# Articles

# The Next Generation of Personal Computers

JOHN P. CRECINE

**Surprisingly affordable workstations with powerful graphics and computational capabilities will be on the desks of students and professionals within the next 2 years. Leading computer manufacturers and universities are creating a UNIX-based systems software regime that allows for portable applications software that can run on a wide range of workstations and that exploits emerging technologies.**

WITHIN 2 YEARS THE NEXT GENERATION OF PERSONAL computer workstations will emerge. These computers will cost no more than fully-configured versions of current microcomputers, but they will be 5 to 10 times more powerful, with 10 to 20 times as much active memory and with the graphics capabilities previously available only on costly, specialized systems.

The next generation of personal computers will provide the most advanced professional design aids, document and graphics design tools, and knowledge-based systems (artificial intelligence programs, expert systems, and intelligent tutors) for $5000 to $6000. This includes the raw computational power that comes with a computer able to execute about 3 million instructions per second (MIPS) and with 2 million or more bytes (a word or portion of a word) available in active memory and another 30 or 40 million bytes stored more permanently on an attached hard disk (*1*). Thanks to a feature called "virtual memory," software developers need not worry about shoehorning sophisticated programs into the particular memory constraints of a given personal computer, and users will have access to programs and databases limited in practice only by the size of their hard disks.

With large screen displays, the next generation of workstations will permit a person to view simultaneously a full page of text, the outline of the paper, and a planning guide or flowchart. All of these will be visible with a resolution rivaling that of a printed page. The graphically oriented user interface will make it possible to use one's intuition in learning new programs: it will use menus and a "mouse" (or another pointing device) to select from a list of commands instead of arcane sets of keystrokes. The array of roughly 1 million dots (called pixels for "picture elements") that makes up the computer's screen permits the use of sophisticated animation to depict simulations of science lab experiments or architectural designs.

Often, one or more scarce information or hardware resources are shared by a number of workstations networked together. For many users, communication capabilities are more important than computation. "Diskless workstations" on a local area network (LAN) can use a common, shared external storage device called a "file server" and access it over the network, rather than have an external storage device or disk dedicated to each workstation. Similarly, small LAN's of workstations with limited computational capability but sophisticated graphics displays might share the computation cycles of a much larger central processor elsewhere on a network of linked LAN's. Access to network services will be important in realizing the full potential of the new workstations (see Jennings *et al.*, p. 943).

Given their likely price in late 1986 or early 1987—roughly $6000 but closer to $3000 with educational or quantity discounts—the new workstations can serve as a vehicle for revolutionizing education (with higher education leading the way) as well as for providing the powerful professional tools needed in engineering, science, and the world of design and commerce. With an overall size about that of a 17-inch portable TV and drawing less power than a 150-W light bulb, these workstations, while not portable, will be small enough to find their way into dormitory rooms, offices, and laboratories.

The high-capacity, low-cost nature of the next generation of personal workstations stems from the convergence of several factors:

1) A coherent vision of what the graphics workstation and computing environment of the future should be like.
2) Hardware development (involving computer processor units, memory and storage devices, and input-output devices) that has led to greater capability at lower cost.
3) The development of software and expertise necessary to exploit both the computing power and display capabilities of the emerging workstation and computing environments.

The key requirements for realizing the potential offered by the next generation of workstations are software portability, hardware independence, and operating system compatibility, all of which make it possible for applications programs developed on one workstation to run on another.

## A Shared Vision for a Workstation

A major revolution in computing began to take shape more than a decade ago with the introduction of powerful computers, dedicated for use by a single individual, and equipped with important new features such as high-resolution graphics displays. The first concrete manifestations of these technologies were created at the Xerox Corporation's Palo Alto Research Center (PARC) in the form of a prototype, the Alto personal computer environment. The greater PARC achievement was the truly revolutionary operating system developed for the Alto, built around a graphically oriented, desk-top metaphor with icons representing files, documents, and programs, and a pointing device in the form of a "mouse" for selecting programs, documents, and complex commands from menus of choices. The notion of a CRT (cathode-ray tube) display as a window into a much larger document or database and workstations linked together on a LAN for communications and resource sharing were also PARC innovations. PARC correctly anticipated the major

John P. Crecine is senior vice president for academic affairs, Carnegie-Mellon University, Pittsburgh, PA 15213.

advances in technology and the increased functionality and dramatically lowered prices associated with those advances, and designed a personal computer system—hardware and software—that exploited the chosen technology.

The PARC ideas spread, slowly at first, to a few major computer science departments in leading research universities and to engineering and design labs in major commercial enterprises. The early adopters of PARC ideas were not price-sensitive.

In 1979, when the Apple II was a little over a year old and regarded as a toy by computer professionals who were generally wedded to mainframes, the computer science department at Carnegie-Mellon University predicted that with ". . . the level of capital investment which today (1979) provides each user with a small slice of a time-shared machine and a crude CRT terminal will, by the mid-1980's, provide that same user with his own powerful machine, far more powerful than today's microprocessors and equipped with such features as high-resolution color graphics. . ." (*2*). The technological forecasts inherent in the 1979 Carnegie-Mellon computer science plan are remarkably similar to the workstation hardware that is emerging, 6 years later (Table 1). Prototype systems based on this forecast, together with similar systems developed at other institutions, have been extremely influential in shaping the move toward distributed, personal computing systems in higher education.

Although Xerox made an early attempt to capitalize commercially on PARC developments in the form of the Xerox Star, no Xerox product was itself a sufficient commercial success for PARC to have a direct impact on personal computers. That was left to Steve Jobs, then with Apple Computer Inc., who recognized a good thing when he saw it while touring PARC facilities and quickly set out to bring PARC-like technology to the general public at a low price. Jobs and Apple brought out the Lisa in 1983 and the much lower-priced Macintosh in 1984, using the new technology in much the same ways as PARC. Although the original Macintosh had too few resources in the form of memory and screen size to exploit fully the technology it embodied, it played an important role by exposing a large number of users and developers to new technology and software concepts at a relatively early date.

