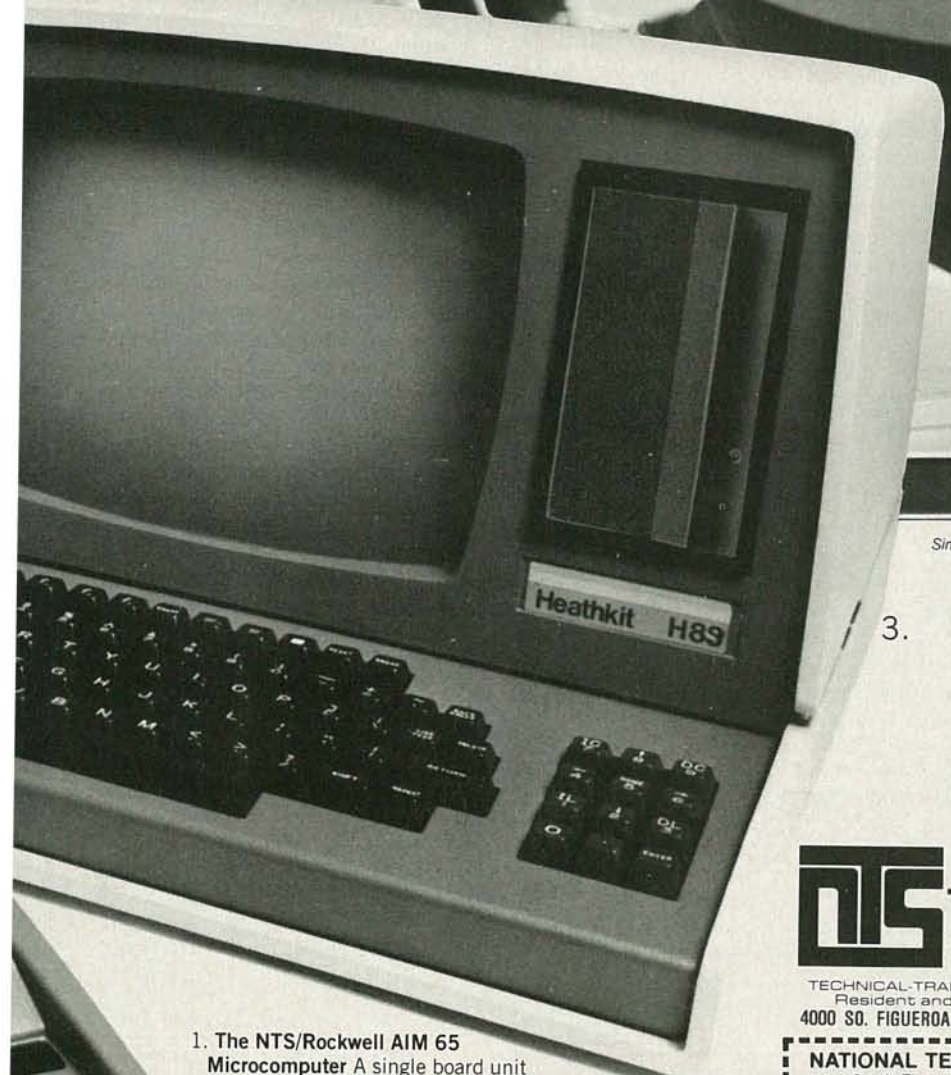
In Communications Electronics you'll be able to assemble and keep your own NTS/HEATH 2-meter FM transceiver, plus test equipment.

But no matter which program you choose, NTS's Project Method of instruction helps you quickly to acquire practical know-how.



Simulated TV Reception

4.



3.

1. **The NTS/Rockwell AIM 65 Microcomputer** A single board unit with on-board 20 column alphanumeric printer and 20 character display. A 6502-based unit 4K RAM, expandable. 2. **The NTS/KIM-1 Microcomputer** A single board unit with 6 digit LED display and on-board 24 key hexadecimal calculator-type keyboard. A 6502 based microcomputer with 1K RAM, expandable. 3. **The NTS/HEATH H-89 Microcomputer** features floppy disk storage, "smart" video terminal, two Z80 microprocessors, 16K RAM memory, expandable to 48K. 4. **The NTS/HEATH GR-2001 Digital Color TV (25" diagonal)** features specialized AGC-SYNC muting, filtered color and new solid-state high voltage tripler rectifier.

NATIONAL TECHNICAL SCHOOLS

TECHNICAL-TRADE TRAINING SINCE 1905
Resident and Home-Study Schools
4000 SO. FIGUEROA ST., LOS ANGELES, CA. 90037

NATIONAL TECHNICAL SCHOOLS
4000 South Figueroa Street,
Los Angeles, California 90037 Dept. 206-042

Please rush FREE color catalog on course checked below

- ☐ **MicroComputers/MicroProcessors**
- ☐ **Communications Electronics**
- ☐ **Digital Electronics**
- ☐ **Industrial Technology**
- ☐ **Auto Mechanics**
- ☐ **Air Conditioning**
- ☐ **Home Appliances**
- ☐ **Color TV Servicing**

Name _____ Age _____

Address _____

Apt. _____ City _____

State _____ Zip _____

☐ Check if interested in G.I. information.
☐ Check if interested ONLY in classroom training in Los Angeles.

automatic hyphenation, will automatically stop printing at preselected points so the user can insert text (a name, for example) directly from the keyboard, can integrate data files into a document in the correct order, will search and replace either strings or data, and can be print-formatted on a line-by-line basis.

A useful "extra" is a user-written "HELP" system, whereby a table of salutations, or introductions for specific titles or positions, etc., is available through a HELP CRT screen-menu selection.

For a few dollars more, there is the previously mentioned *WordStar*, which provides additional business-oriented features that include decimal tabulation, whereby the decimal points in a column of numbers will be aligned automatically.

Most full-blown word processors also permit formatting of the CRT screen so that what you see will exactly match the printed output. (With less-sophisticated software, the screen often displays the text in line-by-line order along with the print-format instructions. Exactly how the final copy will look isn't known until its printed.)

