INTRODUCTION

In order to assess the impact of any technological breakthroughs on the computer user, it is first necessary to understand the nature of the computer user's problem in its present context. Indeed, it is a source of considerable disappointment to some that radical improvements in technology over the past several decades have resulted in noncommensurate impact on the user. The purpose of this paper is to examine the question: "What will be the impact of integrated circuits or integrated electronics on computer architecture?" In the first part of the paper we will study the present picture of computing and show that, if no changes are made from present operating procedures, the impact will be small. In the second part of the paper, we analyze several aspects of computing to find conditions which will enlarge the impact. The final part of the paper will list a number of open problems in computing for which integrated electronics may provide a solution.

PRESENT STATUS

To begin with we define the term architecture, a term first used in this context by F. P. Brooks, Jr.1 Computer architecture is the structure of the process that solves a user's problem. It cannot be regarded as simply a study of a processor itself but rather as a disciplinary concern for the entire environment of the user. This process is much more than one machine or one piece of apparatus. It is, in fact, a whole ensemble of equipment, facilities, people, arithmetic tools, linguistic skills, and economic conditions.

In order to understand the computer user's problem, we must introduce a measure of effectiveness. The measure I have chosen is simply cost per computation (for a standard problem). We are first going to determine the fraction of computing cost represented by processor rental, and then determine the fraction of processor rental attributable to circuitry. In any given installation computation cost is directly related to the annual budget. Expense items in a budget usually include: (1) equipment rental; (2) systems programming; (3) applications consulting; (4) training of users; (5) supplies, including paper and forms; (6) general overhead, building, etc.; (7) nonprofessional personnel, including key-punch operators, machine-room operators; and (8) management. Figure 1 seems to be a fairly representative picture of the budget of several computing centers. Even though the numbers in Fig. 1 are not precise, the processor costs (the area traditionally associated with integrated circuitry) rarely get above 10% of the total budget. The costs described in Fig. 1 are still frequently a small percentage of the true cost of the solution of a user's problem. Neglected
are the many program reruns for debugging, numerical difficulties and program restructuring due to improved problem insight.

Circuitry costs represent a similarly small fraction of the processor costs. The manufacturer's costs (Fig. 2) must reflect programming systems, servicing, maintenance, marketing costs, personnel overhead (which includes support activities and management), basic research and general overhead (facilities, etc.) items, as well as engineering design, records, maintenance, and variable manufacturing costs. The variable manufacturing costs include the circuitry, first-level packaging and assembly, second-level packaging and wiring, power supplies, frames, covers, cables and mechanical hardware, third-level assembly, and manufacturing debugging.

The last breakdown is open to the comment that the manufacturer's costs are not directly related to the price the user pays. This further diminishes the percentage of computing costs represented by circuitry. Thus, if circuitry costs zero, the user's cost of computing would not be affected noticeably. The basic assumption, of course, is that computing systems would be built in much the same way as they have been.

THE ARCHITECTURAL PROBLEM

In order to avoid the pitfalls of the above assumption, we reexamine the problem.

There are three aspects of the architectural problem:

1. The external, man-system communications problem.
2. The internal, system-system communications and processing problem.

The Man-Systems Problem

The fact that the decision-making electronic components represent only a very small fraction of the user cost is an indication that the interface between these components and the ultimate user, man, is a poor one or at least an inefficient one.

The few percent of total cost represented by circuitry is the cost component that actually solves the problem. The remaining costs allow man to communicate with this circuitry.

This communications interface between man and machine is characterized by the following aspects:

1. Bandwidth: total amount of information provided by the machine per unit time.
2. Storage.
3. Language.
4. Responsive interaction between man and machine.

Bandwidth, per se, is not a significant problem if the storage, language, and responsive interaction problems can be solved. Presently typical problems

Figure 1. User's costs.

Figure 2. Manufacturer's costs of a processor.
run hundreds of pages of output. Clearly, much of the output is useless. The cause of the excess is that the human cannot interact readily with the system. The time lags are too great between the original query, the time the machine responds, and the still later time when the human can return to the system for further queries. Thus, the human demands exhaustive amounts of output data to make up for the lack of interaction.

Storage must be available on a semipermanent medium in such a fashion that it can be reanalyzed or acted on at a later, more convenient moment. The paper medium, presently used, has low cost, permanence and legibility. Its disadvantages are that printout is a relatively slow process and that the result is not easily machine-readable. Any new interface must compete with its advantages as well as solve its problems.