The IBM Personal Computer (IBM PC) introduced in 1982, expanded the personal computer market to business and administrative uses with a mainframe-like character display and operating system (MS-DOS), added legitimacy and stability to the fledgling industry, and greatly expanded the potential user base. Gradually, IBM PC systems and clones have begun adopting PARC-like technology with a mouse and graphically oriented user interfaces like Windows, GEM, TopView, and Desqview.

The importance of a shared vision of a workstation is the way in which it helps organize technological development. A vision provides a sense of which potential software and hardware developments are likely to contribute most to overall system performance and which are consistent with or necessary to it. For example, a user-friendly interface implies high-resolution graphics, and multitasking implies sharing information and networks. The PARC model of a workstation provided a grand strategy for organizing research priorities in both the personal computer industry and the university-based computer science research labs, and the merger of these two development streams is resulting in next generation workstations.

## Technology Transition and Program Compatibility

The workstations made possible by continuing technological advances represent both an opportunity and a problem for engineers, scientists, researchers, educators, and students. Exploiting the new technology all too often requires a radical change in work habits. New systems must be learned, and existing programs and functions must be converted if they are to work in a new computing environment. When is the benefit of jumping to a new technology or new system worth the cost in time and resources of leaving existing practice behind? Is there a graceful transition strategy that allows one to bring along existing functions while acquiring the new capabilities made possible by new technology? The insidious aspect of the problem is that technology never stays still. The one safe prediction is that the next generation of workstations, as wonderful and powerful as they are turning out to be, will be succeeded by even more wonderful and more powerful workstations in a year or two. What is needed is a long-term strategy for allowing software developers and computer users to migrate to successive generations of ever more powerful computers, and to do so gracefully.

The technology transition problem is more general than the problem of when and how to adapt to technological progress. A close cousin is the problem of dealing with diversity in hardware, software, and underlying technology at any given time. A strategy

Table 1. Technical characteristics of personal workstations.

| Workstation type | Processor speed (clock rate) | Virtual address space | I/O bus (bits) | Primary memory | Secondary storage | CRT display | Suggested retail price at introduction |
|---|---|---|---|---|---|---|---|
| IBM PC (1981) | 0.4–0.75 MIPS (4.7 MHz) | $2^{16}$ | 8 | 64 kilobytes | 356–712 kilobytes, floppy disk | 12 inches monochrome, 24 × 80 characters | $5,000 (1983); $2,200 (1986) |
| IBM PC/AT (1984) | 0.75–1.50 MIPS (6 MHz) | $2^{24}$ | 16 | 256 kilobytes | 20 megabytes | | $5,000 (1985) |
| Apple Macintosh (1984) | 1–2 MIPS (7.8 MHz) | $2^{32}$ | 16 | 192 kiloytes 576 kilobytes (1985) | 400 kilobytes, floppy disk | 9 inches monochrome, 512 × 342 pixels | $2,400 (1984); $1,700 (512 kilobytes, 1985) |
| Apple Macintosh Plus (1986) | 1–2 MIPS (7.8 MHz) | $2^{32}$ | 16 | 1.1 megabytes | 800 kilobytes, floppy disk | 512 × 342 pixels | $2,300 (1986) |
| 1979 Technology prediction (*2*) | 1.0 MIPS | $2^{30}$–$2^{32}$ | 32 | 1 megabyte (minimum) | 100 megabytes, hard disk | 1000 × 1000 pixels, color | $10,000 |
| Next generation workstation (1986–87) | 2–4 MIPS (12–16 MHz) | $2^{32}$ | 32 | 2 megabytes (minimum) | 30–40 megabytes, hard disk | 1000 × 800+ pixels, monochrome | $5,000–$6,000 |

for technology transition should also be a strategy for dealing with diversity in a distributed personal computing environment. Many of the problems of coping with diversity were hidden from view in a world dominated by a small number of expensive mainframe computers that all users in a community were forced to share. In a world where individual choice is both possible and desired, any user of personal computing who exists in an environment where communication and a sharing of software resources is important must cope with the problem of creating coherence in a world with many computer vendors and many workstations available from a given vendor.

Common technology and technological trends have not led to compatible microcomputer systems and portable applications software. Instead, the tendency of the larger firms in the computer industry has been to develop proprietary systems in order to differentiate one company's product from another's. This approach masks the underlying similarities of the hardware in an attempt to permanently bind customers to a company's product line. The positive aspect of such tendencies is that they fuel investment by computer manufacturers in development of new technologies, making it possible for manufacturers to exploit a (brief) technological advantage. The negative aspect is the fragmentation of the market for microcomputing.

Software incompatibility and fragmentation create serious problems. Consider the case of universities as a user community that wishes to use the emerging computing technology to improve teaching and research operations. Because of their size and diversity, research universities are representative of many other user communities and large organizations. To have a significant impact on higher education, software must be nearly as portable as textbooks. Now, without considerable extra investment or a comprehensive strategy for software technology transfer, a typical piece of educational software will run on only one machine out of the four or five personal computers available. An already-divided market for software in, say, fluid mechanics or 16th-century French history, is divided still further by four or five. This is not a formula for generating the necessary variety and quality of software to support educational computing, let alone a revolution in education. One institution's fate is tied to that of other institutions of higher education. No university uses only locally authored textbooks in its classrooms, and no university can long rely on only locally produced software customized to a particular model of computer, from a particular manufacturer. No single institution can expect to go it alone if it wants a sufficient array of computer software to truly change the way education is delivered. A similar case could be made for research or administrative applications in universities. Each university has a very clear stake in making applications software independent of any particular manufacturer or machine. The problem is no different for nonacademic institutions or for professionals who need access to a body of specialized applications software.

The problem is straightforward, even if the solution has not been. What is needed is a systems software base (an operating system) that is stable and that "floats over" the volatile hardware and technology that represent the world of microcomputing. Needed is a stable base that applications software developers can build on without being forced to target their efforts at only one of the IBM PC's, PC clones, Apple Macintoshes, or VAX-based machines of today. Needed is a stable systems software base that provides for easy extensions to accommodate the high-resolution displays of today, and the bit-mapped, high-resolution color displays and optical-laser storage disks of tomorrow.