One of the problems with full-blown word processors is they have so many features, requiring so many control keys, that it's often necessary to make frequent reference to the instruction manual, or to an instruction-summary card, to take advantage of those features that are used infrequently.

Some systems provide the user with small tabs labelled with control functions to be attached to the terminal's keycaps. (Radio Shack initially provided a set of 15 with their budget-priced *SCRIPSIT*.) Omega's full-blown *VTS/80* word-processing system is about the ultimate in convenience, because it provides a *complete* set of replacement keycaps—one for almost every character key on the terminal!

Dictionaries

One of the most useful add-on programs for a word processor is a piece of software called an "electronic dictionary," also known as an "electronic spelling-checker." There are presently several electronic dictionaries available for personal computers, among them: *Hexspell*, *Spellguard*, and *Microproof*.

An electronic dictionary contains anywhere from about 20,000 to over 50,000 words, the entire collection being known as the "dictionary" (or "library"). It runs as an executive (control program) for your document and checks each work in the document against the words stored in it. Each time it locates a word for which it can find no match, it displays the word on the CRT screen—alone or in context with a phrase of the text, depending on the program.

The user decides whether the "unknown" word is spelled correctly or incorrectly. If the spelling is correct, he can tell the dictionary to "SKIP" (proceed with the check), or "ADD" the word to the dictionary list. If the word is spelled incorrectly, the user can enter the correct spelling and request a "REPLACEMENT" in the original document. If the replacement, or correct spelling, is still unknown to the dictionary, the user has the option of adding that word to its list. In that way the user builds his dictionary to correspond to his particular vocabulary.

For example, the version of *Hexspell* that was used to proofread this article contains a specific computer and communications vocabulary that was built up by the author each time an "unknown" word was entered, using a LEARN command. Another author-constructed version of *Hexspell* has a vocabulary tailored for electronic construction-projects. Once you have the basic dictionary you can build any number of special versions.

There are two ways in which an electronic dictionary can process a document, usually determined by the number of disk drives available. For software that requires only one disk drive, the user's document is checked rapidly against the dictionary and a listing of the "unknown" words is provided, or a character in each "unknown" word is changed to some symbol such as the "#" or the "@". Regardless of how the "unknown" word is indicated, the user must go back and correct the original document through the word-processing software.

An electronic dictionary for use with two disk drives runs the dictionary in the first drive and a copy of the original document in the second drive. (To avoid accidental loss of a document, the disk on the second drive must have enough free space for an "edited version" of the document.) The dictionary software causes the document to scroll at a fast reading pace on the screen; as it scrolls, an edited version of the document is recorded on the second drive. When the software locates an "unknown" word it will stop and allow the user to make a correction by substituting the correctly spelled word in the edited version, and/or add the word to the dictionary, or skip ahead. At the completion of the process, the edited version of the document contains all the corrections, and is automatically recorded in place of the original one, providing two fully corrected document files on the disk (one serves as a backup).

The two-drive dictionary is much slower than the single-drive one, because of the continuous writing to the edited-document file. However, the user ends up with a fully corrected document. With single drive he gets a list of "unknown" words faster—sometimes in less than a minute—but must then go back and correct the original document, so the total time involved is about the same. As a general rule more convenient.

Among the most popular uses for a personal computer is a word processor. Once you've had a taste of how easy it is to work with a word processor, you will find it's extremely difficult to go back to your typewriter. **R-E**

Data Base Management

Store and retrieve data efficiently with a database management-system.

