DECEMBER 1982

HEWLETT-PACKARD JOURNAL

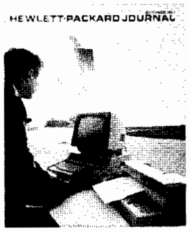
HEWLETT·PACKARD JOURNAL

Technical Information from the Laboratories of Hewlett-Packard Company

DECEMBER 1982 Volume 33 • Number 12

Contents:

In this Issue:



Our cover suggests a potential airfoil design application for two of this month's featured products, the HP-86 Personal Computer and the 7470A Plotter. We're grateful to Pterodactyl, Ltd. of Watsonville, California for their help with the photo. (Their original design calculations were done with an HP-35 Calculator.)

The HP-86 and the HP-87XM, two new members of HP's Series 80 personal computer family, are the subjects of the article on page 3. These computers feature very large maximum memory capabilities, up to 576K bytes for the HP-86 and up to 640K bytes for the HP-87XM. That's quite a lot for a personal computer of this class, and the article tells how they're able to do it. Both computers can talk to a wide range of peripherals and instruments by means of built-in and optional interfaces. A plug-in option, described in this issue beginning on page 8, makes them compatible with the impressive variety of software packages based on the popular CP/M® operating system.

The 7470A Plotter is the latest and most economical of HP's plotters using the low-mass, low-inertia, microgrip paper drive technology first described in our October 1981 issue. In the articles on pages 12, 23, and 27 you can read about the problems its designers had to solve to provide the quality people expect of HP products at the 7470A's very low price. But the full significance of the 7470A isn't just its low price. Its full significance is illustrated by the photograph on page 16, which shows the 7470A being driven by the HP-41C Personal Programmable Calculator. When Chuck Tyler and his group at Hewlett-Packard Laboratories began working on their new plotting technology several years ago, they were motivated by the vision of a plotter for the pocket calculator owner's other pocket. While that goal hasn't been realized, a major milestone has been reached. You can now get high-quality multicolor ink-on-paper graphic output from a battery-powered calculator. What makes it possible is the Hewlett-Packard Interface Loop, or HP-IL, a new interface system for battery-powered devices. We're planning to cover the technical aspects of the HP-IL in next month's issue. By the way, the HP-86 and HP-87XM are both compatible with it, too.

Switching ac power on and off may seem simple; we do it every time we turn on a light. But if you're concerned about electromagnetic interference or need a switch that'll operate a million times before it wears out, you have some problems. The article on page 34 describes a new ac power switch, Model 14570A, that's designed for ten years of service switching power on and off under computer control in automatic systems, where interference can't be tolerated and 50,000 operations, the lifetime of a typical relay, may occur in six months.

-R. P. Dolan

Editor, Richard P. Dolan • Associate Editor, Kenneth A. Shaw • Art Director, Photographer, Arvid A. Danielson • Illustrator, Nancy S. Vanderbloom
Administrative Services, Typography, Anne S. LoPresti, Susan E. Wright • European Production Supervisor, Henk Van Lammeren

Extended Memory and Modularity Are Added to the Series 80 Computer Family

HP's newest Series 80 computers, the HP-86 and HP-87XM, provide memory capacities up to 640K bytes, different combinations of built-in interfaces, and for the HP-86, a modular system configuration.

by John T. Eaton, Andrew W. Davidson, and William R. Frolik

IN 1980, HEWLETT-PACKARD introduced the HP-85, the first member of the Series 80 computer family.[1,2] It was designed to aid the scientist or engineer in collecting, analyzing, storing and displaying data. With its own internal tape drive, printer, display, and four expansion ports, the HP-85 provides effective portable solutions to many data processing problems.

The Series 80 family is now expanded by two new members, the HP-86 and HP-87XM. These computers perform all the workstation (nonportable) functions performed by the HP-85 and many business functions not previously available from Series 80 products. The block diagrams of these two new computers are basically the same, differing only in the amount of internal memory, built-in interfaces, and video display configuration (Fig. 1).

The HP-86 (Fig. 2) is designed to provide the nucleus for the lowest-cost Series 80 workstation system. It has 64K bytes of internal RAM, uses an external video monitor, and has built-in interfaces for two flexible disc drives and a printer. The design provides a low-cost computer that allows the user to select peripheral components from a variety of sources.

The HP-87XM (Fig. 3) was designed as the premier personal computer of the Series 80 family. It has 128K bytes of internal RAM and a built-in 80-character-by-24-line CRT display. It uses a built-in HP-IB* interface to communicate with external disc drives, plotters, and printers.

Both computers have four expansion ports which can be used to enhance their capabilities. These ports are completely compatible with all Series 80 I/O interfaces. ROM and RAM modules for the HP-86 and HP-87XM are not interchangeable with the HP-85's ROM and 16K RAM modules. RAM modules are available in 32K, 64K, and 128K byte sizes. Up to four of these memory modules, in any combination, can be inserted into the ports of the HP-86 to increase its total RAM up to 576K bytes, and in the HP-87XM to increase its total RAM to 640K bytes.

Software

With a large number of HP-85s already in use, it is desirable for new Series 80 computers to be compatible with existing HP-85 BASIC language software. For this reason, a BASIC language interpreter and operating system is provided. Because there are some differences in the internal representation of programs in these two new computers compared to the HP-85's internal representation, the HP-86 and HP-87XM have a built-in translator to convert an HP-85 BASIC program to their internal representation after it is loaded. The translated program can then be run or stored for future use as an HP-86 or HP-87XM program.

*Hewlett-Packard Interface Bus, HP's implementation of IEEE Standard 488 (1978).