If software developers cannot build on a stable base, if they cannot write programs with a half-life longer than that of current microcomputer technology, if they have to change software with every new wrinkle in technology and every new manufacturer or product, if applications software cannot be developed independently of the particular hardware it runs on, then the potential represented by the next generation of workstations will not be realized.

Creating this stable systems software base and machine independence has been a principal task of Carnegie-Mellon University's Information Technology Center (*3*) and Project Athena at Massachusetts Institute of Technology (MIT) (*4*), with important contributions from Brown University. It has also been one of the principal objectives of a larger group of cooperating institutions including the University of California at Berkeley, City University of New York, Columbia, Cornell, Dartmouth, the University of Michigan, RPI, Stanford, Vassar, and the University of Wisconsin, among others which form the Inter-University Consortium for Educational Computing that Carnegie-Mellon founded with support from the Carnegie Corporation (*5*).

The consortium was established to facilitate the development and sharing of educational software. It has been working closely with major computer manufacturers—AT&T, Apple, Digital Equipment, Hewlett-Packard, IBM, NEXT, SUN Microsystems, and Texas Instruments—to create the necessary conditions for software portability in the next generation of personal workstations. The necessary, stable systems software base is built on top of UNIX, a hardware-independent operating system originally created at Bell Laboratories.

Use of the emerging workstation technology to consolidate the microcomputing marketplace through the use of compatible operating systems and systems software will serve to increase dramatically the overall size of the market and of the installed base of systems. This, in turn, will stimulate greater investment in applications software and increase the variety of such software. "Software," and operational capability, "sell hardware," and therein lies an attractive growth dynamic for the market. The proprietary approach is fueled by dreams of market share. The irony, of course, is that, in the aggregate, much proprietary behavior leads to a much smaller overall market. A bigger slice of a smaller pie would be a better outcome.

## Operating Systems: A Key to Technology Transition

For a single, stand-alone computer there is a complex systems integration problem, dealt with partly by piecing together the hardware components in a coherent fashion. The operating system (OS) serves as a higher level manager or executive for the hardware system, coordinating the activities of the various hardware components, retrieving, modifying, and storing information, and translating program instructions and user inputs into action.

There are several important functions any OS is required to provide. It must make provision for the execution of several functions in parallel. For example, input-output (I/O) operations must go on at the same time other computation is taking place, and several programs are often in memory simultaneously. Switching between activities, protecting one activity from another, and coordinating the timing of more-or-less concurrent activities all must be provided for. Many resources such as memory and computing cycles must be shared among processes, devices, users, and computing sessions, so one of the prime requirements for an OS involves rules and procedures, a logic, for resource allocation. Storage of information in both active memory and secondary storage must be managed, with various kinds of protection and access rules and with a specification of archival and other file procedures. Then there is the management of the sequence of a computing task. Part of the

operation is determinate in the sense that a program run under the same conditions should do the same things. Part of the operation is indeterminant in the sense that some events (for example, user-provided input) affecting computation can occur in an unpredictable order (*6*). Finally, an OS should be efficient, reliable, and maintainable and should not consume excessive computational resources itself. And it should exploit the technology embedded in the computer system by making it easy for programmers to fully utilize the capabilities of the hardware (*7*).

The OS mediates between the user, the applications program, and the specifics of the hardware system making up a computer. All operating systems mask some of the details of the hardware from the user of applications programs, be they word processing, statistical packages, simulations, or knowledge-based programs, and from the programmer. To the degree that the features of the OS used by a programmer, such as file management, I/O features, and managing display graphics, are specific to the particular computer hardware the OS functions on, an applications program cannot be run on another computer unless it is customized for that machine-OS combination as well. Applications programs depend on features of the OS that make it possible for computer hardware to perform as directed. If the OS exists on several machines, then an applications programmer can write a program once and need only be concerned with implementing the program on different operating systems. "Hardware independence" means applications programs are written for OS's, not computer brands or models.

For user groups like the higher education community or users in any large organization, the issue of software portability is closely bound to the overall issue of the utility or worth of computing technology. The objective is to make the sharing of programs and information as easy as possible. The current generation of microcomputers features several different operating systems—MS-DOS on IBM PC's and PC clones, Macintosh, CP/M, Xenix and other forms of UNIX—and makes software portability extremely difficult. The minicomputers for which the next generation of personal workstations can substitute have other operating systems. UNIX in its various forms and VMS are the principal ones.

Market dominance in the supermicro world is highly unlikely, so a "monopoly solution" to hardware independence is also unlikely. As a practical matter, there is but one choice as the basis for a hardware-independent, universal operating environment for the next generation of personal workstations. UNIX, developed by Bell Laboratories, is the nearest thing to a hardware-independent operating system available. For hardware systems with a UNIX OS, it is relatively easy to move applications programs from one system to another. The Berkeley extensions to UNIX (embodied in 4.2 BSD