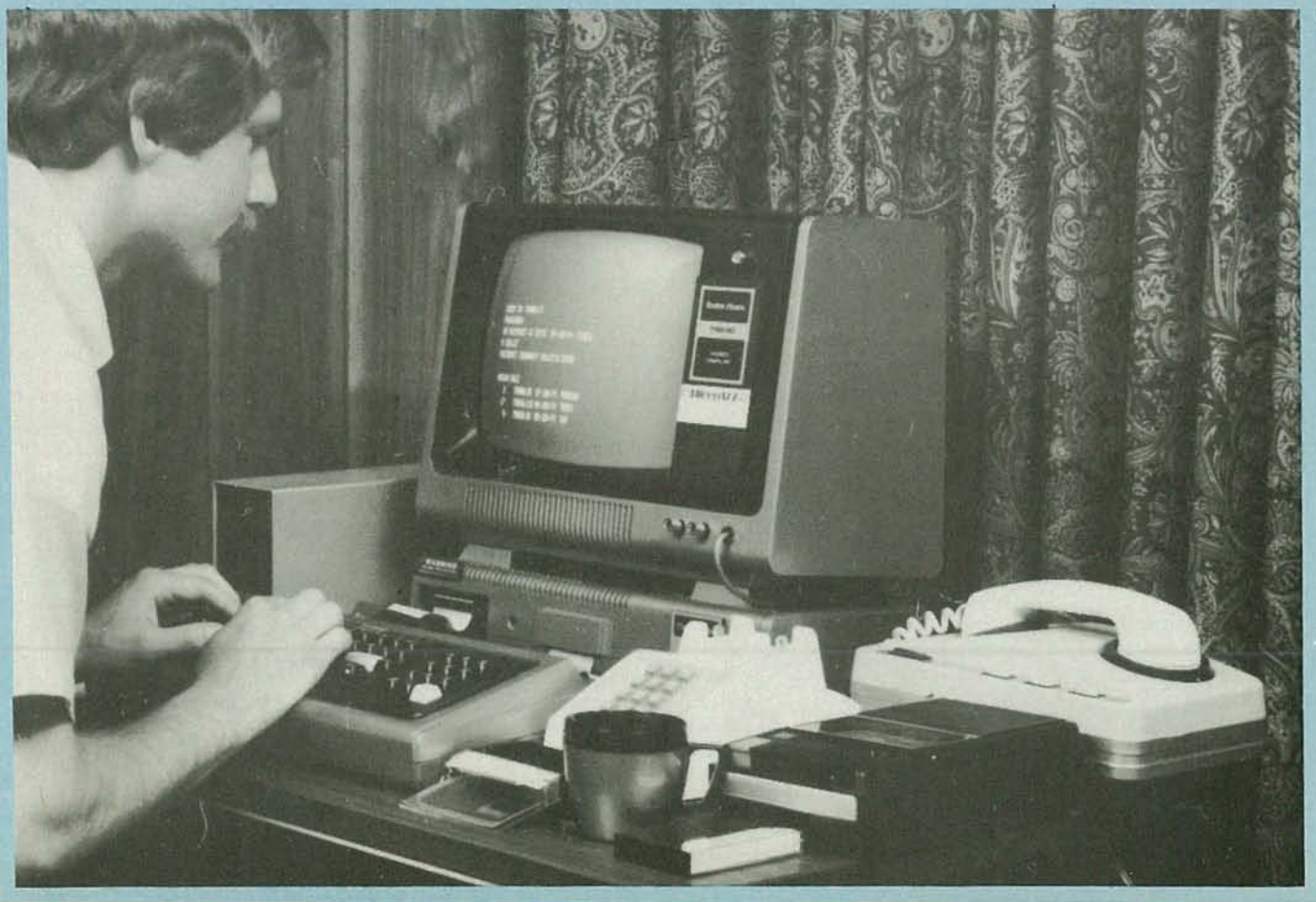
HERB FRIEDMAN

A DATABASE IS AN ELECTRONIC FILING SYSTEM. TO BE MORE precise, it is an electronic representation of a filing system such as you might find in any home or business. A database can represent nothing more than the electronic equivalent of ordinary 3 × 5 index cards, or it might contain every bit of information concerning a business or organization, and have the power to calculate "unknowns" from the data stored in the files. A database can handle all details of a business inventory, such as the accounting, the billing; it can even prepare reports, summaries, estimates, and the like, from the information in the files. It can store newspaper articles, magazines and books, games, even digitized pictures (photographs). In actual fact, a database is whatever the last programmer to write one says it is.

No matter how little or how much information the database might contain, with very few exceptions the inherent "filing" or "storage" system starts out with a base system that approximates the ordinary file cabinet.

Imagine, if you will, the common office 4-drawer file cabinet. The cabinet itself is analogous to the personal computer. What's inside the cabinet is the database. Now there might be nothing more inside the cabinet than a pack of 3 × 5 file cards, each card representing the title of an LP album you have recorded on cassette tape for your permanent music library. The primary entry on the card is the album's title; if there is more than one work on the record, you will have a list of the individual songs, arias, orchestral selections, etc., the catalog number you assigned, and possibly the name of the composer of the album or of each selection. To see what performances were on a particular tape you would flip through the cards until you came to the desired title, and then you'd read any individual entries.

If you did the same thing with a computer you would be creating what we call a database. You could very easily write a BASIC program that would permit you to store each album title, a list of the selections on each album, and its catalog number. Each entry, consisting of album title, selections, and catalog number, is called a record in our database. Each



record serves as an electronic file card that functions exactly the same as the 3 × 5 card file did. You would enter the album's title into the computer and the screen would display a list of the selection and the catalog number.

While it's clever and entertaining to have a computerized tape library, the question is *why*? If the database serves only as a substitute for a file card why use a computer? It's actually easier and faster to use a stack of 3 × 5 file cards, and they are certainly a lot less expensive than a computer. The truth is, if you can do something expeditiously with file cards, then there's no need to computerize it.

But let's expand the way we catalog the music library. With both the hardcopy and electronic file records organized by album title, it's inconvenient to locate a specific album if you know only the name of a selection, so the 3 × 5 card file requires some form of cross-indexing; most likely a separate 3 × 5 card for every selection, showing the album title along with the catalog number you assigned to the tape. That results in a separate 3 × 5 card for each and every selection, as well as for each album title. If you included the composer's name in your card-file system you'd be adding many additional file cards because each composer would require an individual file card listing only his or her works. Cross-indexing produces a lot of paper, and a lot of search-and-find by the user, as is well illustrated by the catalog system of any book library.

It is when we step into cross-indexing that the justification for an electronic database can be understood. Just to maintain a list of the individual albums or tapes in your collection we have created electronic records which are the equivalent of individual file cards. Now the very same program can, in the blink of an eye, search the individual data records for a specific song, catalog number, or whatever. No extra files are needed; the original computer data record for each album contains all the needed data. Rather than expand the total number of individual data records, we simply tell the computer what additional searches we would like to have. For example, instead of having the computer search out one selection, we might have the program search for all the albums that contained one or more items by a specific composer, or albums with songs having common words in their titles.

While all that might sound complicated, it's really quite simple to do electronically. Each entry in our database record is called a field. The album title is a field, each separate title is a field, and our catalog number is a field. The computer simply searches all the fields for the specific data.

As far as the computer is concerned, it does not care whether the data record contains information on cassette tape, and whether it's music or anything else. Enter in the data in the right order and the same program becomes a database for whatever you want. Because that type of database is both useful and relatively easy to write in BASIC, it quickly became one of the most popular database for personal computers in the form of a checkfile—a database that keeps track of your checks for budget analysis or income tax purposes. The user simply enters a list of checks for each month and assigns a category; such as charity, interest, fuel, medical, or whatever. At tax time a touch of a button can retrieve a list of all checks for a specific category, generally in the order they were entered.

If the program has some form of SORT facility—a way to sort in alphabetical or numerical order—each list could be displayed in alphabetical order, or check-number sequence, or whatever order is required. In searching for a specific check, things like misspelled names aren't really a problem, because there's usually not that much data involved. Certainly for the average user all checks in a specific category could be scanned on a CRT screen without much extra effort.

But what happens when there is seemingly endless records of data, with much new data being added almost daily? As you probably have surmised, in our music library and personal checkfile database we have created an electronic database very similar to the card-file system in a book library, only a library has thousands and thousands of individual data entries on individual file cards in its card catalog. It's often difficult enough to locate a book when you're certain of the precise spelling of its title or author, but what do you do when you're not quite certain of the exact title, or the spelling of an author's name?

A simple database for a personal computer is usually unforgiving; it must have the precise string of characters including spaces, or it returns a FILE NOT FOUND display on the screen. If your database is small enough you can display, as we've stated, all the seemingly pertinent or common data, but one could literally drown under the "common" data of a book library. To see how we handle that situation—because it's a key to the enhanced database—let's go back to our sample tape-library database and make some modifications to the software that expands its search and find feature.

