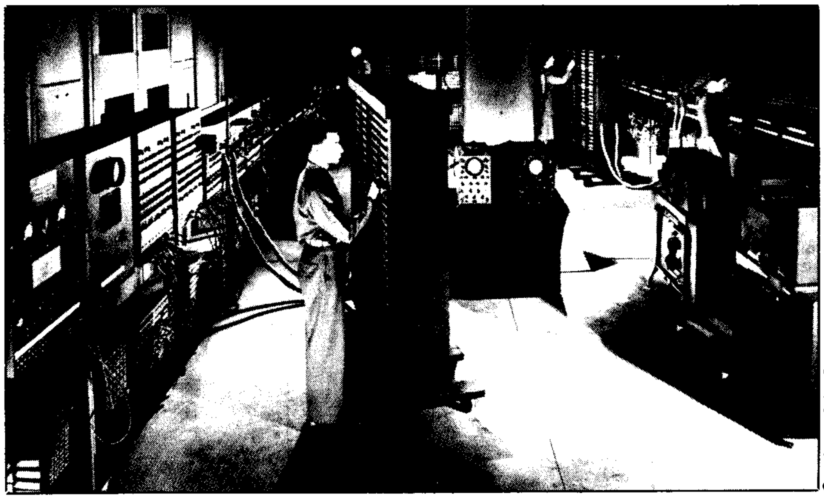
The computer revolution

E. K. Miller

A review of the continuing battle to increase our computational abilities

Man's history on earth has come to be characterized by "revolutions." So-called because they mark a dramatic and permanent turning point in the everyday lives of those who live during and after their occurrence. Until recently, the industrial revolution was in vogue. During the industrial revolution, manufacturing capabilities vastly increased due to standardization and mass production, which both substantially reduced the cost and increased the variety of manufactured goods. Now we are in the first phase of still another revolution, often referred to as the "information" revolution. The information revolution provides leverage for our mental powers in a fashion analogous to the leverage for muscle power that accompanied development of the steam engine during the industrial revolution. The "steam engine" of the information revolution is, of course, the computer. The mental leverage it gives us is at least as profound as the muscle leverage contributed by the steam engine.



Courtesy of the Computer Museum, Boston

Solving a problem on the ENIAC (shown here) required setting thousands of switches and cables.

Mainframe computer development

The electronic computer was preceded by earlier, mechanical versions. Although, Charles Babbage's Analytical Engine in the mid-19th century represents the first attempt to develop a computational power significantly beyond what could be done by hand, its mechanical design required a fabrication precision beyond the capabilities that then existed. In spite of its failure to achieve its designed goals, Babbage's work did provide a foretaste of things to come, including the mechanical tabulating machines used in the early 20th century by the United Stated government for handling census statistics. Somewhat later, a company eventually to become International Business Machines, or IBM, dedicated itself to building and servicing electro-mechanical tabulating machines for which an expanding scope of applications were being discovered.

But electro-mechanical systems were limited in speed and flexibility. So just before and during World War II the first electronic computers made their appearance. In the United States, the impetus was the need to quickly and accurately develop firing tables for the growing variety of guns used by the military. Thus it was that ENIAC (for Electronic Numerical Integrator And Computer) was proposed to the Aberdeen Proving Grounds by Atanasoff, Eckert, and Brainerd of the Moore School of Electrical Engineering, University of Pennsylvania.

ENIAC was envisioned to operate at the then unheard of speed of 100,000 "pulses" (cycles) per second. For comparison, an earlier electro-mechanical calculator designed in 1935 by Atanasoff to solve simultaneous equations by Gaussian elimination operated at 60 pulses. ENIAC was estimated to be potentially 10 times faster than an earlier differential analyzer at the Moore School and 100 times faster than a human with a mechanical calculator. Estimating that a person working on a mechanical calculator can perform one 16-digit operation per minute including the input/output time (to enter the two numbers and record the results), the throughput for the human-calculator system would be about one sixtieth FLOPS [Floating point OPeration(s) per Second]. It actually turned out that ENIAC performed a multiplication in 2.8 msec, a division in 24 msec, and an addition in 0.2 msec. So its actual FLOP rate varied from about 40 to 5000, a range that is much wider than recent computers. Thus, ENIAC operated at a rate ranging from about 240 to 300,000 times faster than the human-calculator system.