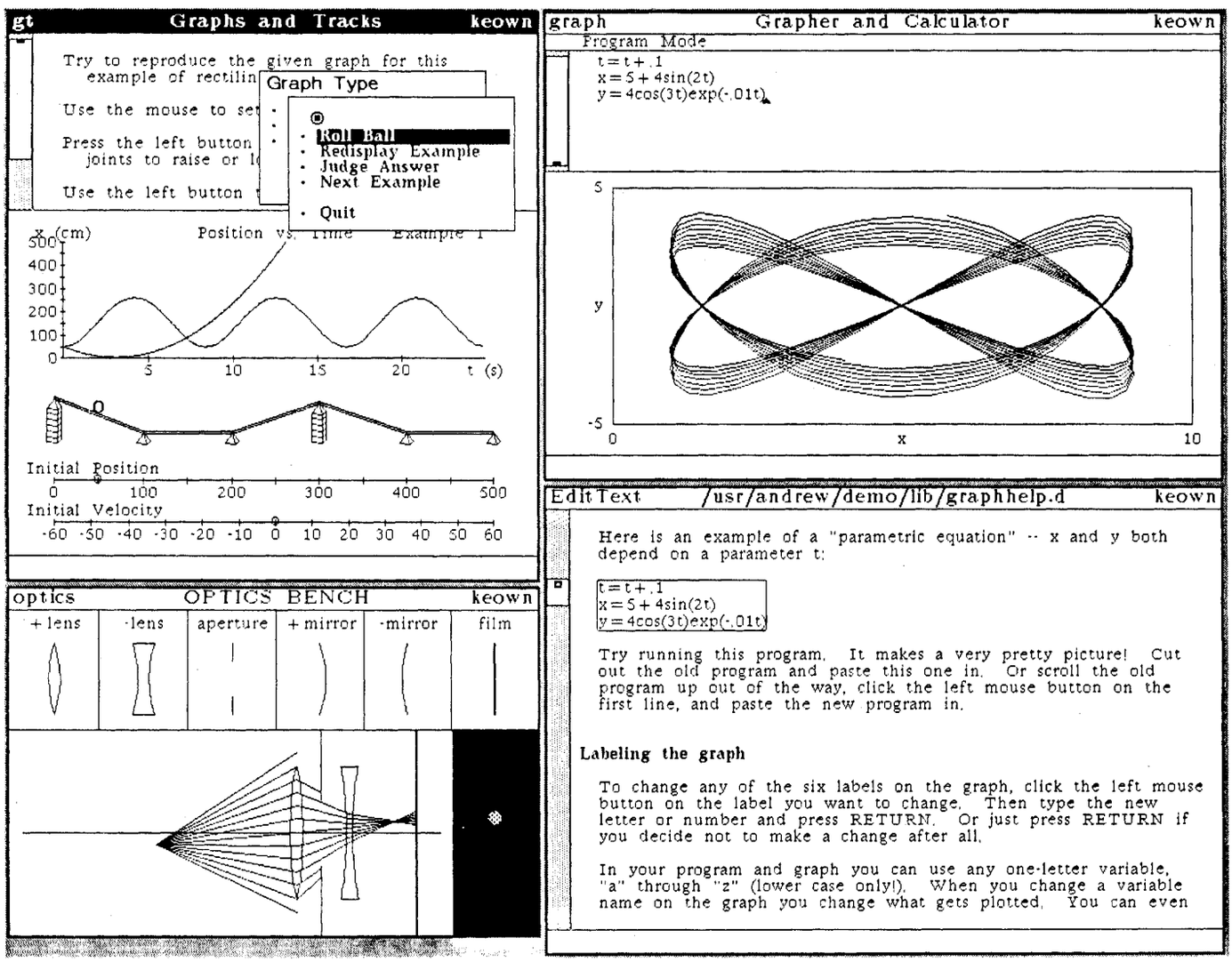


Fig. 1. Multiwindow CRT screen display; a "screen dump" from a 1024 by 1024 pixel display of four concurrent processes running under the Andrew operating environment.

UNIX and in the latest versions from Bell Laboratories of System V UNIX) support the new technology—virtual memory, graphics, and high-speed data communications—and can be found on the early examples of next generation workstations like the SUN 3/50, the VAXstation II, and the IBM PC RT. Equally important, the owner-developer of UNIX, AT&T, has taken an enlightened view by licensing the product freely and by facilitating its use on non-AT&T equipment, making it an economical and reliable choice as well as a functionally appropriate choice.

To fully exploit the new technology, some additions to the current generation of UNIX-based systems software are needed. For applications software developers, what is needed is a window manager for managing the multiple windows into multiple processes that can be displayed on a screen (see Fig. 1 for an example of several "windows" open on several processes, running concurrently and appearing simultaneously on the CRT screen), an editor capable of handling sophisticated graphics and character fonts, and programmers' utilities that facilitate use of system capabilities by applications programmers. UNIX is an OS with the power and flexibility that appeals to computer scientists and professional software developers. To most, the UNIX command language appears arcane and complex and should be "hidden" from users with a graphics-oriented user interface with at least those features found in the Xerox PARC and Macintosh systems.

A UNIX-based, hardware-independent solution to the software portability and technology transistion problem is in view. A complete set of UNIX extensions, known collectively as "Andrew," is near completion by the Information Technology Center at Carnegie-Mellon University. MIT has developed a superior window manager, X, that is "plug compatible" with the Andrew window manager. Brown University has also developed systems software compatible with Andrew and UNIX. As yet there is no user interface running on top of UNIX that meets the PARC-Macintosh standard.

Through the efforts of the consortium and major computer manufacturers, the possibility exists for a common systems software base for the next and succeeding generations of personal workstations in higher education. If this development is successful, the benefits in functionality and applications software portability will be available to all.

A common systems software base will greatly increase the impact of the technologies brought together in the next generation of microcomputers. Understanding the technologies that underlie the hardware components of microcomputers is important to understanding the nature of the potential impact.

## Workstation Hardware: Converging Technology

Shaped partly by the price-performance trends associated with the microelectronic components of a modern microcomputer system and partly by the growing needs of the large, installed base of current generation personal computers, the characteristics of the next generation of personal workstations are fairly clear (see Table 1). The convergence of several computer manufacturers' product lines in the neighborhood of the price and performance relation shown in the last column of Table 1, coupled with a stable systems software base appropriate to the emerging hardware technology, has created the potential for revolutionary advances in microcomputing (*8*). Some of the major technological developments involving the components of modern microcomputers will be reviewed below.

The modern workstation is a complete computing system that includes certain basic components, shown in a stylized fashion in Fig. 2. The CPU (central processing unit) manipulates information

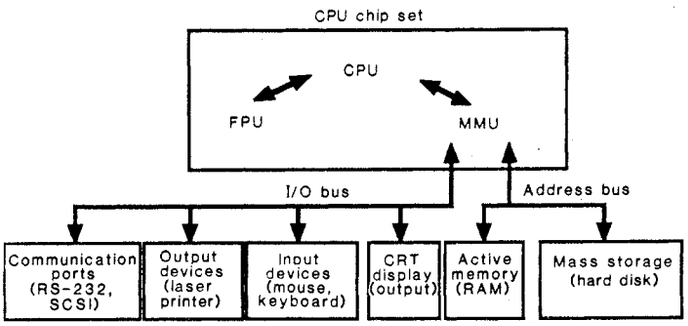


Fig. 2. One of the many ways of configuring microcomputer components. For example, the bus may be used for both I/O and address; the FPU may be included in the CPU; and the communication port may access mass storage.

and executes instructions. Instructions are used to perform operations such as multiplying two numbers together, retrieving information from or storing information in a particular location, moving information between locations, or changing the sequence in which a set of instructions is executed. Different CPU's have different instruction sets, the basic operations a CPU can perform in hardware. Both information and instructions are stored in a computer's memory, and each memory location has its own address or unique number. There are two basic types of memory: (i) primary or active memory and (ii) secondary memory (mass storage devices). The instructions for the CPU are located in the primary memory. Primary memory is fast, and information stored in it can be accessed directly (this is called "random-access memory" or RAM) in contrast to the sequential accessing of information stored in most secondary devices.