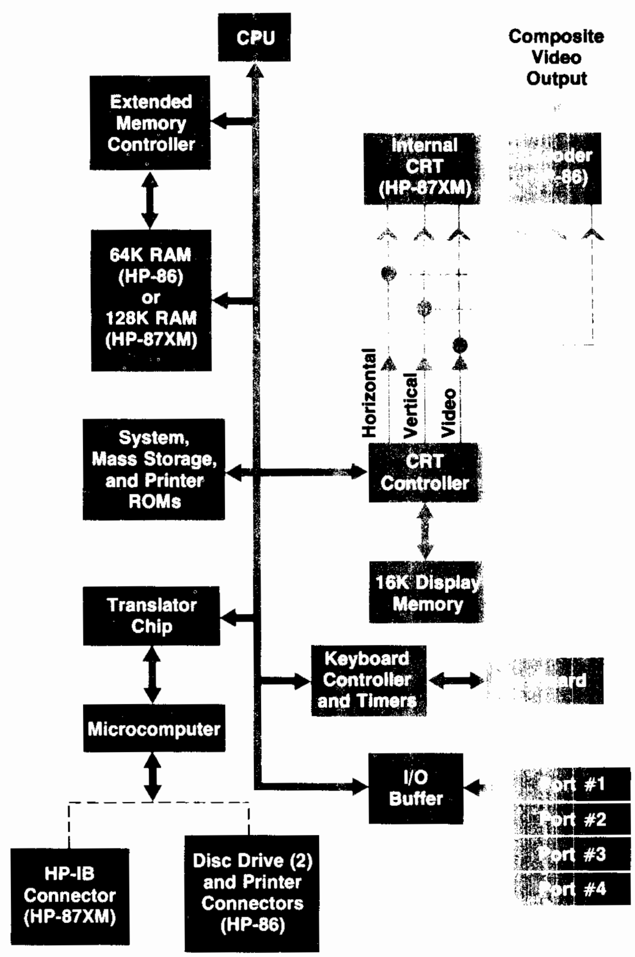


Fig. 1. *System block diagram for the HP-86 and HP-87. Other than the differences in video output configuration, built-in interfaces, and amount of internal memory, the two computers use the same electronic system.*

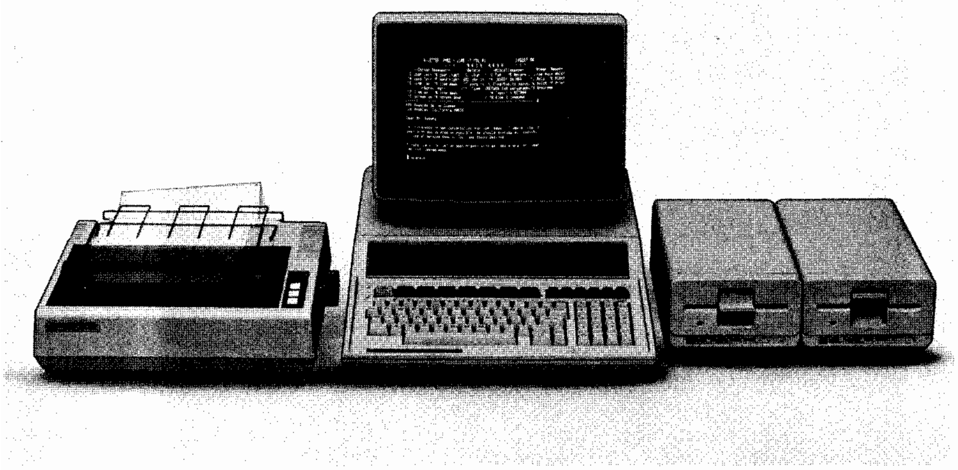


Fig.2. *The HP-86 Computer is the nucleus of a modular computer system. It can drive up to two HP 9130A flexible disc drives and a Centronics-compatible printer directly with its built-in interfaces. Video output is displayed on an external monitor. Both 9-in (82912A) and 12-in (82913A) monitors are available from HP.*

Binary programs written for the HP-85 will not run on the HP-86 or HP-87XM. These programs do not know how to access the additional memory and do not know the locations of system subroutines in the HP-86 and HP-87XM ROMs. Therefore, HP-85 binary programs must be modified and recompiled before they can be used in the HP-86 or HP-87XM.

Most of the HP-85 software, including its BASIC language and on-screen editing features, is available in the HP-86/HP-87XM. Several enhancements are added. Alphanumeric line labels and multicharacter variable names allow the programmer to use more meaningful names. The programmer can also use spaces to indent portions of program listings and make them easier to read and understand. These spaces are retained by the HP-86 and HP-87XM for future program listing and storage.

With larger CRT displays, the new Series 80 computers can support a larger number of user-defined softkeys, so the number is increased from the eight on the HP-85 to fourteen. These are also enhanced by making them default typ-

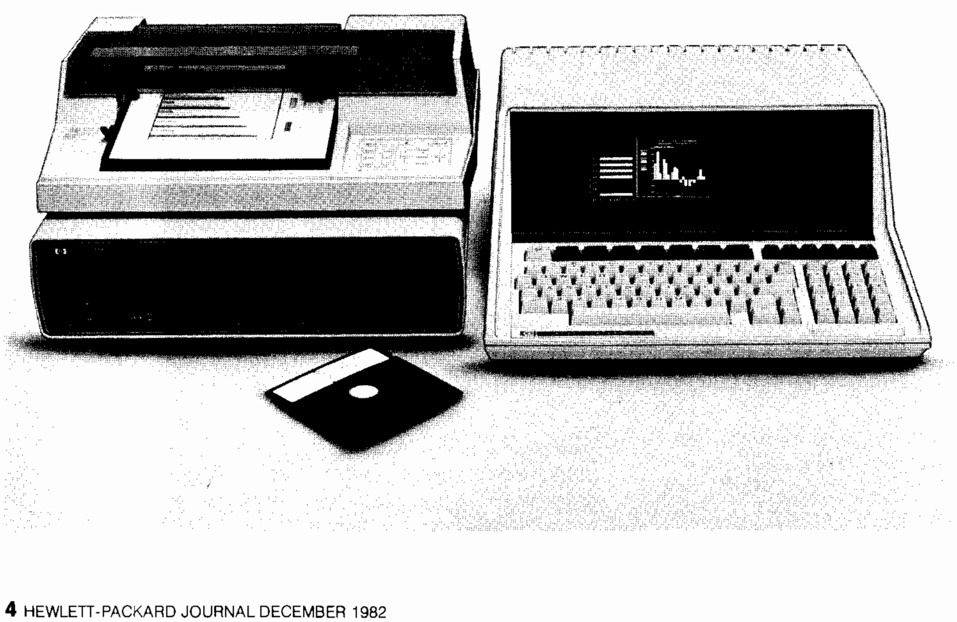


Fig. 3. *The HP-87XM Computer is the premier member of the Series 80 family. It features a built-in CRT display and an HP-IB interface for communication with disc drives, printers, and plotters. Its large (128K bytes) internal user memory can be expanded up to 640K bytes by adding RAM modules to its four expansion ports.*