One of the things we can do is program the software to search by three or four characters. For example, assume our software searches in a "Song Title" mode for a four-character string anywhere in a field. If we entered "Hell" the computer would locate the records for the songs "Hellfires", "Hello Sunshine", "Hello Happiness", and myriads of other songs starting with "Hello," because it's not that rare a word in songs. So we modify the software one more time to recognize another short string in any other string that's separated from the first by at least one space. We'll have the software search for three characters in the second string. We instruct the computer to search for the key "HELL SUN". In moments the computer puts the two key words together, finds the correct file, and displays on the screen:

HELLO SUNSHINE TAPE # 135 SONGS OF THE 60's.

Alternately, our program in an "Album Title" mode would accept the key "SON OF " (note the space after "of" because we need three characters— and a space is a character), and display:

SONGS OF THE 60's TAPE # 135.

As you can see, we can search on virtually any type of key. The partial string system we've illustrated is very sophisticated and is used by libraries and complex databases, but it is coming into common personal computer use as the programmers get more adept. A more common personal-computer key system works on complete fields, with everything spelled exactly as saved by the database.

Storage schemes

For personal-computer databases there are two types of storage schemes: in-memory, and everything else, which usually means a high speed retrieval device such as disk. In-memory schemes are almost always used for cassette-storage systems, and a few somewhat unsophisticated disk-based databases. In the in-memory database, all the operating software (program) and all the data are transferred from the tape directly into the computer's RAM memory. All operations are performed directly on the data in memory—updating, modifications, etc.—and the new database is then stored back to the tape. If it is recorded on a blank section of tape the user has tape versions of the old and new data. If the data exceeds the memory capacity of the tape, it must be broken into separate recordings.

Since in the most elementary in-memory data systems, operations are performed directly in the RAM memory, any changes made are permanent in the sense they erase previously entered data. If you goof, the data is gone and must be reloaded from the tape. In the more advanced in-memory programs, a block of RAM memory is reserved as a

DATA BASE MANAGEMENT SYSTEMS

Database management systems are available from a number of independent software suppliers. Some of those suppliers are listed below.

APPLE ORCHARD
131 Highland Avenue
Vacaville, CA 95688

ASHTON-TATE
3600 Wilshire Boulevard, Suite 1510
Los Angeles, CA 90010

BLUE LAKES COMPUTING
3240 University Avenue
Madison, WI 53705

CONDOR
PO Box 8310
Ann Arbor, MI 48104

DANA INDUSTRIES
2612 Croddy Way
Santa Ana, CA 92704

HAYDEN BOOK COMPANY
50 Essex Street
Rochelle Park, NJ 07662

HIGH TECHNOLOGY
PO Box 14665
Oklahoma City, OK 73113

INNOVATIVE SOFTWARE
9805 Holly
Kansas City, MO 64114

JINI MICRO-SYSTEMS, INC.
PO Box 274
Kingsbridge Station
Riverdale, NY 10463

LJK ENTERPRISES INC.
PO Box 10827
St. Louis, MO 63129

MICRO ARCHITECT
96 Dothan Street
Arlington, MA 02174

MICRO DATA BASE SYSTEMS, INC.
Box 248
Lafayette, IN 47902

MICRO DECISIONWARE
4890 Riverbend Road
Boulder, CO 80301

MICRO LAB
811 Stonegate
Highland Park, IL 60035

MICROPRO INTERNATIONAL CORPORATION
1299 4th Street
San Rafael, CA 94901

THE MUSE COMPANY
330 N. Charles Street
Baltimore, MD 21201

PACIFIC SOFTWARE
2608 Eighth Street
Berkeley, CA 94701

PERSONAL SOFTWARE
1330 Bordeaux Drive
Sunnyvale, CA 94086

SOFTWARE PRODUCTS INTERNATIONAL
5482 Complex Street, Suite 115
San Diego, CA 92123

SYSTEMS PLUS
3975 East Bayshore
Palo Alto, CA 94303

THE BOTTOM SHELF
PO Box 49104
Altanta, GA 30359

UNITED SOFTWARE OF AMERICA
750 Third Avenue
New York, NY 10017

temporary buffer. Data is first stored in the buffer where they can be manipulated, modified, or even erased. When the user is satisfied with the data, a PUT command (or its equivalent) transfers the data from the buffer to the memory, where it adds to, or replaces the original data. A SAVE command transfers the data from the memory to the cassette tape.

Note that just to change one single datum entry the entire database, or at least one complete tape file, must be moved into RAM memory.

Depending on the particular software, a disk-based database can be in-memory or random access. Generally, simplified disk database software will be in-memory, and simply use the disk as a more convenient and faster storage medium. A more sophisticated disk database can open a specific disk file and search for the data, moving an image of only that data into memory. Alternately, it can permit updating or modification directly to the disk data. Also, a disk-based database can reference selective data entries from several disk files; the software can search on one or more keys across multiple files and then restructure the data for a presentation on the CRT or as hardcopy in a specific format. In essence, disk files expand the simple index-card database into a complex filing system, because each disk file can represent a file folder that can be accessed independently or as a group, so that the data in several folders can be studied, or integrated into a single document.

Using a database

Let's look at a filing system database as it might apply to records for your home. If you had the hardcopy records stored in a file cabinet they would be located under the letter "H" in a folder labelled "HOME", which is a good choice for the name of the computer file consisting of the home records. Within this folder would be other folders, perhaps labelled: Mortgage Payments, Taxes, Assessments, Fuel, Utilities, Repairs, Gardening Supplies. All of those individual folders would be converted to data records for the computer database. Depending on the particular software, the database might be entered exactly as you would manually search through the file cabinet. For example, the data might be stored on a disc with several other databases. To access the house records you would call up the file HOME, which might produce a menu on the screen listing all the available records (file folders). It might appear on your screen this way:

HOME—Make Selection
1-MORTGAGE PAYMENTS
2-TAXES
3-ASSESSMENTS
4-FUEL
5-UTILITIES
6-REPAIRS
7-GARDENING SUPPLIES.