The third aspect of the problem is language. For various reasons, the computer as a processor does not work at the same linguistic level or in the same semantic mode that the human does. The human must adapt himself, therefore, to learn highly formalized and stylized languages. This presents a barrier to the interface since the burden is on man. As pointed out in one of the other papers in this session,^ 2 tradeoffs can be made to some degree to allow more ready linguistic interplay between man and the machine.

Attempts to implement more responsive interaction between man and machine have usually taken the form of time-sharing systems^ 3,4 or "the utility concept." These user-shared systems are not the most efficient computing mechanism (due to the overhead of swapping storage and control between users). However, they do represent a higher level of man-system efficiency which, as we have discussed, is the more significant. While integrated circuitry may play a role in increasing the internal efficiency of the user-shared processor (via buffering) in this section we will consider the device with which man communicates: the display. The immediate access display is a necessary condition for the implementation of a responsive system.

The display itself has developed in phases. Early displays were unidirectional, acting as output-only rather than I/O devices. Their use was mainly for data compression or formatting (as in curve plotting). Later, semiresponsive displays were developed where interaction was restricted to manual keys or paper tape. More recently, the availability of the so-called light-pen gives a new dimension to the interaction, with an immediate program interrupt to focus on any point of the display. In order to satisfy the aforementioned man-machine requirements, there should be another phase of development characterized by:

1. Interactive Display—The continued use of the light-pen with refinements in electronics would be suitable. The major problem is the development of sufficiently versatile programming languages directed at graphic, interactive devices.

2. Solid-State Display—The electrical mismatch between cathode-ray tube displays and integrated circuitry is an unfortunate cost impediment. A low-voltage, two-dimensional, variable-size solid-state display would be a major breakthrough.

3. Storage—A truly useful display should have facility for large quantity of storage which is machine-readable and console-readable. A photographic medium is an example of a low-cost, permanent storage, suitable for such purposes as data reanalysis, program storage, etc. Current technology allows the resolution of $10^4$ to $10^8$ bits on a $2 \times 2$ or $70$mm film slide without involving mechanical motion. Photography can fulfill both of the basic functions of storage: the final (output) document, which needs only to be man-readable, and the interactive, temporary results, which should be both machine and man (via console) readable.

4. Adaptability—A versatile console must be usable in conjunction with any standard communication link (telephone, etc.) — i.e. it must be proximate to the user.

5. Cost—The overall cost should not exceed keypunch cost. It is here that integrated circuits could play the key role by providing complex functions at reasonable cost.
"utility" computers, I feel that the latter will offer the general user a much more attractive interface. By interface, I mean libraries of programming systems and subroutines, the availability of human consultants for difficult problems, and the availability of, or remote access to, very large data files.

In spite of the obvious importance of the display area, quantitative data on relevant aspects of human learning with respect to the computing process are scarce. And while an encouraging amount of interest has been shown by manufacturers, too often the context of their interest is gimmickery, rather than studied usefulness.

**Systems—Systems Communications and Processing Problem**

Within the system, communications is an electrical signal processing problem and has been widely discussed. At least within the processor, large monolithic arrays will be the basic decision subunit of the system. Storage and I/O storage functions pose a difficult problem. Storage is usually regarded as a hierarchy in which the most immediate element, the main storage, has fast access with limited amounts of storage capabilities. Higher up on the hierarchy are the drum and the disk, which have substantial capacities, but have latencies of several milliseconds to access the first piece of data. Once access is achieved, transfer is usually made at rates substantially less (a factor of 10) than the main storage rate. This mismatch and the mismatches at higher hierarchical levels such as tape and I/O represent a key systems problem.

While integrated circuitry holds promise for some implementations of main memory, it is still prohibitively expensive to consider as a storage hierarchy replacement. The only hope, it would seem to me, is the use of integrated circuits in conjunction with an inherently high-density storage medium. Developments along this line would be most welcome.