ing aids when the new computers are in the calculator mode. Fourteen commonly used commands can be displayed on the screen at a touch of a key. The user need only type in the parameters and press the **ENDLINE** key for the command to be executed. These aids can also be redefined by the user to suit individual programming preferences.

User-defined functions are changed to allow more than one parameter to be passed to the function. Capabilities for handling one- and two-dimensional string arrays are also added.

Memory

A major feature of the HP-86 and HP-87XM Computers is their ability to accommodate extremely large amounts of user memory. This enables the user to manage larger programs and data files than previously possible on a personal computer.

With the development of memory intensive programs such as VisiCalc® and linear programming software, the need for more and more memory becomes apparent. While only a couple of years ago the maximum 64K-byte memory option for most 8-bit machines was considered an expensive luxury, now it is considered a minimum necessity. This has been caused by the rapid drop in the cost of RAM devices, coupled with the increased density with which RAM can be packed on logic boards.

The HP-85 has a maximum addressable memory space of 64K bytes (see Fig. 4). The first 32K bytes are reserved for the ROM operating system and external plug-in ROMs. The next 16K bytes are internal RAM. A RAM module can be installed that occupies the next 15.75K bytes. The last 256 memory addresses are used for I/O devices and access to the CRT display memory.

One of the design objectives of the HP-86 and HP-87XM was to provide a minimum of 32K bytes of user RAM. While this could have been done by making a simple integrated circuit mask change for the HP-85 RAM controller, it still would have been unable to control more than a 64K address space. A new dynamic RAM controller called the extended memory controller (EMC) was developed to access memory addresses beyond the 64K limitation of the HP-85 controller. The EMC serves two functions. First, it is an upgrade of the HP-85 RAM controller capable of handling either 16K or 64K dynamic RAMs (the HP-85 controller can handle only 4K and 16K RAMs). Second, it provides for 24-bit addresses

VisiCalc is a registered trademark of VisiCorp.

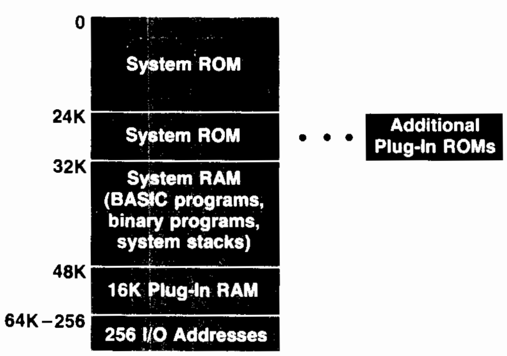


Fig. 4. *Memory map for HP-85 Computer.*

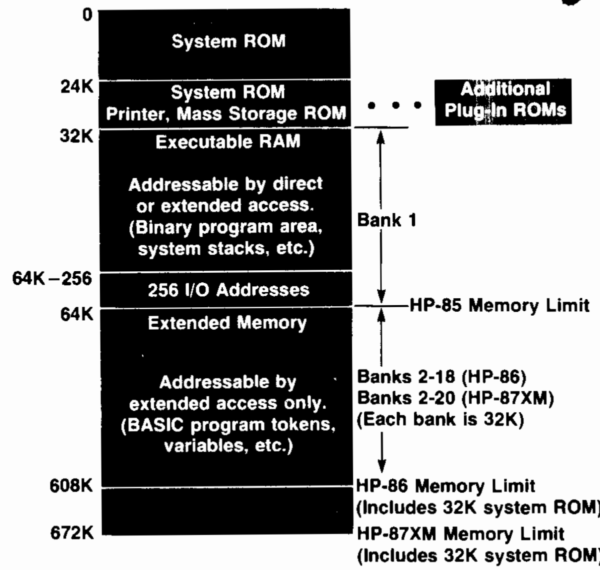


Fig. 5. *Memory map for HP-86 and HP-87XM Computers.*

which makes it possible to address an effective memory space up to 16 megabytes, considerably more than the current physical memory limit (including the 32K-byte system ROM) of 672K bytes for the HP-87XM or 608K bytes for the HP-86.

Since many microprocessors (including the HP-85's) are designed to access only 64K bytes of information, some special techniques are needed to expand memory beyond this limit. Some systems accomplish this by page swapping in which the central processing unit (CPU) can enable or disable many different RAM pages. These systems have more than 64K bytes of RAM, but only 64K bytes are enabled at any one time. This and other techniques for memory expansion were considered, but most were rejected because they placed too much of a burden on the system software to be efficient. The method chosen for the EMC in the HP-86 and HP-87XM permits access to memory outside the directly addressable 64K range without burdening the operating system with excessive page swapping requirements.

The extended memory blends in with the Series 80 architecture in a very direct manner (Fig. 5). The software specifies a 24-bit physical address and the EMC hardware manipulates it under software control. The result is similar to adding a 24-bit stack pointer to the system, except that the pointer is not kept inside the CPU, but rather in each memory controller.

Each EMC can control either 32K, 64K or 128K bytes of dynamic RAM. Since the total memory space is broken into 32K-byte banks, this represents either one, two or four memory banks per EMC. A select-code input into each EMC determines where the RAM it controls is located in the overall 24-bit memory address space.