Entering the appropriate number would call up the associated record. Let's assume we wanted to check the fuel bill for last December. We would enter a "4". Next, we enter DECEMBER, and the screen would display the entire fuel record for the previous December. Similarly, we might enter all the winter months to get a visual picture of fuel consumption for the entire winter.

For a picture of how property taxes were increasing we would enter a "2" and then key on, say, the past five years, 1977 through 1982, and the screen would display the entire tax record for that period. If we had to prove we paid the property tax, in order to resolve a dispute with the tax district, we could key on the specific quarter for a given year and the screen would display the necessary information, including the number of the check used to pay each particular tax bill.

If we had really extensive files—say we owned the property for more than 30 years—we might break up each record into mini-records. For example, entering a "2" would put us into

the TAX records file which would produce its own menu, which might read:

TAX RECORDS FOR:
A-1951 TO 1960
B-1961 TO 1970
C-1971 TO 1980
D-1981 TO PRESENT

Obviously, you can see that the concept of multiple, imbedded files can be applied to any complex file structure. The same idea applied to business records would give us a database that contained virtually all business records. If sufficiently large, if might easily contain individual records and systems for accounts payable and receivables, general ledger, journal, payrolls, including automatic calculations and tax schedules, inventory for all departments, job-cost analysis, and the utilities for maintenance and updating of the individual files.

As a general rule, when we get into the area of commercial-type files, the software usually contains word- or information-processing software so that printed reports, documents, and projections can be made from the data within the files. For example, with information-processing software, the same database will generate printed billing and monthly reports. It can print automatically sales projections based on last week, last month, last year, or last decade. It can print payroll checks, W2's, and charts of accounts. Exactly how the data information is used and presented in a printout is determined by the capabilities of the inherent word or information-processing software. In fact, major differences in business-database software is often the information-processing and information handling routines.

The common mailing list can be used as a simplified example of what is meant by information processing of a database. The usual mailing list will keep a list of clients, customers, or whatever. It provides for maintenance and updating of the list, and sorting on any key: names in alphabetical order, addresses by zip code or street number, even by the number of persons living at an address if that information is included in the data record. The normal hardcopy software for personal-computer mailing lists will provide a printout of the list on a sheet of paper or on standard pre-pasted mailing labels.

If the very same database also contains an information processor, it will merge in the names and addresses into prepared text or documents. That is how computers produce personalized form letters such as:

Dear Mr. Jones:

We're happy to inform you and Mrs. Jones of Anyplace, U.S.A., that you and your children Robert and Jane have won a free trip on the first public space shuttle trip to Astro Station Number 12.

The original form-letter text might have appeared this way:

Dear Mr. N:

We're happy to inform you and Mrs. >N of >T, >S, that you and your children >B and >G have won a free trip on the first public space shuttle trip to Astro Station Number 12.

When the database information-processing software runs the letter text, it searches the data record for a surname to plug in at >N, a town for >T, a state for >S, a boy's name(s) for >B and a girls name(s) for >G. Each time it prints a letter, it searches for the appropriate information from each data record. On a previous preparation sort, only the records that indicated married couples with children might have been selected for processing.

The same type of information processor works with the enhanced databases, only instead of preparing just mailing labels or form letters it can be used to create estimates, reports, summaries, and the like. The information-processing software might take daily totals (of anything) and prepare weekly or monthly summaries, even projections if accessed to some form of spread-sheet program. As a general rule, the more sophisticated the database the greater the number of enhancements that permit the data to be crunched into reports, projections, and the like; even as a modifications for data in other records.