The fact that ENIAC was ac-

0278-6648/89/0005-0027$01.00 © 1989 IEEE

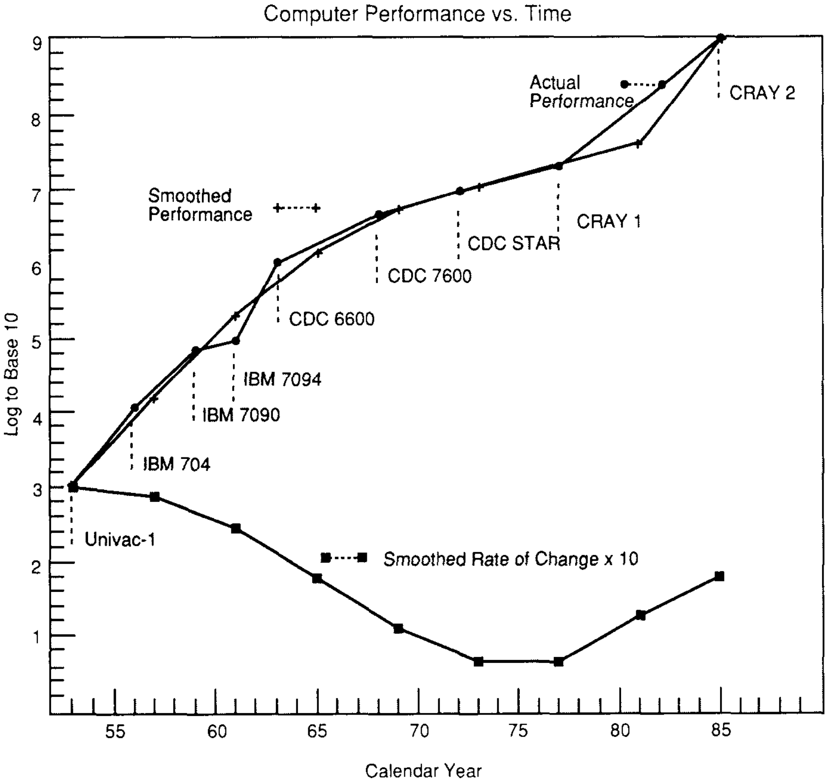


Fig. 1. Performance improvement of mainframe computers in terms of peak FLOP rate since introduction of the UNIVAC-1 computer. Shown are the actual data points for a number of computers, a smoothed curve based on these points, and a smoothed rate-of-change of performance.