The computer must be able to access the large numbers of files that can be found on mass storage devices connected to it and therefore must maintain lists of files and information about their physical location. Different computers have different ways of organizing files or different file systems.

Computers have different ways of communicating with the world and different I/O devices. The most common input device is the keyboard. When a user wants to provide the computer with an instruction or command, the command is generally typed on the keyboard. Output from a computer to the user most commonly is provided through a video display or CRT or printer. Finally, most computers have the ability to communicate with other devices by means of a direct physical connection or network. A communications adaptor and special port is usually required to electrically transmit a stream of information to and from a computer. In addition there must be an agreed upon set of conventions or communication protocols to enable computing devices to exchange information.

Each major microcomputer component has its own underlying technology and, because each is part of a system, the components' technologies are interrelated, as are the components in a given system. Technical advances on one component in the system often create performance problems for other components. A computer with a million bytes (1 megabyte) of RAM on a machine with an 8-bit CPU or that (without special tricks) is capable of independently addressing only 64 kilobytes of that memory, makes little sense. A large-screen bit-mapped display (CRT), with a million separate pixels constituting the display, requires corresponding processing power and clever coding of screen data to move the million pieces of information quickly. Computer designers must be extremely sensitive to the functional interdependencies of the capacities and performance constraints of the components of a computer system. Where this becomes difficult for both designer and consumer is in

providing the opportunity for expanding the various capacities of individual components over time, making sure that adding capacity or components to a system leads to an improvement in overall system performance rather than a discovery that relaxing one performance constraint merely allows the system to butt up against another constraint in some other part of the system. For example, adding a hard disk storage device capable of rapidly accessing large amounts of information and then forcing disk-CPU communications through a slow communications port will do little to improve overall performance.

## The Central Processing Unit

Generally, CPU's are single, self-contained electronic circuit chips the size of a postage stamp. A microcomputer's CPU can be described in terms of (i) the word length or length of the string of bits it is able to process internally and communicate to other elements of the computer system, (ii) the speed at which it is able to perform basic operations, (iii) the size and complexity of its instruction set, and (iv) the way the CPU does transactions with physical memory.

The first CPU's for microcomputers were 8-bit chips, able to process information in 8-bit chunks, where a bit is a binary digit. The CPU in the Apple II, for example, handles 8-bit units of information internally and communicates with other computer components by means of electronic connections that transmit 8-bit words (an 8-bit address bus and and 8-bit I/O bus). The leading personal workstation of the current generation, the IBM PC, has a 16-bit processor to process information internally, and it communicates with other computer components through an 8-bit address bus.

The next generation of workstations will have 32-bit processors, with 32-bit data and address registers (temporary storage locations on the CPU), linked to other components with a 32-bit address or I/O bus. The word length that a computer is capable of processing is important for two reasons: it determines the number of significant digits in standard numerical calculations and it determines the number of different memory locations that can be accessed. A 32-bit machine can carry more significant digits while doing arithmetic calculations and a 32-bit CPU can, theoretically, directly address 4 billion bytes (4 gigabytes), whereas a 16-bit processor can only address 1 million bytes and an 8-bit processor, only 64 thousand. In terms of word length and width of the data path in I/O and address buses, bigger is, more than proportionally, better.

Another important characteristic of a CPU is clock speed, or the rate at which basic computer operations are performed. In most 8-bit processors, the clock clunks along at about 1 to 2 megahertz (MHz). The 16-bit Intel 8086/8088 processor in the IBM PC moves along more briskly at 4.7 MHz, whereas the Motorola 68000 chip in the Macintosh and the Intel 80286 in the PC/AT dash along at about 6 MHz. The next generation of personal workstations will zing through operations at 12 to 18 MHz. Clock speed is only one of the features of a CPU; the number of instructions that a CPU can process per second is another overall performance measure, but it depends on the type and complexity of the instructions performed, the word length of the address or data, the number of registers, and the bandwidth of the bus or buses that connect the subelements of the CPU. The most common measure of processing power is MIPS (millions of instructions per second). The Motorola 68000 with one of the larger instruction sets is rated at 1 to 2 MIPS, whereas the simpler instruction set on the Intel 8086/8088 is rated somewhere between 0.4 and 0.75 MIPS. The next generation of workstations will have processors in the 4- to 6-MIPS range. For number-crunching operations common to the many science and engineering applications needing accurate floating point arithmetic operations, the number of floating-point operations per second, FLOPS, is a better measure of processing power. Most versions of the next generation of workstations will have, at least as an option, a special floating-point unit (FPU) or chip optimized for performing floating-point arithmetic operations at high speed.

Processes that must function in real time—animation or sound—place greater demands on processor power and speed and on memory than more conventional calculations simply because delays destroy functionality and the manipulation of million-pixel displays chews up memory and computing cycles. Extra processing speed is crucial if one is to fully exploit the new technology available in high-resolution, bit-mapped displays, larger programs, and sophisticated operating systems and user interfaces.