Each EMC contains two 24-bit address registers that the CPU can read or write via a set of dedicated I/O addresses. To read an extended memory location, the CPU must write its three address bytes (low-order first) into one of the 24-bit address registers in each EMC. The CPU then performs an indirect read operation to all EMCs, but only the EMC controlling the desired memory location responds to the indirect read by supplying the desired data byte to the CPU. Writing is performed in the same manner using an indirect store command.

While this may seem like a lot of overhead to read or write a single byte, the EMC contains some features that greatly simplify the operations. The 24-bit registers in the EMC are provided with some autoincrement and autodecrement features similar to the stack pointer used in many microprocessors. Although it takes three bytes to access a single memory location, the CPU can access the next higher (or lower) byte simply by using another indirect load. This allows the CPU rapid access to an extremely large number of consecutive data bytes. Since all EMCs load and modify their address registers together, they always contain the same values. The CPU can start using data from one EMC and continue reading past its assigned memory space into the address space of a second EMC. Since the controllers are tracking one another, the operation is passed between them without requiring any attention from the CPU.

This type of extended memory operation is especially suited to the Series 80 architecture. The built-in BASIC language interpreter considers the entire BASIC program to be a large data file. By using the other 24-bit address register in each EMC as a program counter, the interpreter can step sequentially through the entire program, regardless of its length or location.

Taking advantage of the architecture of the Series 80 processor,[2] some special features were added to the EMCs. The processor has a number of multibyte commands that can operate on data one to eight bytes at a time. Because multiple bytes are loaded and stored in the same sequence, the EMC has to know the number of bytes being operated on to use these commands with the autoincrement/autodecrement features. For example, if the command is to autodecrement and store five bytes, then the EMC has to decrement its pointer by 5 before the bytes are stored. This is done by allowing the EMC to monitor the instruction-fetch line from the CPU. Each EMC keeps track of the current instruction, and therefore knows what number of bytes is being accessed. This operation is transparent to the CPU and requires no additional processing time.

Display

The HP-85 has a built-in CRT that displays 16 lines of 32 characters each or a graphics display of 256 by 192 dots. While this is adequate for most analysis and controller applications, it can be very limiting when trying to do word processing or other business applications. An 80-character/line display is considered necessary to perform these applications properly. Although the Series 80 computers are not intended to be a family of business computers, it is realized that many scientists and engineers have enough word processing and report generation needs to justify using a computer.

Displaying 80 characters on the HP-85's screen is impractical because the resulting small characters would be difficult to read. The increasing concern that computer buyers have for ergonomics and human-engineered computers dictates that the size of the CRT must be increased to display 80 characters per line.

How the CRT size is increased depends upon the product definition. The HP-87XM is defined to be the top of the line for the Series 80 family. It is designed with a built-in 228-mm-by-102-mm high-resolution CRT.

The HP-86 is defined to be the nucleus of a low-cost modular computer system. A composite video output is provided to drive an external monitor. This gives users some flexibility in pricing, in that they can purchase the display that best fits their needs and budgets. The modular concept also allows a user to select from various sizes and colors of display monitors. A separate monitor also can be adjusted for optimum viewing distance and angle.

The HP-86 and HP-87XM CRT controller is designed to provide an 80-character/line display in which a user can specify either 16 or 24 lines. It has two graphics modes, depending upon how much of the 16K bytes of CRT memory that a user wants to allocate to graphics. NORMAL graphics gives a display of 400 by 240 dots and allows switching between the alpha and graphics modes. GRAPHALL mode allocates all the CRT memory to graphics and produces a display of 544 by 240 dots. CRT graphics operations with the new CRT controller are faster than HP-85 graphics.

Graphics programs written on an HP-86 will run on an HP-87XM (and vice versa), but since the sizes and shapes of the CRT display are different, they will produce differently shaped plots. The HP-87XM provides an isotropic display with a 1:1 aspect ratio. The physical distance between the dots is the same in the horizontal and vertical directions. The monitor selected by the user for the HP-86 determines the aspect ratio, but a SCALE statement is provided to allow the display to be adjusted to any monitor.

Internal Interfaces

Built-in interfaces are other features of the HP-86 and HP-87XM. With the many Series 80 I/O interface and expansion modules available, some HP-85 users find that the four expansion ports are quickly filled. The problem could be worse for HP-86 and HP-87XM users because they also have an assortment of RAM modules to choose from. To help relieve this problem, the new Series 80 computers are designed with commonly used interfaces built-in. Not only does this free more expansion ports for specific user applications, but the cost of a built-in interface is less than that of a plug-in module. Additional ports are made available by including the mass storage and printer ROMs inside the HP-86 and HP-87 instead of having to use two ROM drawer positions as required in an HP-85 system.

The HP-87XM contains a built-in HP-IB interface. It performs all the functions done by the HP 82937A HP-IB Interface Module. It has an output connector on the back panel along with switches to set its address and control mode. Most HP-87XM systems are able to do all necessary I/O, mass storage, and printing operations using only this interface.

The HP-86 takes the built-in interface concept one step further. The most common use for the HP-IB in the Series 80 family is to connect a computer to a disc drive and printer. In the HP-86, a disc controller and printer interface are built-in to eliminate the need for an HP-IB interface. The HP-86 uses the HP 9130A Flexible Disc Drive (shown in Fig. 2). This unit contains the same drive mechanism as the HP

82901M Series of 5.25-inch flexible disc drives. The only difference is that the 9130A drive mechanism is individually packaged and receives all of its signals and power from the HP-86 mainframe. The HP-86 user can purchase two of these units for less money, and have the same storage capability as provided by an 82901M Flexible Disc Drive. The savings comes from not having to purchase a separate power supply and controller.

The built-in parallel printer interface provides a Centronics-compatible connector that will work with the HP 82905B (Option 242) Printer or many commercially available printers.

By not using the HP-IB, the HP-86 system is able to lower the cost of mass storage capability. If the HP-IB is required for other devices such as plotters or instruments, an 82937A HP-IB Interface Module can be plugged into any free expansion port in the HP-86's back panel.