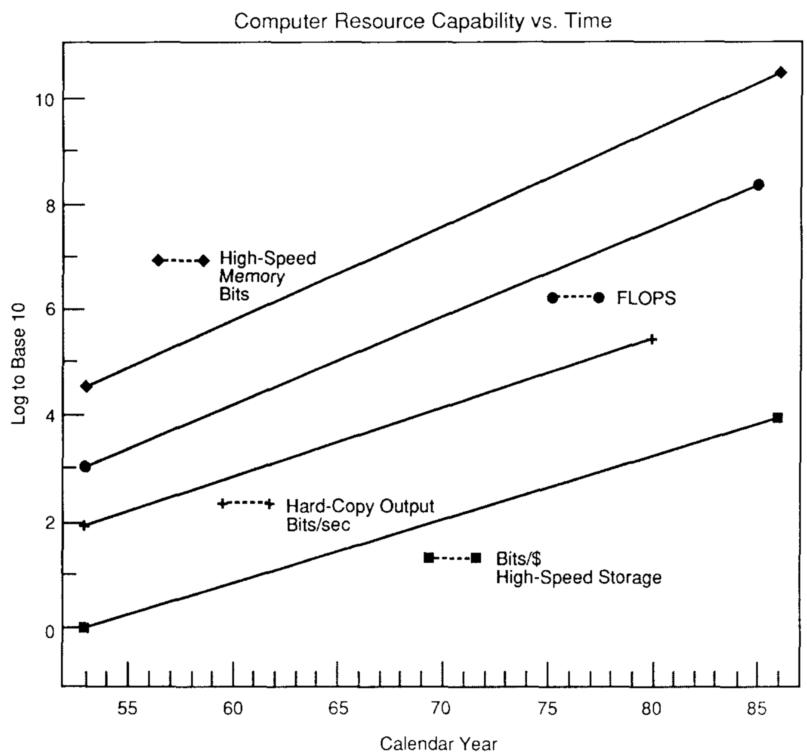


Fig. 2. Performance improvement of various computer resources since introduction of the UNIVAC-1 computer. Only end points are shown for clarity.

tually built and worked as anticipated was due certainly to the persistence and dedication of its designers and builders. For it contained 18,000 vacuum tubes, or more than 180 times the number of the largest tube-based system that preceded it. But besides personal dedication, there was also the impetus provided by military needs, certainly not the first, nor the last, example of such synergism.

ENIAC was followed by a number of other "ACs" including EDVAC (Electronic Discrete Variable Arithmetic Computer), ORDVAC, ILLIAC, SWAC and UNIVAC-1. The last achieved perhaps the greatest fame (in the United States) among these early computers. The reason was that the UNIVAC-1 was used to predict the results (correctly) of the 1952 Presidential election. For that and other reasons, it has provided a benchmark by which to measure subsequent improvements in computer technology. The UNIVAC-1 provided a peak rate of about 1,000 FLOPS (about 1/1,000,000th the speed of a CRAY-2), at a clock speed of 2.5 MHz. It contained 10,000 vacuum tubes and employed mercury delay lines as memory elements, giving it a central memory of 1,000 91-bit words. It lacked even an assembler, requiring the programmer to work at the lowest level of machine language.

Since the UNIVAC-1 was an integer-hardware machine with the floating-point operations done in software, it is somewhat inconsistent to use the term FLOPS to describe such a machine. However, defining the performance capability of a given computer by a standard process allows the user to infer the likely relative performance of other computers for similar applications.

Following introduction of the UNIVAC-1, mainframe computer power has grown at an approximately exponential rate. As illustrated in Figure 1, where the peak operating speed of a number of computers since UNIVAC-1 is plotted in terms of their year of introduction, there has occurred approximately a six order-of-magnitude improvement.

We observe that during the mid-fifties, the improvement rate in computer speed was about a factor of 100% per year. A nearly monotonic decline took place until a low point of only 25% per year improvement was reached in the mid-seventies, at which time a turnaround appears to have occurred. This conclusion should perhaps not be given too much weight since it involves only a few data points beyond the Cray 1, but the trend is certainly encouraging.

The peak-FLOP rate is only one of the various ways by which computing power might be characterized. Also important are the word length and high-speed memory size, the rate at which hardcopy output can be produced, the cost of memory, and so forth. We summarize in Figure 2 some of these characteristics over the same time span as for Figure 1. Somewhat surprisingly, the rate of improvement of these various aspects of computing power over more than 30 years are quite comparable on a log-linear scale. It is not overstating the case to observe that nowhere else in the span of recorded history have comparable changes occurred over as short a period of time.

Development of PC's

In summarizing development of the personal computer, we consider in order: the microprocessor itself, random-access storage, mass storage, and the evolution of PC software and computing performance.