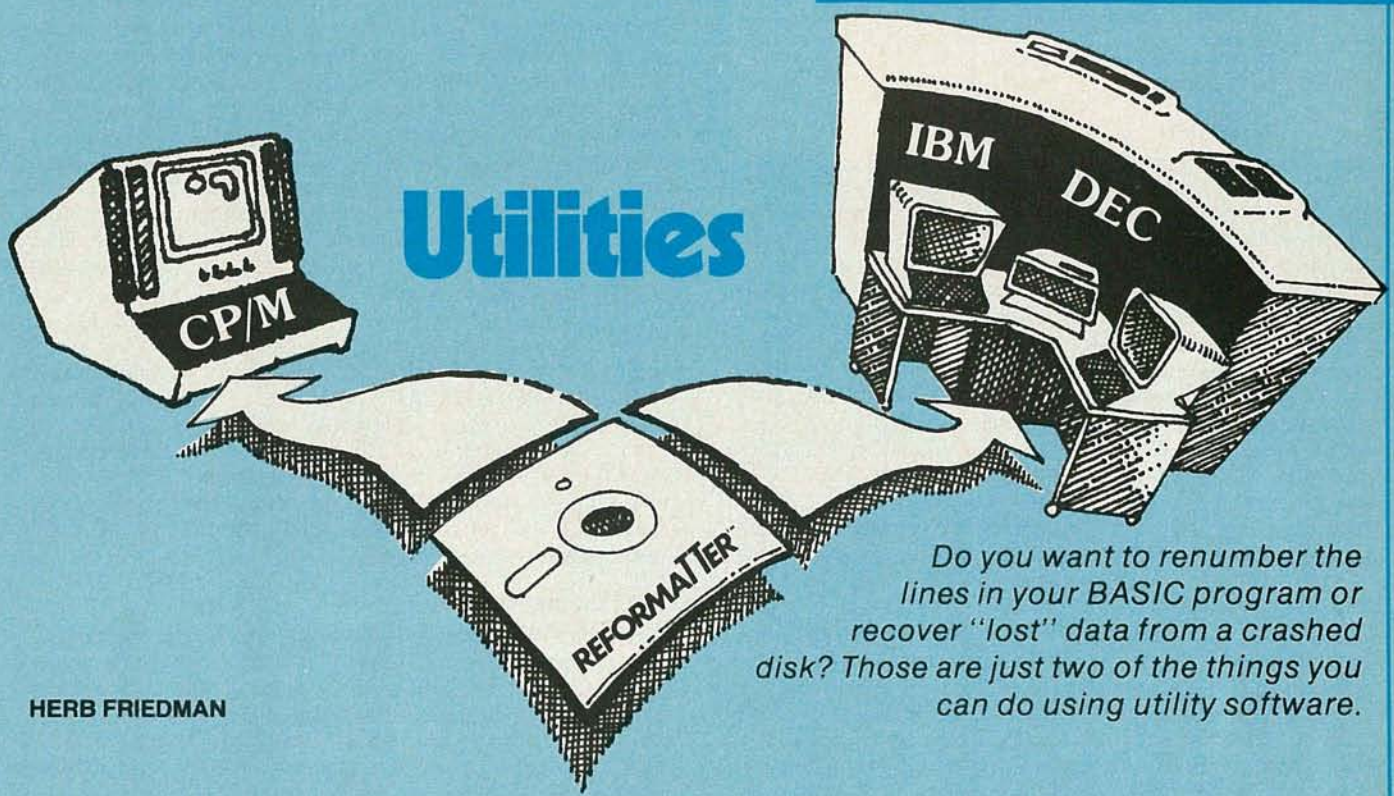
Dial-up databases

While personal-computer databases concern themselves primarily with cataloging, business data, and mailing lists, at present, there are many different types of databases available to users of personal computers through the time-sharing services such as MicroNet, Compuserve and The Source.

Just to mention a few typical examples, there are the DOW Jones Stock and financial reports, feature articles from the *New York Times* (among other newspapers), even feature stories from the news services and...well, just about anything that someone would want to read about, and would pay a reasonable price to use...can and will be put into a general-access database. The important thing to bear in mind is: If it's in a general-access database, it can probably be downloaded into your personal computer for use at your convenience. **R-E**

Radio Electronics
YOUR OWN Computer
APRIL 1982
Software

Utilities



Do you want to renumber the lines in your BASIC program or recover "lost" data from a crashed disk? Those are just two of the things you can do using utility software.

HERB FRIEDMAN

ALTHOUGH THERE ARE MANY PIECES OF SOFTWARE FOR PERsonal computers that are described as being "utilities," there is no hard and fast rule as to what that means. Until recent times, it was more or less accepted that a utility was software that did not usually "stand-alone;" that its purpose was to support other software. Or, a utility was software that modified the monitor or operating system of a computer to provide a function or capability not originally present in that computer.

Today, just about any software that isn't a "language" or an interactive program might be called a utility; and the modern definition of a utility intended for use with a personal computer could be "...software that provides a computer or software function not originally intended, supplied, or supported by the manufacturer of the computer or its software."

As a general rule, utility software is loaded into the computer's memory and remains there in a protected area, acting as a "modifier" for the primary software or computer functions. In some instances the utility software "enhances" and becomes part of the primary software and is then no longer needed.

One of the earliest utilities for personal computers illustrates the "modifier" and "enhancer" modes.

Early personal computers—and even present day budget-priced or basic computer systems—were supplied with a BASIC in ROM as firmware, or one that was loaded into RAM from a cassette tape. When the user completed writing a BASIC program, the line numbering, which probably started out in neat steps of 10, might have become somewhat muddled. Depending on how much coding was squeezed in during debugging, a program might have wound up with numbering as shown in Table 1.

While that program will run, we have come to expect BASIC programs to have "professional-appearing" line numbering in constant increments (usually multiples of 10). Therefore, one of the earliest popular utilities was "renumbering" software for BASIC programs.

As a "modifier," the renumbering software is loaded into the computer first, followed by the BASIC program. (As far as the computer is concerned, the active software is the "renumber" program. The BASIC program is accepted as a text or data file to be processed by the renumberer.) The renumberer analyzes the BASIC program for GOTO and GOSUB statements and then renumbers the BASIC program without losing the proper GOTO/GOSUB structure.

As an "enhancer," the renumbering software is appended to the end of the BASIC program and becomes part of the BASIC program. It might either remain as part of the program (a poor way to program), or it might serve to renumber the program the first time it is run. The user then saves the new renumbered version and deletes the appended utility from the BASIC program.

When a utility is used to modify existing software, so that the new version can be run independently of the utility (as in the case of the BASIC program we've just discussed), the new software is said to be "enhanced."

As a general rule, enhancement is used to add features or commands to a language—such as *LEVEL III* BASIC, an enhancer for Radio Shack's *Level II* BASIC; or it is used to modify word-processing software, like the special utilities that can be added to Radio Shack's *SCRIPSIT* to permit the use of specialty printers that can subscript, superscript, and backspace.

Another early utility—something more popular now than ever—is some form of "terminal" software that converts a personal computer into a communications terminal that can "talk" or "communicate" with large time-sharing computers and data bases such as *The Source, MicroNet,* company-owned mainframes, and computer bulletin-boards. More im-

TABLE 1

```
10 PRINT
13 INPUT "TYPE YOUR NAME: ", N$
15 GOSUB 35
18 PRINT
20 PRINT "YOU HAVE PRINTED YOUR NAME"
30 PRINT "TEN TIMES. TRY PRINTING"
33 PRINT "USING LOWER CASE."
34 END
35 FOR I=1 to 10
36 PRINT N$
38 NEXT I
39 RETURN
```

portant there is an ever increasing need for personal computers to be able to communicate (swap data and programs) with other personal computers.

While all one needs in order to communicate with remote computers and data bases is a connection to the telephone line (through a modem) and a device capable of transmitting and receiving ASCII code—called a "dumb" terminal—the fact of the matter is that a personal computer isn't "dumb" enough to function directly as a terminal. Even if connected to a modem, and then into the telephone line, no personal computer currently available can function directly as a terminal.

However, if the computer has some form of RS-232 I/O port, an inexpensive utility will allow the computer to function as a terminal. There are several "terminal" utilities available for all popular models of personal computers. More important, with a utility that converts the computer into a "smart" terminal, the user can download or upload programs and data to or from other personal computers equipped to do so, as well as communicate with time-sharing systems. In fact, there are utilities—again available for most popular computers—that allow a personal computer to serve as a host computer, a "host" computer being one that is accessed by terminals and by other computers functioning as terminals.

Dumb, smart and genius

Before going any further, let's find out how a personal computer utility creates dumb and smart terminals, and host computers.