Although the HP-86 internal interfaces and external devices are different from those used by the HP-87XM, the two mass storage systems are completely software compatible. The HP-86 interfaces are designed to emulate an HP-IB with a dual disc drive and printer on it. An HP-86 user sees what looks like an HP-IB card with a dual disc drive at address 0 and a printer at address 1. Additional flexible disc or even hard disc drives can be added by plugging an HP-IB interface card into one of the HP-86's four expansion ports.

Normally all Series 80 I/O is done through interface cards which communicate with the CPU using an integrated circuit called a translator chip. An I/O card contains a microcomputer that is programmed to talk to the CPU, using an I/O protocol, and operate an external interface bus. For example, the CPU orders an HP-IB card to send commands and data over the bus to a disc drive. Another processor at the disc drive receives the commands, performs them and passes the results back to the CPU.

In the HP-86, a single microcomputer is used to do disc and printer operations. It talks through the translator chip with the CPU and emulates the I/O card protocol. It also operates the disc drive(s). When the CPU orders it to send data or commands to a disc, the microcomputer stores them in memory, and then performs the requested operation. The CPU does not know or care whether it is using a full HP-IB and external disc drive system or the emulated system, because it uses exactly the same command sequence in either case. The interface also recognizes when data is being sent to a printer and outputs the data on the Centronics-compatible connector on the back panel of the HP-86.

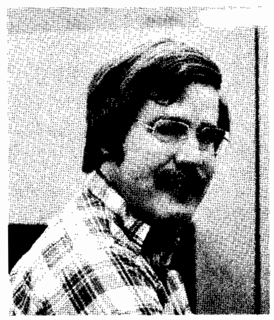
Acknowledgments

Many contributions are needed to develop products like the HP-86 and HP-87XM. Wan Cheng Chan designed the CRT controller, Robert Orozco, Lori Cook, and Ching Chao Liu wrote the BASIC language interpreter, Joe Sikich did the logic design, Jim Bausch designed the CRT electronics and power supply, Ching Chao Liu conceived the EMC scheme, Ella Duyck designed the packaging, and Mike Moore developed the XM version of the HP-87. Jerry Erickson was the HP-87 project leader.

Cliff Cordy designed the HP-86 power supply, Lonnie Ford was responsible for the video monitors, and Tom Pearo designed the HP-86 packaging. The project leader for the HP-86 was Clement Lo.

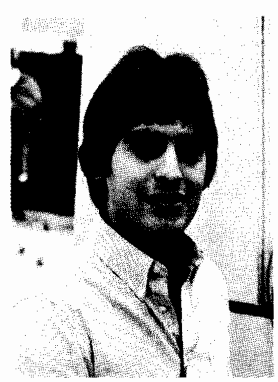
References

1. Complete issue, six articles, Hewlett-Packard Journal, Vol. 31, no. 7, July 1980.
2. T.R. Lynch, "A Custom LSI Approach to a Personal Computer," Hewlett-Packard Journal, Vol. 31, no. 8, August 1980.



John T. Eaton
John Eaton was born in Vincennes, Indiana, and is a graduate of Purdue University, receiving a BSEE degree in 1980. He joined HP that same year, was an R&D engineer for the HP-86 and a production engineer for the 82901 Disc Drive, and now is working on new product design. John served six years in the U.S. Navy before attending Purdue. He lives in Corvallis, Oregon, and enjoys backpacking, folk dancing, whitewater rafting, and cross-country skiing.

William R. Frolik
Bill Frolik received a BA degree in physics from Willamette University, Oregon in 1979 and the BSEE and MSEE degrees from Stanford University in 1979 and 1980, respectively. He then came to HP, worked on the logic design team for the HP-87, designed the EMC for the HP-86 and HP-87, and now is involved with CMOS ROM design. Bill is co-author of one other article on the HP-87XM and lives in Corvallis, Oregon where he was born. His outside interests include photography, travel, cycling, skiing, piano, and computer programming.



Andrew W. Davidson
Andy Davidson was on the logic design team for the HP-87 and designed the memory modules for the HP-86 and HP-87. With HP since 1980, he now is a production engineer for Series 80 products. Andy was born in Abington, Pennsylvania and attended Clarkson College of Technology, New York, earning a BS degree in electrical and computer engineering in 1980. He is married and lives in Corvallis, Oregon. Andy enjoys jogging, going to concerts, and playing guitar, and is interested in audio equipment.

Module Brings CP/M to HP's Latest Series 80 Computers

This small computer system plugs into the HP-86 and HP-87XM Computers to allow them to use the wide variety of CP/M-compatible software available to the personal computer user.

by Timothy V. Harper

THE CP/M® OPERATING SYSTEM is the basis for some very well written software to help make your computer more friendly and useful. To make it possible to use this software in the HP-86 and HP-87XM Computers, Hewlett-Packard developed the HP 82900A CP/M System Module.

The main focus of the 82900A design project was to develop a simple, cost-effective solution to providing CP/M capability and still have the power of the host computer available. The 82900A CP/M System Module is easy to use. The module plugs into any available I/O expansion slot in the back panel of an HP-86 or HP-87XM Computer (Fig. 1), a CP/M disc* is inserted into an attached drive unit, and the system is turned on. In a few seconds the second computer system provided by the 82900A for CP/M operation is available for use.

System Implementation

This new computer module uses a Z80A as its master processor and the custom Series 80 processor as a slave I/O processor. The I/O section (BIOS) of the CP/M operating system is modified for use with HP Series 80 products. A binary executive communicates with the CP/M module to make the HP-86 or HP-87XM host processor behave as a slave I/O processor.

The 82900A system has some special features most implementations of CP/M do not offer. The HP-86 and HP-87XM can address a great deal of external RAM, but CP/M only recognizes the 64K of RAM addressable by the Z80A. This implementation of CP/M uses the mainframe RAM as a print buffer, thus providing a pseudo print spooler.

For easy installation of CP/M application software, the 82900A offers subsets of two different terminal screen control protocols. Both the HP-262X and Soroc 120 protocols are supported to the extent necessary under CP/M.