Microprocessors. The past 10 years have seen a dramatic evolution in the capabilities of the PC over essentially four distinct generations. The first-generation PC's circa 1975, as exemplified by the Altair 8800 and the IMSAI 8080, were largely the domain of the technically capable electronic hobbyist. These early PC's, which were only available in kit form, had to be assembled and de-bugged by the user. (An EE degree was a definite advantage!) Input/output terminals were typically slow teletypes and program storage was usually on punched paper tape or cassette-tape recorders. Memory chips were also very expensive at the time; and they restricted the typical home computer to a few KBytes (4 to 16) of memory. The user was also largely left alone in terms of both hardware and software support, since assembly language and Basic were the only available computer languages during this time period. Very few hardware and software developers were in business.

It was not until the arrival of ready-to-run computers such as the Commodore PET, the Apple II, and the Radio Shack TRS-80 (about 1977), that PC's became available to a much wider audience. These second-generation PC's also led the way to increased convenience and user support through a burgeoning new hardware and software industry. Available memory increased significantly from typically 4K to 64K or more, and teletype output was replaced by the video screen and dot-matrix printer. Mass storage also evolved during this period from the plodding cassette tape to the speed provided by the floppy diskette. Software improvements were also made to the Basic language; and languages such as FORTRAN, C, and Pascal became readily available to the scientific and engineering community.

The third generation of personal computers began with the introduction of the IBM PC (1981), and continued with such PC's as Apple's Macintosh, Commodore's Amiga, and the IBM PC-AT. They offer computing capabilities which make PC's truly productive tools for engineering design and analysis including storage of several MBytes. This evolution of the PC into a useful engineering tool is the result of many parallel developments in both hardware and software. Advances in microprocessor design and large-scale-integration technologies have led to faster clock speeds, larger word sizes, and greater addressable memory. The original Altair used an Intel 8008 microprocessor operating on 8-bit words at a clock speed of 2 MHz, while more recent systems utilize clock speeds of 20 MHz or more and 16- or even 32-bit microprocessors.

And, the future is now here as the fourth-generation of PC's based on 32-bit microprocessors, such as a MacII, the IBM System 2, and the Compaq have become widely available since early 1987. In contrast with the memory-limited first-, second- and even third-generation PC's, these new machines can address an unimaginable (to me at least) 4 GBytes of memory! To someone who learned programming on the IBM 7094 with its 32k of 36-bit words, this nearly *30,000* times increase in potential available memory (PC's with 5 MBytes are common but that's a long way from 4 GBytes) is amazing. Still to come as PC implementations are "mainframes on chips" as exemplified by the Intel 80486, for which the term "personal mainframe" has been used.

Random-Access Storage. Microprocessors able to address such huge memories would not be so worthwhile if not for the dramatic advancements in the memory chip. As a rule of thumb, the size of memory chips has increased by a factor of four every two to two-and-one-half years. The Altair started out with 256 words of memory. Today multi-MByte memories are the accepted norm. This is also due to the fact that along with the increases in memory-