Architectural features can also be important. As CPU's have evolved, chip designers have often taken the instruction set contained on the previous generation of a chip and simply added new instructions on top. The most notable example is the Intel chip family, which began as a 4-bit chip, the 4004, in 1972, evolved into the 8-bit 8080 and 8085 in the late 1970's, the 16-bit 8086/8088 (with an 8-bit bus) on the IBM PC, the 80286 (with a 16-bit bus) on the IBM PC/AT, and, finally, the recently announced 32-bit 80386. At each stage of development, new instructions were added to the existing instruction set so that the number of elementary, chip-based instructions grew from about 30 to 91 on the IBM PC's 8086/8088 to more than 300 on the 80386. There are many advantages to this evolutionary approach to chip design, the major one being upward compatibility: because the instruction set of the latest chip includes the previous chips' instructions, any software written at the machine language–assembly language level runs on the latest chip in the family. The disadvantages are that chip designers can anticipate only a fraction of future developments and sooner or later the chip architecture under an evolutionary regime gets too complex.

At some level, a CPU must coordinate and manage all of the elements on the CPU chip itself, including the more than 300 elements of the instruction set on many 32-bit chips. By reducing the size of the instruction set, coordination overhead could be reduced and overall performance might be improved. It is also true that larger instruction sets occupy more real estate on a silicon chip, which in turn means less room for other components. The idea of a reduced instruction set computer CPU (a RISC CPU) that did fewer things in hardware, but did them much faster, originated with David Patterson and a series of student projects at the University of California, Berkeley (*9, 10*). The first nonproprietary 32-bit RISC CPU available, the Fairchild CLIPPER, has reduced the instruction set to about 120 instructions. Early experimentation with RISC machines suggests that gains in speed for general-purpose machines are not dramatic, that floating-point calculations may be performed better on conventional 32-bit chips, and that major performance improvements over conventional chip architectures occur when a RISC chip is customized for a particular function or compiler language (*10, 11*). It is likely that RISC-based 32-bit microcomputers will be present on the next generation of workstations.

## Virtual Memory and Memory Management

An extremely important feature of the CPU's for the next generation of workstations is virtual memory. Stated most simply, virtual memory allows a program to treat all physical memory—RAM and secondary storage—as if it were one large area of RAM. Basically, memory from secondary storage devices (for example, a

hard disk) is swapped in and out of active memory, so that when the CPU needs particular data or instructions from memory, secondary or active, it is most likely to be available in RAM and rapidly accessible. The complex swapping or memory management activities take place in hardware [the memory management unit (MMU)] and are of no concern to applications program developers or users. Virtual memory means that programs and databases are limited in size only by the amount of secondary storage available and that software developers can focus on things other than complex memory management tasks and the squeezing of large programs into (relatively) small RAM. Along with large-screen bit-mapped displays and fast CPU's, virtual memory is perhaps the most important new technology available for the next generation of workstations. All CPU chip sets will have virtual memory or an MMU.

There are other features of processors that are important. Some are physical and relate primarily to the electrical and mechanical properties of the materials that make up the semiconductor electronic circuits and the chips; for example, silicon, CMOS (*12*), various germanium-silicon alloys, and other approaches to creating stable, thinner, more tightly packed chips with superior electrical conductivity (*13*). Here, smaller is faster and better.

A final point worth making about CPU chips concerns price. Most of the unit cost of CPU's is attributable to design and development. If written off over a large number of units, the unit cost of chips becomes very inexpensive. Prices of CPU chip sets to computer manufacturers (*14*) depend on quantities purchased and the way in which design and development costs are amortized. A reasonable prediction is that the price of CPU chip sets for the next generation of workstations will fall from their current $300- to $1000-range to $100 to $300 in the next 12 to 18 months. The most widely used 32-bit chips today are the Motorola 68000/68010/68020 family; the MC68020 will undoubtedly be present on several next generation workstations. Three other nonproprietary CPU chips are candidates for the new workstations as well, the National Semiconductor NS32332, the Intel 80386, and the Fairchild CLIPPER. Proprietary CPU's, such as Digital Equipment's VAX-architectured chip on the VAX station II and the various RISC machines expected from Hewlett-Packard and IBM, will also form the basis for next generation machines.

## Primary Memory or RAM

Primary memory, those storage locations that can be directly addressed by a CPU, continues to be subject to technological improvements even more impressive than those affecting CPU chips. The pieces of silicon and other physical devices that constitute RAM for microprocessors are becoming denser and faster, with more capacity. In 1968, the state of the art was a 16-bit chip for $16. In 1980, the state of the art, a 4,000-bit chip, cost $4. Today, the microcomputer industry is in the midst of a change from 64,000-bit RAM chips to 256,000-bit RAM chips. In the fall of 1984, 256,000-bit RAM chips cost manufacturers $23 to $26 each, and were obtainable only in relatively small quantities. By the fall of 1985 they were widely available, more reliable, and cost less than $3 each. One-megabit chips are just coming on the market; it is only a matter of time until they also drop in price.

One can anticipate several technological developments affecting the RAM component of the next generation of personal workstations (*13*). All of the foreseeable technological developments promise to further increase the memory capacity of a RAM chip, improve the speed of access of the memory, reduce the size and power consumption of the chip, improve reliability, and decrease the unit price.

## Secondary Memory and Mass Storage

Regardless of how inexpensive RAM chips become, there will always be a need for nonvolatile information storage such that information is not lost when the power is turned off. The early personal computers used simple, magnetic tape, similar to audio tape, to record computer programs and data for later use. In one form or another, the current and most likely future technologies for creating mass, "permanent" storage represent simple improvements on the original magnetic tape technology. Information is stored as a sequence of magnetized domains on a physical medium. Access time for finding a particular storage location is slower than with RAM because mechanical positioning is slower than electronic addressing. The smaller the spacing of the elemental units or storage locations, the more rapid the transfer and, of course, the greater the storage capacity of a particular device. Improvements, such as variable-speed, floppy-disk drives that allow tighter spacing of data on the outer rings of the disk, greater density of disks, and the possibility of lining up the elemental magnetic domains vertically, closer to each other, rather than horizontally, increase the density and storage capacity and usually decrease the access times (*15*).

The reliability of secondary or mass storage devices is inherently less than that of RAM because they are mechanical devices that whirl and spin and have recording heads that move in and out across the rotating surface of the disk. Nearly every major computer manufacturer has had reliability problems at one time or another with floppy- or hard-disk components. The search for greater reliability seems concentrated on having fewer moving parts and keeping the head far enough away from the disk to avoid physical contact, so-called "head crashes."