Hardware/Software Description

A key to this project's success has been its short development time. From start of investigation to production was only nine months. To adhere to such a tight schedule, it was important to keep the design simple and stay with the original design definition.

*Programs must be stored on the disc using HP's disc format (see pages 20 to 24 in the June 1980 issue of the HP Journal) to be read properly by the disc drive.

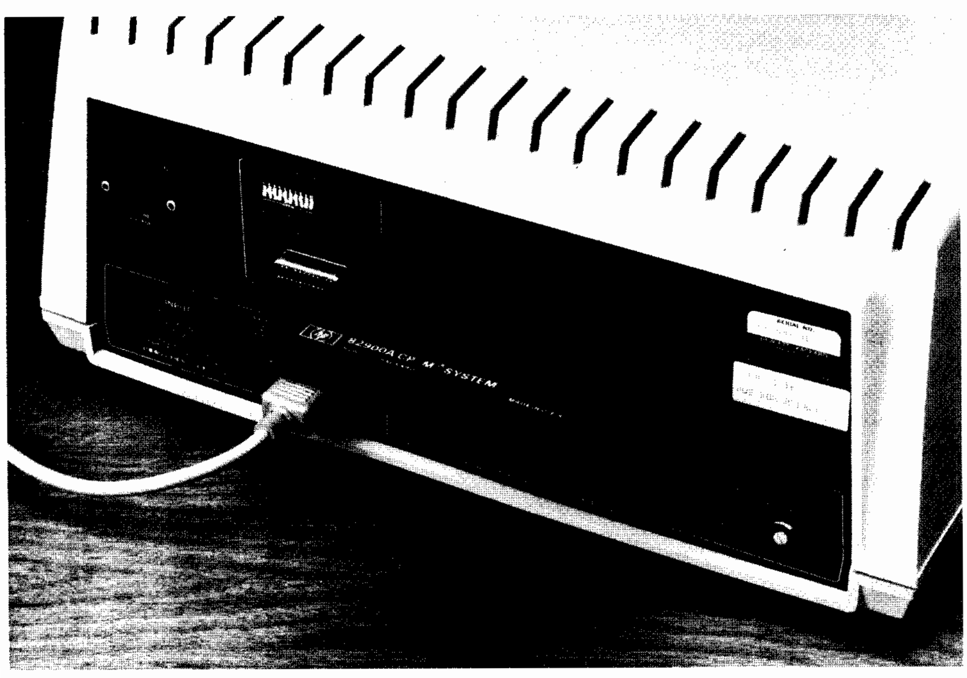


Fig. 1. *The HP Model 82900A CP/M® System Module is a small computer system containing a 64K-byte RAM that plugs easily into the back panel of an HP-86 or HP-87XM Computer. Its addition to these two Series 80 Computers allows their use of software designed for the CP/M operating system.*

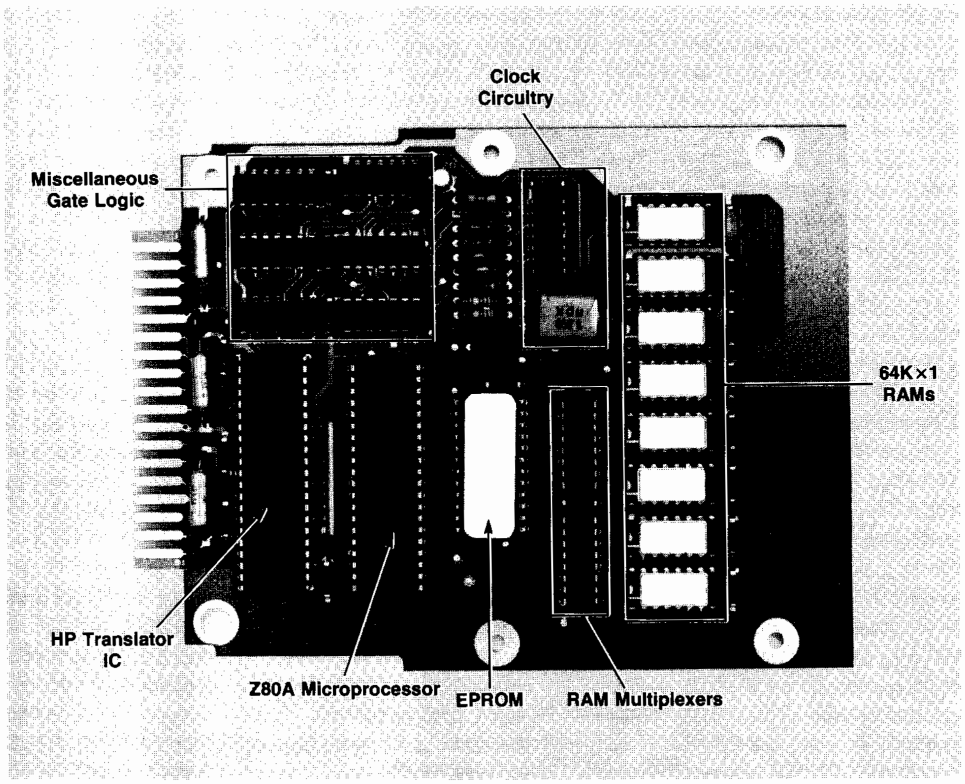


Fig. 2. *The electronics for the 82900A is mounted on a single printed circuit board.*

Cost was a concern too. Not many people would be willing to pay, for example, $1000 U.S. more for CP/M capability on their HP system. Another big problem was designing a hardware system with 64K of RAM and a 4-MHz Z80A microprocessor that uses less than 1.7 watts. Only about 130 cm^2 of printed circuit board space was available in the standard Series 80 module, not to mention a height constraint of 0.7 cm. Just a few years ago a system like this would not have been possible, but today's technology has provided the necessary pieces, such as 64K RAMs.