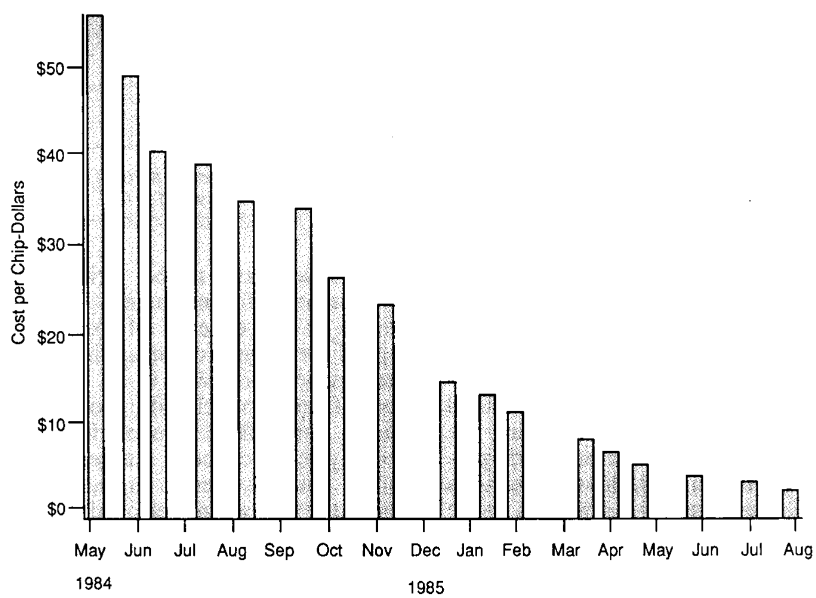


Fig. 3. Cost decrease with time of 256k dynamic memory. The dramatic fall in cost over the first year or two after introduction of a new component is a typical result that so far characterizes the "learning curve" effects we have come to expect in the production of VLSIC.

YR	Microprocessor	Bits	Clock MHz	Mem Bytes	Typical Computer
1975	Intel 8008/8080	8	2	64K	Altair,IMSAI,SOL-20
1975	Motorola 6800	8	1	64K	Altair680,SWTP6800
1977	Zilog Z80	8	2/4	64K	TRS-80,Morrow,Osbourn Comemco,Northstar
1977	MOS Tech 6502	8	1	64K	PET,AppleII,Vic20, Commodore64,Atari800
1981	Intel 8088/8086	8/16	4.77	640K	IBM PC & Compatibles
1983-6	Motorola 68000	16/32	8	16M	HP200,Macintosh,Amiga, Atari ST
1984	Intel 80286	16	6	16M	IBM PC-AT & Clones
1985	Nat'l NS32032	32	10	4G	
1985	Motorola 68010	16/32	10	4G	AT&T Unix PC
1985-6	Motorola 68020	32	16	4G	HP320,Macintosh
1986-7	Intel 80386	32	20	4G	AT&T,Compaq, IBM
1988-?	Intel 80486	32	33	4G	??

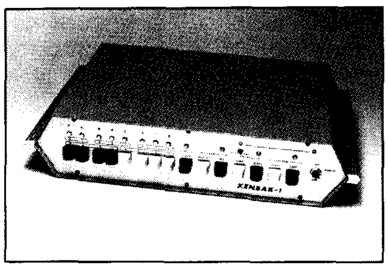
Table 1. PC Microprocessor Evolution. The question marks in the last row indicates the status existing in late 1988. The first 32-bit systems have been available for some time, and fifth-generation 486-type systems should be appearing over the next two years.

chip size, the cost of memory has decreased dramatically. Figure 4 shows how the retail cost of the 256K dynamic memory chips dropped during 1984 and 1985. Other factors can alter this decrease in price as demonstrated by the recent cost increases in the MByte RAMS. The cause was a shortage in supplies and reduced numbers of suppliers due in large part to international trade policies and agreements.

Mass-Storage. Substantial enhancements have also taken place in digital storage media. The simple cassette tape recorder with which the PET computer was equipped for program and data storage operated at data rates on the order of 10 bytes per second. The slow and unreliable cassette was soon replaced by first the 8-inch floppy diskette, then the 5-1/4 inch floppy diskette. Now, the 3.5 inch floppy diskette is the new standard. Storage capacity has increased from 80K to more than 800K bytes per floppy. Improvements have also occurred in Winchester hard-disk technology to the point where one can purchase a 20 MByte hard disk for less than $500 (U.S.) and 100 MByte drives are nearing $1,000 (U.S.) in price.

The Compact Disk ROM (Read Only Memory) is now entering the personal computer arena with prestored databases such as dictionaries and encyclopedias as well as entire software libraries. However, even more exciting is the development of the gigabyte optical WORM (Write Once Read Manytimes) disk. As one indication of evolving CD technology there is the recently announced NEXT computer, which employs an optical *read-write* disk of 240 MByte capacity. This is roughly 2400 times more storage than that provided by the floppy disks first used for PCs in the late 1970s.