A "dumb" terminal is comparable to a simple teletypewriter, in that it can only send and receive characters. It must be connected to a host computer (one that accepts I/O from "remote" terminals), operates in real time (it transmits ASCII code the instant a key is depressed), and displays characters on a CRT or printer upon receiving an ASCII transmission from the computer. Even if a computer is equipped with 48K or 64K of RAM, and as many disk drives as it can support, if the utility only permits the computer to function as a "dumb" terminal; all it can do is transmit and receive in real time to or from another computer.

A "smart" terminal utility is completely different because, while it turns a computer into a terminal, it also allows the computer's RAM and storage system to be used on demand by the user or by the host computer.

Programs and data stored on disk or tape—or in memory—can be exchanged between the personal computer and the host computer, or between the personal computer and another smart terminal. An enhanced smart-terminal utility such as *OMNITERM* can convert binary data for transmission as ASCII code, receive or transmit directly in binary, and even convert from one format to another—like receiving IBM's EBCDIC code and storing it as ASCII and vice versa. One common use of a smart-terminal utility is to swap programs directly between different computers—like from a Heathkit to a Radio Shack—by having the utility convert the output signals to ASCII.

When we say that a personal computer can function as a "host," we mean that it will function as a full-blown computer for a remote smart or dumb terminal. The best way to illustrate that function is with a practical example. Imagine, if you will, that you are an electrical contractor, and your personal computer's active data-storage contains all the formulas, order information, and other data you need for calculating a sound-reinforcement system. The computer is connected through an auto-answer modem to a telephone line. You're out in the field calculating a job and your client wants his information *now!* You connect a portable terminal to a convenient telephone, dial your office, and when the auto-answer modem connects the computer to the phone line you can operate the computer from your portable terminal just as if it were there in the room with you. Your personal computer is serving as a host computer for your portable terminal. Similarly, another personal computer functioning as a smart or dumb terminal could access your host computer through the modem connection. (To protect your system or its database, from "unauthorized" use, you could provide for password protection at different levels.)

An unusual utility package is something called *Micro-Courier* for Apple computers. Essentially, it's a package of utilities that permits the *Apple II* computer to function as a "communications center" for access to Western Union, TWX, Telex, international cable-services, and other computers. It contains its own data base, allows pre-writing of messages, will receive remotely under control from the "other end," and it can receive and transmit under the control of a clock/calendar for fully unattended operation. It even features answer-back validation.

Tapping the "biggies"

Another utility certain to become one of the outstanding ones of the 1980's is *REFORMATTER*. Basically, it is a translator/convertor that allows personal computers to transfer data to and from "large" mainframe computers that use IBM protocols. In the "plain vanilla" configuration *REFORMATTER* transfers files between computers using CP/M, CP/M-like, or MP/M operating systems and those using IBM protocols. It performs an automatic ASCII-to-EBCDIC character-code conversion, and organizes data to the IBM 3740 disk data-entry format. It allows direct access to IBM disks for both examining and altering data.

Essentially, it gives the personal-computer user access to any software that can be transferred through IBM-format equipment. For example, it would permit a user to develop programs in FORTRAN, COBOL, Pascal, or other languages on a personal computer, convert them to IBM format, and, finally, run them on a large time-sharing system. The process could also be worked in reverse, with the original work being done on the larger system and then transferred to the smaller computer system.

In short, *REFORMATTER* moves the personal computer away from the need for special personal-computer software and into the world of commercial mainframe and time-sharing software.

Special versions of the utility are presently available for *CP/M*-to-DEC (Digital Equipment Co.)—which allows access to the DEC library—and Radio Shack *Model II TRSDOS*-to-IBM.

Because it permits direct access to the vast library of mainframe software, we can expect versions to become available for most, if not all, of the personal computers.

Users of Heath/Zenith computers also have a chance to tap the software used on the "biggies"—and the other Heath computers—through disk operating-system conversion utilities. Those utilities will convert ASCII or non-ASCII files that run under *CP/M* to run under *HDOS* (Heath's own DOS) and vice versa. ASCII files, such as a program in BASIC, will run under either DOS. However, while non-ASCII binary files can be copied they cannot be run. Those DOS conversion utilities, therefore, permit tapping only ASCII-based software and data.

UTILITY PROGRAM SUPPLIERS

In addition to the manufacturer of your computer, utility software can be obtained from many independent software companies. While, due to space, we could only include a few of those companies, the following listing may be of some help in finding the utility programs you need.

ALLEN ASHLEY
395 Sierra Madre Villa
Pasadena, CA 91107

BASIC BUSINESS SOFTWARE
PO Box 2032
Salt Lake City, UT 84110

BT ENTERPRISES
171 Hawkins Road
Centereach,NY 11720

COMPCO
8705 North Port Washington Road
Milwaukee, WI 53217

COMPUTER DESIGN LABS
342 Columbus Avenue
Trenton, NJ 08629

COMPUVIEW PRODUCTS INC.
618 Louise
Ann Arbor, MI 48103

CORNUCOPIA SOFTWARE
PO Box 5028
Walnut Creek, CA 94596

CYBERNETICS, INC.
8041 Newman Avenue
Huntington Beach, CA 92647

DATA RESOURCES CORPORATION
Business and Professional Center
8000 East Quincy Avenue
Denver, CO 80237