The hardware is simple and straightforward in design. The 82900A case consists of a standard-size plug-in module designed for the Series 80 computers. Inside the plastic case is a single printed circuit board (Fig. 2) that has a Z80A microprocessor, a 2K × 8 ROM, eight 64K × 1 dynamic RAMs, a translator chip, some miscellaneous LSTTL** gates, and discrete devices. The translator chip is a custom integrated circuit manufactured in HP's Corvallis, Oregon, facility. It is very similar to the translator chips found in all the I/O modules for the Series 80 computers and provides a convenient way to interface to the Series 80 bus.

The architecture of the system (Fig. 3) is straightforward. The bidirectional data bus allows communications between the CPU, ROM, RAM, and translator chip. The address bus of the Z80A extends to the ROM and is multiplexed onto the RAM. The translator chip resides at two of the I/O addresses of the Z80A. One Z80A port is the translator's address register and the other is the data register. Any I/O instruction turns the ROM off and enables the RAM.

The miscellaneous logic can be divided into five functional areas: oscillator and clock driver circuitry, ROM select logic, wait state generation, Z80A-translator communication control, and RAM control logic.

A simple Colpitts oscillator circuit is used to provide an 8-MHz sine wave. This signal is divided by two to create the 4-MHz, 50%-duty-cycle, square-wave clock necessary to run the single-board computer.

The 2K × 8 boot ROM is switched into the address space whenever the module is reset from the mainframe (a command is provided in the BIOS so that the Z80A can reset the system if necessary). At power-on, the module is reset and the processor begins executing code from the boot ROM. The system is organized so that all memory reads are initially from ROM and all memory writes are to RAM. The boot ROM is switched off by executing any I/O instruction, that is, "INPUT port" or "OUTPUT port" in the Z80A software, and then all read or write operations are from or to RAM, respectively.

Wait state generation is necessary on instruction fetches to allow the use of slower 200-ns RAM. This causes a processing speed loss of 10 to 15%, depending on the software being executed. This was considered a good tradeoff for the cost and power saved. This circuitry is implemented with a spare flip-flop and two gates.

The translator chip was originally designed to interface with the 8049 microprocessor, which has a multiplexed address/data bus. Because the Z80A does not have a multiplexed bus, it was necessary to design the logic to do the multiplexing. By using software to do the multiplexing, we saved printed circuit board area. Two Z80A instructions do the job. One sends the address to the translator chip via the data bus, and the second instruction sends or receives data to or from the translator chip. It takes seven gates, about one and one-half LSTTL IC packages, to implement this function.

**Low-power Schottky transistor-transistor logic.

The RAM control logic is more complex than the other logic support area. This logic generates four signals that control the RAM: row address strobe (RAS), switch address multiplexers (SMUX), column address strobe (CAS), and read/write (WRITE). All these signals are negative true.

A memory access to RAM consists of the following sequence. The Z80A presents the desired address on the bus and signals a memory request (MREQ). The WRITE signal goes high or low depending on whether the access is a read or a write. RAS becomes true and the RAMs latch in the eight row address bits. The next edge of the clock switches the address multiplexers. CAS becomes true one-half clock cycle later and strobes the column address bits into the RAM. After a finite delay the RAM has either read or written the data and is ready for the next memory cycle.

From a hardware viewpoint here is what happens at power-up. The ROM is the selected memory device and the Z80A executes its instructions from ROM. The Z80A does some housekeeping tasks (RAM self-test, load boot code from ROM to RAM, et cetera) and then asks the mainframe to load the CP/M system from the disc into the module's memory. While the module is busy at power-up, the mainframe (HP-86 or HP-87XM) performs its own self-test and then loads the autostart program on the CP/M disc, which in turn loads and executes the binary program that services the I/O requests of the 82900A. The request to load the CP/M operating system is granted and the standard CP/M prompt (A>) is displayed. At this point CP/M is operative and will execute any valid command.

CP/M System

CP/M is an environment in which programs can be run and files can be accessed. It provides a shell so that any standard CP/M program can be run on any machine that can run the CP/M operating system. The operating system itself is not very complex but is an important link. Without it, much of the application software commercially available for small computers would not be executable on the Series 80 machines.

CP/M is built of three basic parts. The console command processor, or CCP, interfaces to the user and interprets the CP/M commands the user gives it. The basic disc operating system, or BDOS, handles file management operations and allows application programs to access the wide range of system functions available by using simple subroutine calls like home disc, read file, et cetera. The basic input/output system, or BIOS, implements the low-level I/O functions such as get character and put character. Of these three pieces, only the BIOS needs to be rewritten for use on a new computer system. This is one of the reasons CP/M is such a popular system. It is relatively simple to get it running on any computer with an 8080, Z80, or 8085 microprocessor.

Some other reasons for its popularity are the number of good application programs available for the operating system. CP/M has been around a long time and a lot of programmers have had time to experiment with writing code that runs under it. Because it is a software development environment, it is naturally attractive to programmers. CP/M also has a large installed base and is available on many different manufacturers' hardware. This means a big market for anything that a programmer writes.

The Series 80 implementation of CP/M is a standard, no-frills version of CP/M. That was the design goal and what is provided. However, the software provides for later additions to the system. Currently, things like graphics and HP-IB* function calls are not available because these would make the CP/M for the Series 80 computers nonstandard.

Software

The BIOS code implements a software interface in two directions. When communicating with BDOS it must recognize all the standard BIOS calls. To handle the mainframe interface, a protocol suitable for this application was invented. In most CP/M systems, the BIOS actually does the low-level I/O driving of peripherals. In the 82900A, BIOS sends commands to the mainframe where the low-level drivers reside in the binary executive and the system ROMs of the HP-86 and HP-87XM.