Evolution of Software and PC Performance. Software has also advanced. We can observe a close relationship between the performance capabilities of the software and the hardware available on which to run it. The first PC's were severely limited in available memory, which forced software to be written in a memory-efficient manner, sometimes at the expense of execution speed for large programs. Similarly, processor word size is an important factor in scientific computations, particularly if numerical accuracy is of concern. Larger word sizes naturally result in more efficient code when performing numerical calculations, largely because double-precision operation is then not always required.



The Kenbak-1 pictured here, (1971) is believed to be the world's first personal computer. It had 256 bytes of memory.

Faster microprocessors with larger word sizes, more and more memory at ever decreasing costs, and the availability of fast mass storage devices clearly place the PC into the realm of a useful and productive engineering tool. The availability of computer languages, such as the fully ANSI Standard FORTRAN for the Macintosh and IBM, make conversion and adaptation of software developed for mainframes to PC's very easy. This makes the downloading of mainframe scientific and engineering codes and packages easier, making the PC an even more productive engineering tool. Alternately, the PC can become an effective development station for software ultimately intended for mini- or mainframe computers. The future directions of PC development can be described by such terms as "Personal Mainframe" as mentioned in Business Week magazine (September 26, 1988, issue). Other names being used are "microsupercomputer" in an attempt to convey the increasing downsizing of multi-megaflop computers to chip-size systems and their growing performance (Fig. 4).

What next?

While the growing power of mainframe computers expands the horizons of problems for which computer modeling is practicable, there is a limit to what can be accomplished through "doing more of the same." For example, if computer throughput or FLOP rate increases by a factor of 10^x, the size of the matrix that can be factored in a given clock time increases only by a factor of $10^{x/3}$. Thus, if a million-times speed increase in computer speed is achieved (i.e., $x = 6$) then the problem size or number of unknowns that can be computed in the same time increases by a factor of only 100. In order to model problems in electromagnetics, weather forecasting, quantum physics, and so forth, we need to devise alternate computational strategies in terms of mathematical formulation and its numerical treatment, as well as in the hardware design or computer architecture. We conclude with a few comments about the latter.

Hardware improvements have been predominantly responsible for the six-order-of-improvement in computer operating speed since the UNIVAC-1. These improvements can be chronicled according to the computer generation with which they are associated. First-generation computers (1950-1959) such as the UNIVAC-1 used vacuum-tube technology. The