The optical laser disk promises even more than the magnetic technologies, however. The principle is the same—bits of information stored in a fixed sequence and organized into segments or tracks on a rotating disk. Because the bits of information are detected by a laser beam, far greater storage densities can be obtained and, because the laser beam is smaller and can be aimed with more precision from a greater distance, the laser source can be located further away from the optical disk, thus eliminating the catastrophic "head crashes" all too common with magnetic technologies. By comparison, a 40-megabyte hard disk can be replaced today with a 400-megabyte optical disk of about the same size and cost. The laser or optical disks, similar to audio compact disks (CD's), are removable but currently are read-only memories (hence their name, CD-ROM's).

Given the external storage capacities possible with CD technology, there is great interest in developing read-write CD technology. Prototypes exist, but none are yet economical enough for the marketplace. In the next 2 to 3 years, it is likely that CD technology will be used in connection with personal workstations on write-once-read-many (WORM) storage devices. The principle is fairly straightforward. A substance is spread over a disk and covered with a thin, opaque media. In write mode, the laser beam passes through the opaque covering and literally "burns" a spot on the recording substance that cannot be erased. In read mode, the same laser beam, operating with about 20 percent of the power used in write mode, will detect the presence or absence of "burned spots" on the substance (*16*). The bad news about a CD's tremendous storage capacity is that it is no trivial task to organize 400 million pieces of information in a way that allows quick access to any portion of it. Simple memory organization schemes and search algorithms may prove inadequate for such large memory stores.

When write-once or WORM technology becomes available at an affordable price, how will it be used? More than likely, a user would copy systems software, data files, and application packages onto the

CD first, using perhaps 20 to 40 megabytes of the disk's capacity. The remaining 360 to 380 megabytes would probably be used to write each file worked on and to keep a string of archival copies of each version of the file. A research paper or report might then appear on a 400-megabyte CD in each of the 30 to 40 generations of a 0.04-megabyte document. With 400 megabytes of storage, one can store many copies of many documents before exhausting the capacity of a single CD.

## Input and Output Devices

In many ways, developments in processor and memory technologies, though technically demanding, have been of the "more of the same" variety. Early personal computers and current mainframe and minicomputers are curious devices in many ways. Until recently, users and computers could only communicate by exchanging strings of text through keyboards and character printers and displays, a very limiting form of communication. The bit-mapped display, the mouse, and now the laser printer have suggested the potential of relaxing constraints on I/O devices and have led to a revolution in the way many people think about computing.

Two generic developments characterize I/O devices for personal computers. One is that more and more information—bits—passes into and out of the computing system through I/O devices, and the other is that I/O devices are increasingly diverse. For example, on the character displays found on computer terminals and early IBM PC's, a full CRT screen consists of 80 columns and 24 rows of characters or 1920 pieces of information. The smaller Macintosh bit-mapped CRT displays about 180,000 pixels and the next generation of workstations will have roughly 1000 by 1000 pixels or 1 million pieces of information. The information escalation is even more astounding when one considers laser printers, which have higher resolution bit-mapped displays than CRT's.

Because personal computers are dedicated to individual users and are likely to be found in specialized settings such as research labs or design studios, it is far more likely that dedicated, customized I/O devices will be produced for personal computers than for large, time-shared computers that serve a diverse user community. For example, optical scanners, able to recognize written and printed characters, are now practical alternatives to data entry through the keyboard. Hardware and software for voice recognition and speech synthesis also provide alternatives to keyboards. There are many devices that perform the same function as a mouse, that of allowing a user to point to a location on a CRT or to select an item from a list. Graphics tablets allow more precise locational positioning for drawing, and touch screens allow the user to point to a location on the screen or a touch pad. Similarly, locational data can be entered through a device worn like eyeglasses and aimed at a location on the CRT, a device currently obtainable for less than $200. Musical keyboards can substitute for character keyboards. A variety of analog devices for entering instrument readings are also available, and an array of devices can be obtained to monitor and control the various mechanical systems in buildings. The various forms of printers—matrix, character, laser, color, ink jet—create many of the same problems for computer transactions involving output that are created by the diversity of input devices. Conceptually, writing to a bit-mapped CRT display is no different from writing to a printer: a large area must be painted, dot by dot or pixel by pixel.

The result of the proliferation of I/O devices is that it is becoming harder to anticipate what provisions should be made in new computing systems for unknown, future devices. The increased complexity of I/O operations means that more and more processing and communication power must be devoted to receiving and decoding inputs and to generating complex output codes to drive increasingly sophisticated devices. Often a separate processor is dedicated to handle the transactions between the specialized I/O device and the rest of the computer system. In this way, the CPU itself does not have to be tied up for purposes of translating between "internal signals" and the particular electrical signals of the I/O device.

Because there must be extensive coordination between I/O operations and systems and applications software code, a great deal of concern and attention among computer system designers is being devoted to studying the way I/O devices interact with other elements of the system. A general approach that computer designers seem to be converging on is to establish a set of conventions that I/O devices would conform to when communicating with the computer system. All I/O devices would conform to the same conventions; the same pattern of voltages on the same input or output line would mean the same thing, independent of the I/O device.

## Modest Efficiency Gains or Revolutionary Impact?

As I argued earlier, a stable systems software base, floating over the more varied and rapidly changing hardware and technology base, is a necessity for applications software developers and users of the new technology. The Andrew and X extensions to UNIX constitute the best and only current alternative for providing a universally available, stable systems software base for exploiting the full and revolutionary potential of the next generation of personal workstations.

Whether the phenomenal computing power available from the next generation of machines forms the basis for a broad consolidation of increasingly sophisticated, easy-to-use applications software that exploits the capabilities of the new technology or whether it simply fuels a further fragmentation of the electronic marketplace will depend on developments in the systems software described above. In a very real sense, the next generation of personal workstations will consist of hardware powerful enough to consolidate the existing user communities—science and technology, design, business administration, and education. Such consolidation will increase the number of workstations, attract the scarcest resource of all, software development talent, to the task of exploiting the new technology, and accelerate the inevitable computerization of major segments of American society. From an individual user's point of view, such consolidation will make the creative efforts of a wide array of software developers available to all, not just the owners of a particular machine.