The binary executive makes the mainframe a slave to the CP/M system. It interprets commands sent to it from the BIOS and causes the requests to be executed by the Series 80 peripherals. Part of the problem in coding this binary program is that the HP-86/87XM operating system is already a complex piece of software. It is necessary to operate at a lower level than the available routines to achieve the desired results. A good example of this is the possibility of simultaneous interrupts from multiple devices. Since it is not necessary to handle this situation under CP/M, no software for it is included in the binary executive. This speeds up CP/M operations. In addition, the binary executive insulates the CP/M system from the Series 80 peripherals, thus off-loading some processing. An example of this is key mapping. To the normal Series 80 user, an escape key is unnecessary, but CP/M applications do occasionally require an escape. The binary executive maps the **TR/NORM** key on the HP-86 and HP-87XM so that to CP/M, it appears as an

*Hewlett-Packard Interface Bus, HP's implementation of IEEE Standard 488 (1978).

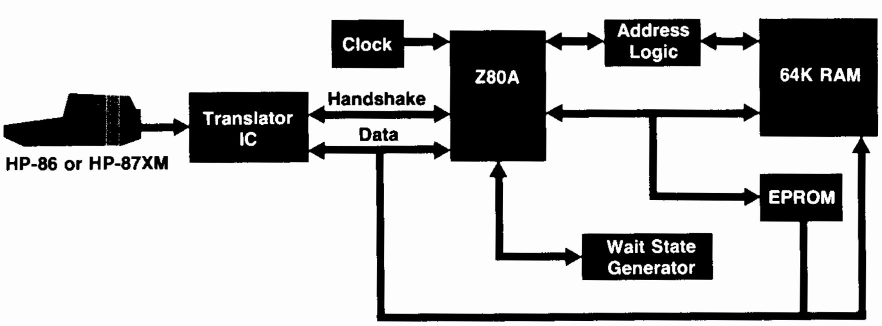


Fig. 3. *Block diagram of electronics for the 82900A CP/M System Module.*

ESCape key. Another nice feature is that the HP-86 and HP-87XM can have more than 400K bytes of memory (less than the maximum possible because the CP/M system module occupies one of the four mainframe ports). The binary executive for CP/M occupies 4K bytes of the mainframe RAM. Although CP/M cannot access the rest of the mainframe memory directly, the binary executive makes it look like a printer so that files can be printed from this memory buffer at the same time that other CP/M activities are taking place. Complete control is provided over this print buffer through the use of another key on the HP-86 and HP-87XM that CP/M does not use. Pressing the **PAUSE** key toggles the printer on and off and pressing **STEP** (**SHIFT-PAUSE**) deletes the print buffer completely. All of this occurs without affecting CP/M operation.

The third piece of software, the boot ROM, is simple in function, but vital for initializing the system properly. The code transfers itself into high RAM and executes a test of low RAM. The boot code is then placed in its proper location and tests the remainder of the RAM. After determining that the 82900A Module is functional, the code reports to the Series 80 mainframe and requests that the CP/M system be loaded. Once the system is loaded, the boot code turns execution over to CP/M and disappears.

A memory map of the 82900A RAM is shown in Fig. 4.

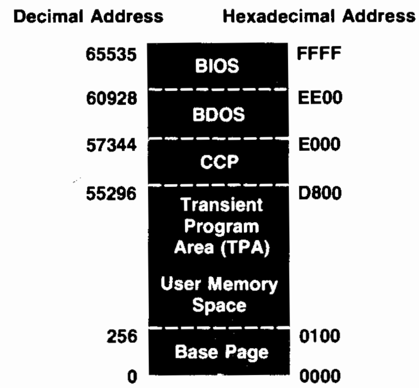


Fig. 4. *Memory map for 82900A system.*

Applications

The real power of the CP/M operating system is the number of application programs available. These applications run the complete range of microcomputer software. In many cases a new piece of software is written for CP/M before being coded for other systems.

The HP-PLUS software program aids the user in finding and purchasing the HP-disc-formatted CP/M software necessary to make your HP-86 or HP-87XM Computer into a powerful new machine.

One of the most comforting things about CP/M is that it offers every major language for computers. So, no matter what language you are used to programming in (Pascal, C, COBOL, FORTRAN, all versions of BASIC, Forth, or assembly), all are available under the CP/M umbrella.

Word-processing software such as Wordstar™, and electronic spreadsheets such as SuperCalc™ will run under CP/M. Accounting software for a number of business applications is also available.

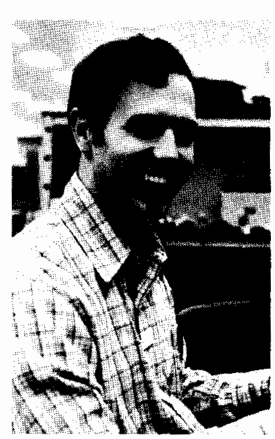
A novel application that could be quite useful is a PERT (program evaluation and review technique) chart generation program called Milestone™. With a friendly user interface, it asks you questions about how tasks are scheduled and arranged. It then generates a PERT chart printout showing when tasks should be performed on a project to meet its scheduled completion date.

One interesting application we have discovered allows the engineer to design a project on a microcomputer. Dasoft™ (design automation software) provides a library of ICs from which the engineer calls up pieces and specifies the signal names that correspond to the pins on the device. The designer then specifies device placement on the schematic and printed circuit board. The computer and its software take over and draw the schematic, generate sorted network and material lists, generate a wire-wrap list that includes information on wire length, autoroute 80% of the printed circuit board, allow editing of the board layout, and generate artwork for the printed circuit board manufacturer.

Wordstar is a trademark of Micropro International Corporation.
SuperCalc is a registered trademark of Sorcim.

Acknowledgments

Jay Phillips was responsible for the boot ROM and BIOS software and was a key person in getting the 82900A done on time. Ching Chao-Liu did the binary executive.



Timothy V. Harper

Tim Harper joined HP in 1980 after earning a BSEE degree in computer design at Brigham Young University. He has worked in both R&D and production engineering on Series 80 plug-ins such as the 82940A GPIO and 82929A programmable ROM modules. Tim was the project leader for the CP/M module and now is a project leader for new system electronics design. He is a member of the IEEE and was born in Medford, Oregon. Tim is married, has two children, and lives in Corvallis, Oregon. Besides being active in his church, his greatest enjoyment is spending time with his family.