With respect to the processor and main storage, use of monolithic arrays has two important implications: (1) Logic design must be viewed as an insular problem wherein substantial amounts of circuitry are connected on a single chip; the criterion which determines the size of these logical islands is the number of external connections which are made to it. This requires that optimum use of connection bandwidth be made between units; otherwise, cost premiums will exist due to first and second level packages. Also, for similar speeds, the communication to an external unit demands substantially more power than to an internal circuit decision element. Hence, to make most efficient use of power, the minimum number of external connections should be made. Thus one would tend to make maximum use of external connections or the bandwidth of the external connections. (2) The most significant factor in costs is not the number of circuits per chip, or even by the total number of chips employed, but rather the number of different kinds of chips.

Thus, the problem reduces itself to one of partitioning the system into a set of logical islands, each with a minimum number of external interconnections which involve the minimum number of different type of islands. We will examine main storage, controls, data paths, arithmetic, and general logic design considerations as the five fundamental areas of the system to determine the influence of these constraints.

1. **Main Storage.** The nature of storage is such that the connectivity and replication constraints are always satisfied. The number of input/output pins increase linearly with the logarithm of the number of bits contained on the chip. The problem of main memory is not these initial constraints but rather the economic advantage of competitively established technology, such as magnetic core or film, which have very efficient methods of interconnection and data storage. For integrated circuits to provide a valuable storage function, one must take advantage of the availability of its logical power. This allows consideration of such storage arrangements as the associative or content-address memory. In the associative memory, a known subportion of a data word is presented to the memory and inquiry is made on the appropriate portion of all such words contained in that storage. If match is detected, it is indicated, and the remaining (associated) information contained in the word is retrieved. With such memory, for example, sorting of data is no longer required. The cell structure of the associative memory is repetitive, and with the absence of address inputs the cells require less interconnection than a corresponding array of coordinate-addressed storage.

2. **Controls.** A traditional implementation of the control function is difficult if not impossible to duplicate on the basis of array-replicated circuitry. Thus it is most natural to consider micro-programming for this function. Here storage plays the role of
combinatorial and sequential decoder. The instruction represents merely the address of the initial element in the sequence of gating descriptions which are retrieved and acted upon. The micro-program arrangement is much more versatile than the traditional implementations. The fact that it is reloadable makes possible rearrangements of major portions of the machine and introduction of new instruction repertoires. Most micro-program machines to date have been of the read-only variety, where changes in the micro-program storage is an involved task. There is no reason why, with availability of integrated circuits, a machine could not dynamically restructure itself by changing the contents of its micro-program storage.

3. Data Paths. The principal functions of a data path are transfer of information from one subunit to another and provision of temporary storage at these entry and exit points. The data path then is a communications medium for logical units or subunits of the system. Data path circuitry may well be thought of as integral to the logical unit with which it is associated, thus insulating the unit from the outside environment. Further discussion on the external communications implications of the data paths will be made below.

4. Arithmetic. Arithmetic has equipment implications because it involves communications not only from the data paths but also across the numerical field of the word operated on. In selecting an algorithm for add, multiply, or divide, one must take care to maximize the efficiency of the implementation; for example, shifts could be accomplished on a digit rather than a bit basis, thus simplifying the number of interconnections for multiplying.

5. Considerations in Logical Design Using Integrated Circuits. As with the intra-system problems, the logic design problems can be divided into two groups: the internal problems, and the external problems of communication between modules:

(a) Internal logical design. Decision elements internal to a monolithic array have an ideal environment, with excellent tracking of component values and power supply tolerances. Logical structures, such as multivalued logical structures, threshold logic gates, and specialized current summers, which were marginal at best in discrete implementations, now become realizable. Respectable speeds may be maintained at the expenditure of much less power than the externally directed circuit. Thus the only problem is to keep as many logical nodes as possible free from the restrictions of the outside environment.

(b) External logical design. On a typical small card of about 25 square inches, fully populated with “flat packs”, less than one per cent of the area is taken up by decision-making logical components; fully 99% of the area is devoted to leads and to connections. Power is also a problem, but power is normally associated with driving the stray connector and line capacities.

It is also interesting to note that present communication techniques are very inefficient. For example, it is rare that any line passes more than $50 \times 10^6$ bits per second; whereas the capacity of that line exceeds $10^9$ bits per second. This would be true for most terminated transmission lines. To relieve the communication-interconnection problems, it would be sufficient to serialize the data and maximize the transmission rate. Of course this introduces complexity in the array, because now the information must be commutated; but since we have presumed that the array originally had substantial amounts of logical decision capabilities this complexity would be acceptable.

Thus we may see the partial return to the old serial computer where we have one input line, one output line operating at very high speeds, and processing done within the array at lower speeds and in parallel.