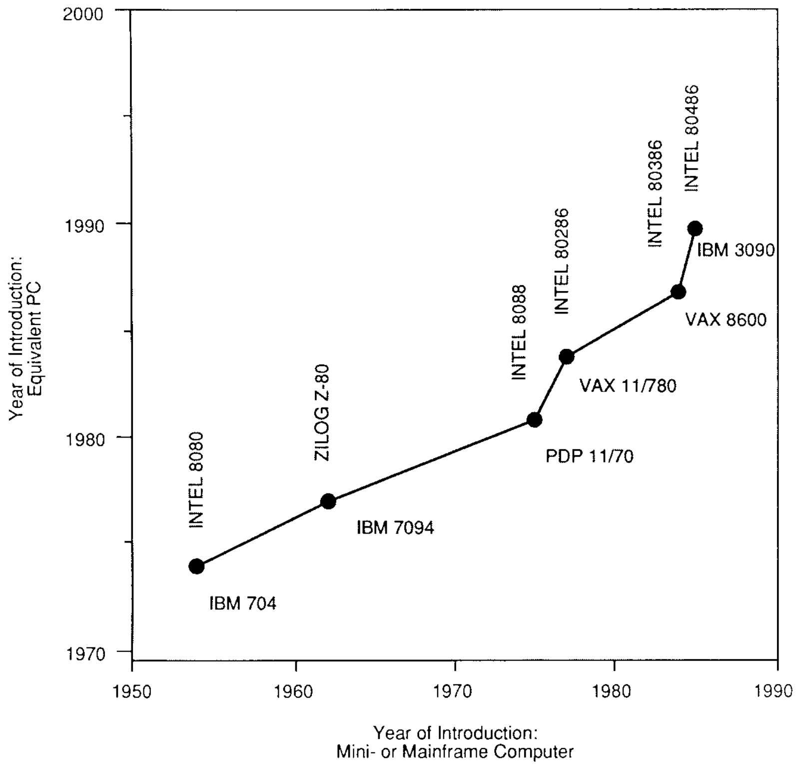


Fig. 4. PC/mainframe comparison in terms of relatively equivalent performance and year of first introduction. Whereas the first-generation PC's were about 20 years behind the equivalent mainframe when PC's first came onto the scene, they are now only five-to-ten years behind. And the gap is closing.

second generation of computers (1960-1968), which includes the IBM 7094, was based on transistors. The integrated circuit provided the innovation that made third-generation computers (1969-1977) like the CDC 7600. Appearance of the large-scale integrated circuit made possible even faster fourth-generation (1978-19??), mainframe computers, as well as the PC and workstations.

But fundamental physical limitations are being reached where straightforward extension of what has worked in the past cannot be continued indefinitely. The finite speed of light, which determines the maximum rate at which information can be propagated around a computer system, is one such limit. Another is the minimum physical size at which electronic circuits and components can be fabricated.

If component performance appears to be reaching a saturation point, then the answer for achieving still higher computer speeds must be found elsewhere. One possibility is developing alternate architectures. These might include pipelining and parallel (array) processing. In the former, increased throughput is obtained by designing a computational stream with as few "blank spots" as possible. This is done to reduce the "wait time" that otherwise occurs when an entire computation step is completed before the next one is begun. One of the first computers in which pipelining appeared was the CDC-7600 computer. Code design must take pipelining into account if its potential benefits are to be realized.

Parallel, or array, processing provides another alternative, in which the computation is shared among many, possibly identical, processors. The opportunities for achieving increased computer performance through this approach appear virtually limitless, especially with the decreased cost of microprocessors themselves, which makes "massively" parallel processing practicable. But, even more so than for pipelining, parallel processing utilized fully will require appropriate code design. This can be expected to be especially challenging since little programming experience has been acquired in that area.

An alternative to an array processor, which is a general-purpose computer, might be systolic arrays, which also consist of large collections of processing units. Systolic arrays differ from array processors in that systolic arrays are essentially "hardwired" to implement a particular algorithm or class of algorithms and have limited or no programmability. This possible disadvantage is offset by the fact that their performance for the particular problem for which they are designed can be quite impressive.

Finally, special-purpose chips, possibly having thousands to millions of gates and designed for a particular kind of computation, are beginning to appear. As the design tools become more powerful, and the hardware technology becomes more mature, the possibility of achieving "silicon" compilation, i.e., putting a compiled program directly onto a chip, will become available. Then mathematical operations most frequently encountered, such matrix manipulation as inversion, diagonalization, singular-value decomposition, etc., will be available most likely in hardware libraries. It may not be too much to expect that modeling codes will also eventually be programmed in the form of chip hardware. Computer design will have come full circle, as the first electronic computers were programmed via patch panels, the discrete-component equivalent of special-purpose chips.

Acknowledgment

Material presented here is partially excerpted from an article, "Some Aspects of the Present Status and Future Directions of Computational Electromagnetics," prepared by Fred Deadrick of Lawrence Livermore National Laboratory and the author for JINA '86 (International Meeting on Antennas in Nice, France, 1986).

Read more about it

- Bell, C. G., "The Mini and Micro Industries," *IEEE Computer*, October, pp. 14-30, 1984.
- Metropolis, N., J. Howlett, and Gian-Carlo Rota (editors), *A History of Computing in the Twentieth Century*, Academic Press, Inc., New York, NY, 1980.
- Shurkin, J., *Engines of the Mind*, Washington Square Press, New York, 1984.

About the author

Dr. Edmund K. Miller works at General Research Corporation, Santa Barbara, California. He is a *Potentials* Associate Editor. □