DELTA COMPUTER SYSTEMS
668 Chenango Street
Binghamton, NY 13901

DIGITAL RESEARCH
PO Box 579
Pacific Grove, CA 93950

DYNACOMP, INC.
1427 Monroe Avenue
Rochester, NY 14618

HAYDEN BOOK COMPANY, INC.
50 Essex Street
Rochelle Park, NJ 07662

HOWE SOFTWARE
14 Lexington Road
New City, NY 10956

INSTANT SOFTWARE
Peterborough, NH 03458

KV33 CORPORATION
PO Box 27246
Tucson, AZ 85726

LIFEBOAT ASSOCIATES
164 W. 83rd Street
New York, NY 10024

LINDBERGH SYSTEMS
41 Fairhill Road
Holden, MA 01520

LJK ENTERPRISES INC.
PO Box 10827
St. Louis, MO 63129

META TECHNOLOGIES
26111 Brush Avenue
Euclid, OH 44131

MICROSTUF, INC.
1900 Leland Drive
Marietta, GA 30067

MICROTECH EXPORTS, INC.
467 Hamilton Avenue
Palo Alto, CA 94301

OMEGA MICROWARE, INC.
220 S. Riverside Plaza
Chicago, IL 60606

PHASE ONE SYSTEMS, INC.
7700 Edgewater Drive, Suite 830
Oakland, CA 94621

RACET COMPUTES
1330 N. Glassell, Suite M
Orange, CA 92667

SNAPWARE
3719 Mantell
Cincinatti, OH 45236

SOFTWARE CONSULTANTS
6435 Summer Avenue
Memphis, TN 38134

SOFTWAREHOUSE INTERNATIONAL
5070 N. Sixth Street, Suite 103B
Fresno, CA 93710

STATCOM CORPORATION
5766 Balcones, Suite 202
Austin, TX 78731

SUPERSOFT ASSOCIATES
PO Box 1628
Champaign, IL 61820

THE ALTERNATE SOURCE
1806 Ada Street
Lansing, MI 48910

THE SOFTWARE PLANTATION
PO Box 44623
Tacoma, WA 98444

TWENTY-FIRST CENTURY SOFTWARE
1607 North Cochran
Hutchinson, KS 67501

UNITED SOFTWARE OF AMERICA
750 3rd Avenue
New York, NY 10017

WESPER MICRO SYSTEMS
14321 New Myford Road
Tustin, CA 92680

Little things mean a lot

While it's nice to be able to tap the software used by the giants of industry, for the average user the most important utilities are those that make possible new or more convenient uses for a personal computer or its associated peripherals.

For example, for $20 there's a utility that permits use of Apple's floating-point BASIC in Apple computers that don't have the *Applesoft* card.

And ten dollars is all it takes to buy a cassette-based "sort" utility with multiple sort-keys. It's not the fastest program of its kind, but it is rock-bottom priced, and not everyone has a disk system yet.

For "hackers" there's a utility that edits a BASIC program, almost instantly changing all PRINT'S to LPRINT'S or vice versa.

Another utility will duplicate Radio Shack SYSTEM format tapes. Yet another—*Locksmith*, for the *Apple* computer—permits copying "protected" disks, the kind that normally only allow two backups. (That gives you a little added protection by solving the annoying problem of what you do when the second backup "crashes.")

Admittedly, little in the way of utility software falls in the ten-to-twenty-dollar range, but much of it is reasonably priced (under $100) and is worth what it costs. For example, there are several utilities that provide a "shorthand" for BASIC programmers that lets them use a shift/single-key entry to represent a complete command, or a user-defined instruction, making entering programs easier.

Again for programmers, there are several compression utilities that will remove blanks and remarks from BASIC programs so the programs take up less memory space and run faster.

Among the most widely used utilities are those for the Epson *MX-80*, one of the most popular printers for the Apple and Radio Shack lines. (The *MX-80* is also sold under the IBM and Hewlett-Packard names.) The printer has many special features that can be accessed only through commands that are transmitted before or during the program. It's easy enough to imbed those commands within BASIC programs, but almost impossible to do so with commercial or assembly-language software.

That problem is overcome by special utilities that permit transmission of the Epson commands directly from the DOS, as commands imbedded within word-processing software, or within a customized assembly-language printer driver. (In most instances the *MX-80* utility is used to enhance the word-processing software so it all becomes one package, similar to the superscript-enhanced *SCRIPSIT* that runs as *Script*.) A special utility is also available that permits the *NEWDOS* JKL function to dump screen graphics to the printer while it remains in the standard mode, thereby retaining all the standard printer-functions while adding the capability of printing screen-graphics.

Not so well known are utilities such as *Disk Doctor* for *CP/M* and *Super Utility* for *TRSDOS*, which are used to recover "crashed" disks. Perhaps it's a defective directory that prevents any use of the disk files, or maybe a file was accidentally erased (actually only the file reference in the directory is erased, unless a new file is recorded in the space occupied by the "erased" file). In many cases, the disk-saver utilities can recreate the directory, or locate files no longer listed in there. Some can virtually repair an entire disk, losing perhaps only 256 bytes.

Sometimes the crashed-disk utilities sound better than they work. Before you purchase any of them, it's a good idea to read their specifications, to see what they don't say they do, or what they say they don't do (there's a difference). Will a utility work with the operating system you use? If so, will it have unusual side effects that might cause the loss of the data you're trying to recover? I raise that point only because the disk software may be irreplaceable, and the wrong utility can cause its loss forever.

Also in the "doctor" category are utilities that test each memory address and indicate on the memory any defects (assuming the RAM that's failed isn't used for the screen display or needed for the utility).

There are utilities that test disk drives, particularly for precision and constancy of speed. (One of the main causes of disk-read problems.) Many "disk test" utilities are provided along with other utilities in a single package, such as *"Locksmith,* for the *Apple,* which contains six utilities of various types—including a disk "bulk erase".

Finally, we come to the source of many of the most convenient and important utilities: the aftermarket disk operating-systems such as *NEWDOS 80, LDOS, VTOS, DOSPLUS*. In many instances the principal difference between what came with the computer and the aftermarket-DOS is the utilities provided in the new software. Certainly, the *NEWDOS* JKL screen-print function is one of the most famous and—to some computerists—the most important. To others, the ability of an aftermarket system to format different disk-track counts, change disk stepping-rates (for faster operation), or copy software from other DOS's is the most important consideration. In fact, the aftermarket DOS-systems are so extensive that much of what formerly was available as discrete utilities is now found all together in the aftermarket DOS. Many users of personal computers will find a "new" DOS offers most of the utilities they want at a moderate price—certainly less than they would have to pay for a collection of individual utilities if they were bought separately.

Something for everyone

We have tried to touch on the highlights of the utility software presently on the market. By the time you read this some may have disappeared, but many others will have appeared. No computer, peripheral, or software can ever meet 100% of everyone's needs. And as long as there is someone who is not satisfied, there is someone else who will write the utility he's looking for. **R-E**