Problem-System Interface

Thus far, we have discussed the potential for integrated electronics in the man-system relationship and in intrasystem optimization. There is a much broader level of application of integrated electronics: the direct problem-system implementation, with man eliminated as intermediary. “Closing the loop” has been the preserve of analog computers due to cost factors. Note that it is exactly in such a “people free” environment that the cost potential of integrated electronics is maximized. But even this misses the
singular promise of integrated electronics in the development of low-cost, versatile, high-performance digital transducers. This “digitizing the universe” (environment) is one of the truly expansive markets of the future.

OPEN PROBLEMS

Over the past decade, in device areas, logic design, and systems work, we have optimized our components along several dimensions: ultrahigh-speed computing, minimum-cost computing, or cost-performance computing. I believe many of our problems lie outside these broad classifications, and the pursuit of these open problems would provide substantial “fallout,” namely a much greater understanding to the entire computing process. They represent overlooked dimensions of computing. In addition to man-machine interactive computing, previously discussed, some of these areas might be:

1. The Ultra-Reliable Computer. The problem is to build a computer whose failure rate is arbitrarily small. It is replaced rather than field-serviced. It should be able to operate in spite of any single failure, and there should be some confidence of operating over a large class of multiple failures. In order to take intelligent steps toward the realization of such a system, reexamination of many areas is required: device failures, logic design techniques, coding techniques, diagnostics, new concepts in input/output equipment, and many others. It is clear that much more reliable systems can be built than the traditional triple-modular redundancy systems; but even more significant than the resulting system itself would be the improved understanding of failure mechanisms in general.

2. Physical Environment Area. There would be numerous advantages in a computer which could operate under the most extreme environmental conditions and/or use the lowest possible power. Here again, this is a problem of more than just a device; it is a problem in circuit design and in logic and system techniques.

3. Electronic Library. The third area is the problem of the electronic library. This involves the construction of a very large (about $10^{10}$ bits) electronic storage with any bit accessible in under 10 microseconds. In present systems such arrangements must be handled, in part, by electromechanical storage media, and random accessing must be in the order of milliseconds to seconds; thus many problems which have requirement for very large random memories (with interaction among any and all data in the memory) take excessively long times to be performed. The existence of such an electronic library would present a three or four order-of-magnitude breakthrough for these problems. To be most useful, such a memory should be operable in either the associative or coordinate address mode.

4. The One-Chip Computer. The one-chip computer has been long discussed by Holland and others. While each chip is a complete computer, its nature would be computationally atomic. The utility of the one-chip computer is in the possibility of assembling many chips into a vast network, the size of the network determining the performance of the total processor. Thus, the processor may be expanded to suit any computational need.

5. The Maximum Efficiency Computer. It is recognized that most general-purpose computers are general-purpose only in the sense that they are equally inconvenient for all to use. It should be possible to design a system whose organization is flexible and versatile enough so that it might undergo a maximum restructuring by each user in a dynamic fashion. Thus, it might employ the micro-programming techniques previously discussed and incorporate them into a versatile arrangement of data paths and logical subunits. The user would first describe the machine he would like to have work on his problem, and essentially load the instruction repertoire that he desires into this micro instruction storage. He then performs his program on the computer that he has designed to be optimum for the solution of his problem.

I feel that these problem areas have merit in themselves—as much as very-high-speed computing—and that they warrant the attention and interest normally associated with the more familiar computing dimensions. They would not necessarily be the fastest and/or the most economical system; rather hopefully, they would be useful new horizons in computing.

CONCLUSIONS

We have seen that the present interface between man and machine is inadequate, and properly designed displays may provide a significant part of the solution. Within the framework of a computing system, integrated electronics is equivalent to monolithic
integrated circuits. There appear to be two important considerations for the logic designer: maximizing the number of decision elements per interconnection, and minimizing the number of different array types used.

On a broader level, the more removed from the system man becomes, the more the potential of the integrated electronics can be realized.

The message of this paper is a simple one. It is that integrated electronics should be much more than monolithic circuitry. The entire realm of physical resources, integrated with the systems resources we call computer architecture, should be directed toward furthering the best possible solution of the computer user's problems. The traditional form of these problems frequently masks their essential nature. It is only when this essential nature is understood that optimum technologic implementations are possible.

REFERENCES