### REFERENCES AND NOTES


1. A byte usually consists of a string of 8 binary digits (value, 0 or 1) or bits, the basic memory element of a computer. In some computer systems, a byte is 6 or 7 bits. A word consists of one or more bytes and the central processing unit of a computer is characterized by the length of the word it can process—a 32-bit processor can process a word consisting of a string of 32 bits of information.
2. *Proposal for a Joint Effort in Personal Scientific Computing* (Computer Science Department, Carnegie-Mellon University, Pittsburgh, 1979).
3. The Information Technology Center is supported by IBM.
4. Project Athena is supported by Digital Equipment Corporation and IBM.
5. The Consortium includes other institutions that are broadly representative: California State University, Northridge; EDUCOM; Iona College; Howard University; Mills College; Online Computer Libraries Center, Inc.; Research Libraries Group; and Southwestern College.
6. A. M. Lister, *Fundamentals of Operating Systems* (Springer-Verlag, New York, ed. 3, 1984), pp. 7–11.
7. ———, *ibid.*, pp. 11–12.
8. Current examples of engineering workstations close to the next generation in performance are the SUN Microsystems Models 3/75 and 3/50, Digital Equipment's

VAX station II and the IBM RT PC. The UNIX PC, HP's Integral Personal Computer, the IBM PC/AT, and the Macintosh Plus are close in price but require significant performance improvements.

9. O. Serlin, *UNIX/World* **2** (No. 7), 20 (1985).
10. ———, *Sci. News* **128** (No. 9), 140 (1985).
11. J. Markoff, *Byte* **9** (No. 12), 191 (1984).
12. CMOS is the acronym for "complementary metal-oxide semiconductor."
13. J. C. Bean, *Science* **230**, 127 (1985).
14. As used here, "chip set" refers to the CPU, a dedicated FPU, and the MMU.
15. This is not always true. Moving through a particular disk segment to arrive at a particular location is faster on denser media, but other factors, such as getting to the appropriate segment on a disk, take overhead and can lead to an overall reduction in access time as total disk capacity goes up. The organization of the memory matters.
16. *High Technol.* **5** (No. 11), 35 (1985).
17. UNIX and the UNIX PC are trademarks of AT&T; TopView, the IBM PC, the IBM PC/AT, IBM PC RT, and IBM are registered trademarks of the International Business Machines Corp.; Windows, MS-DOS, and Xenix are products of the Microsoft Corporation; Macintosh is a product of Apple Computer, Inc.; Desqview is a product of Quarterdeck Office Systems; GEM is a product of Digital Research, Inc.; VAX, VMS, and VAX station II are products of Digital Equipment Corporation; CLIPPER is a registered trademark of Fairchild Industries.

# Computer Networking for Scientists

DENNIS M. JENNINGS, LAWRENCE H. LANDWEBER, IRA H. FUCHS, DAVID J. FARBER, W. RICHARDS ADRION

**Scientific research has always relied on communication for gathering and providing access to data; for exchanging information; for holding discussions, meetings, and seminars; for collaborating with widely dispersed researchers; and for disseminating results. The pace and complexity of modern research, especially collaborations of researchers in different institutions, has dramatically increased scientists' communications needs. Scientists now need immediate access to data and information, to colleagues and collaborators, and to advanced computing and information services. Furthermore, to be really useful, communication facilities must be integrated with the scientist's normal day-to-day working environment. Scientists depend on computing and communications tools and are handicapped without them.**

A SCIENTIST SHOULD BE ABLE TO USE COMPUTING AND communications tools by working at an advanced graphics workstation. Through that single window, the scientist may gain access to required computing facilities and databases and communicate with peers, colleagues, and scholars throughout the world. This combination of computing and communications is called computer networking. Computer networks provide the base that combines geographically dispersed researchers, computing resources, and information into a single integrated computer and communications environment. Unfortunately, the development of computer networks has been fragmented and incomplete. The result has been a bewildering array of different technologies and of different and incompatible networks. The scientist has been burdened with multiple access procedures, applications software interfaces, operating systems, and data formats. However, recent developments, including the National Science Foundation's new networking program NSFnet, the emerging convergence of the community-based computer networks, and the growing focus on the adoption of standard computer networking protocols should reduce this burden. Nevertheless, the promise of the convergence of computing and communications (*1*)—of computer networking—remains to be fulfilled.

D. M. Jennings is program director for networking at the National Science Foundation's Office of Advanced Scientific Computing, Washington, DC 20550. He is on leave of absence from University College, Dublin, where he is director of the University Computing Service. L. H. Landweber is professor of computer science at the University of Wisconsin, Madison 53706. I. H. Fuchs is vice president for computing and information technology at Princeton University, Princeton, NJ 08544. D. J. Farber is professor of electrical engineering and of computer science at the University of Delaware, Newark 19716. W. R. Adrion is deputy director of the Division of Computer Research at the National Science Foundation.

## NSFnet

NSFnet will probably have the most impact on science of all networking activities in the United States at this time. Being based on new and existing networks, it will provide both high-speed access to supercomputers and communication between scientists in all disciplines throughout the nation. Although initially designed for supercomputer users to gain access to supercomputers and to communicate with each other, NSFnet is expected to be a general-purpose computer communications network for the whole academic research community and associated industrial researchers.

The development of NSFnet is part of the NSF supercomputer initiative. This program resulted from the growing concern in the research community over the last few years that academic research has been severely constrained by the lack of access to advanced computing facilities. Several reports (*2–4*) highlighted the problems: (i) large computers have become an important means of making new discoveries, (ii) there is an immediate need to make supercomputers available to U.S. researchers, and (iii) computer networks are required to link researchers to supercomputers and to each other.

In response to these concerns, NSF established the Office of Advanced Scientific Computing (OASC), which immediately initiated two programs: the supercomputer centers program to provide supercomputer cycles, and the networking program to build a national supercomputer access network—NSFnet.

In 1984–85, OASC purchased supercomputer cycles from three